



Development of a Maintenance Analytics Framework for the Diagnosis and Prognosis of Marine Systems

Authored by

Christian Velasco Gallego

*A thesis presented in fulfilment of the requirements for the degree of Doctor of
Philosophy*

Department of Naval Architecture, Ocean and Marine Engineering

University of Strathclyde

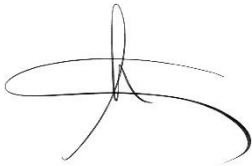
Glasgow, UK

2022

Declaration

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

The copyright of this thesis belongs to the author under the terms of the United Kingdom Copyright Acts as qualified by University of Strathclyde Regulation 3.50. Due acknowledgement must always be made of the use of any material contained in, or derived from, this thesis.

A handwritten signature in black ink, appearing to be 'Christian Velasco Gallego', with a stylized, cursive script.

Signed: Christian Velasco Gallego

Date: 08/12/2022

“Quan em poso davant d’una tela, no sé mai què faré, i soc jo el primer sorprès del que surt.”

“Cuando me coloco delante de un lienzo, no sé nunca lo que voy a hacer; y yo soy el primer sorprendido de lo que sale.”

“When I stand in front of a canvas, I never know what I’m going to do – and nobody is more surprised than I at what comes out.

– Joan Miró

Acknowledgements

I would like to express my sincere appreciation to Dr. Iraklis Lazakis for supervising this research study. I would also like to thank Dr. Beatriz Navas de Maya, Dr. Maria Clementina Ramirez Marengo and Conor Kennedy Burke for their valuable suggestions.

I would also like to acknowledge the support given by my family María, Juan Manuel, Vero, and my friends Ale, Angie, Cinta, Cristian, David, and Joana. Especially to Ale and his family, who taught me the importance of resilience and persistence.

To all my support circle, but especially to my grandparents María and Tomás.

Research Output

Journal Papers

Velasco-Gallego C., Lazakis I., 2022a. Mar-RUL: a remaining useful life prediction approach for fault prognostics of marine machinery. Engineering Applications of Artificial Intelligence (Revised version submitted, waiting final decision).

Velasco-Gallego C., Lazakis I., 2022b. Development of a time series imaging approach for fault classification of marine systems. Ocean Engineering 263, pp. 1-10, doi: <https://doi.org/10.1016/j.oceaneng.2022.112297>.

Velasco-Gallego C., Lazakis I., 2022c. Analysis of variational autoencoders for imputing missing values from sensor data of marine systems. Journal of Ship Research 66 (3), pp. 1-11, doi: <https://doi.org/10.5957/JOSR.09210032>.

Velasco-Gallego C., Lazakis I., 2022d. RADIS: a real-time anomaly detection intelligent system for fault diagnosis of marine machinery. Expert Systems with Applications 204, pp. 1-13, doi: <https://doi.org/10.1016/j.eswa.2022.117634>.

Velasco-Gallego C., Lazakis I., 2022e. A real-time data-driven framework for the identification of steady states of marine machinery. *Applied Ocean Research* 121, pp. 1-12, doi: <https://doi.org/10.1016/j.apor.2022.103052>.

Velasco-Gallego C., Lazakis I., 2021a. A novel framework for imputing large gaps of missing values from time series sensor data of marine machinery systems. *Ships and Offshore Structures*, pp. 1-10, doi: <https://doi.org/10.1080/17445302.2021.1943850>.

Velasco-Gallego C., Lazakis I., 2020. Real-time data-driven missing data imputation for short-term sensor data of marine systems. A comparative study. *Ocean Engineering* 218, pp. 1-23, doi: <https://doi.org/10.1016/j.oceaneng.2020.108261>.

Conference Papers

Velasco-Gallego C., Lazakis I., 2022f. Pre-ONA: a data pre-processing tool for marine systems sensor data. *Proceedings of the International Conference on Postgraduate Research in Maritime Technology (PostGradMarTec 2022)*, 8-9th November 2022, online.

Velasco-Galelgo C., Lazakis I., 2022g. Analysis of Time Series Imaging Approaches for the Application of Fault Classification of Marine Systems. Proceedings of the 32nd European Safety and Reliability Conference, 28th August – 1st September 2022, Dublin (Republic of Ireland).

Velasco-Gallego C., Lazakis I., 2021b. A real-time semi-supervised anomaly detection framework for fault diagnosis of marine machinery systems. SNAME WES Paper Contest 2021 (SNAME Western Europe Section 11th Annual UK Symposium), 8th October 2021, London (United Kingdom).

Velasco-Gallego C., Lazakis I., 2021c. Data imputation of missing values from marine machinery systems sensor data. Evaluation, visualisation, and sensor failure detection. Proceedings of the 2021 RINA Maritime Innovation and Emerging Technologies, 17-18th March 2021, online.

Invited Presentations

Lazakis I., Velasco-Gallego C., 2022. Smart condition monitoring and ship maintenance. Held at Politeknik Perkapalan Negeri (PPNS), Indonesia, online.

Prizes Won

SNAME Graduate Paper Honor Prize Winner 2022. Degree of recognition: International. Awarded at SNAME Maritime Convention (SMC) 2022. Houston, USA (September 2022).

SNAME Western Europe Section Paper Contest 2nd Prize Winner 2021. Degree of recognition: International. Awarded at 11th Annual UK Symposium (London - UK, October 2021).

Contents

Declaration	i
Acknowledgements.....	ii
Research Output.....	iii
Contents	vii
List of Figures.....	xiii
List of Tables.....	xxiv
Acronyms.....	xxviii
Abstract.....	xxxiii
Chapter 1. Introduction	1
1.1. Chapter Overview.....	1
1.2. Maritime Industry and Digitalisation.....	1
1.3. Research Direction.....	9
1.3.1. Research Question	9
1.3.2. Aim and Objectives	10
1.4. Thesis Layout	12
1.5. Chapter Summary	15
Chapter 2. Literature Review.....	16
2.1. Chapter Overview.....	16
2.2. From Reactive to Smart Maintenance	18
2.2.1. Implementation of Maintenance Strategies within the Shipping Industry.....	18
2.2.1.1. Reactive Maintenance (RM)	19

2.2.1.2. Preventive Maintenance (PvM).....	19
2.2.2. Cutting-Edge Technologies: Perspectives on Smart Maintenance.....	20
2.2.3. Applicability of Smart Maintenance in the Shipping Sector.....	26
2.3. Implementation of Maintenance Analytics within the Shipping Industry	32
2.3.1. Data Pre-processing.....	32
2.3.1.1. Data Imputation	39
2.3.1.2. Identification of Operational States.....	52
2.3.2. Diagnostic Analytics.....	53
2.3.2.1. Fault Detection	54
2.3.2.2. Fault Identification	62
2.3.3. Predictive Analytics	68
2.4. Identified Gaps	73
2.5. Chapter Summary	77
Chapter 3. Methodology: Part I. Introduced Novelties and Overview of the Developed Holistic MA Framework.....	78
3.1. Chapter Overview.....	78
3.2. Novelties.....	78
3.3. Overview of the Developed Frameworks	81
3.4. Chapter Summary	85
Chapter 4. Methodology: Part II. The Data Pre-processing Module.....	86
4.1. Chapter Overview.....	86
4.2. Comparative Methodology of Univariate and Multivariate Imputation Techniques	87
4.2.1. Data Preparation.....	89

4.2.2. Missing Values Generation	93
4.2.3. Time Series Cross-validation.....	93
4.2.4. Univariate Imputation.....	94
4.2.5. Multivariate Imputation	99
4.2.6. Evaluation.....	109
4.3. Hybrid Data Imputation Framework	113
4.3.1. Steady states' identification.....	115
4.3.2. Univariate Imputation. First-order Markov Chain.....	120
4.3.3. Multivariate Imputation	124
4.4. Analysis of LSTM-based Variational Autoencoders for Regression for Data Imputation	128
4.4.1. Data Pre-processing.....	129
4.4.2. LSTM-VAE-based Regressor Analysis.....	130
4.4.3. Evaluation.....	133
4.4.4. Comparative Study.....	133
4.5. A Novel Framework for the Identification of Steady States	134
4.5.1. Data Pre-processing.....	137
4.5.2. Image Generation	137
4.5.3. Connected Component Analysis	137
4.5.4. Data Post-processing.....	140
4.6. Chapter Summary	141
Chapter 5. Methodology Part III. The Diagnostic and Predictive Analytics Modules	142
5.1. Chapter Overview.....	142

5.2. Diagnostic and Predictive Analytics Module Overview	142
5.3. Diagnostic Analytics	143
5.3.1. Fault Detection	143
5.3.1.1. Time Series Denoising	146
5.3.1.2. Fault Detection	148
5.3.2. Fault Identification.....	151
5.3.2.1. Time Series Imaging.....	152
5.3.2.2. Image Classification with ResNet50V2 and CNNs.....	153
5.4. Predictive Analytics.....	155
5.4.1. Remaining Useful Life (RUL) Prediction Framework	155
5.4.1.1. Data Augmentation.....	157
5.4.1.2. Condition Indicator Estimation through the Implementation of the Piecwise Linear Algorithm.....	158
5.4.1.3. Time Series Imaging through the Implementation of the First-order Markov Chain Model	159
5.4.1.4. Remaining Useful Life (RUL) Prediction.....	159
5.4.1.5. Weighted Average Ensemble.....	161
5.5. Chapter Summary	162
Chapter 6. Case Studies and Results: Part I. The Data Pre-processing Module	163
6.1. Chapter Overview.....	163
6.2. Case Study 1. Comparative Methodology for Data Imputation.....	163
6.3. Case Study 2. Hybrid Imputation Framework	190
6.4. Case Study 3. Analysis of LSTM-based VAE Regressor for Data Imputation	197
6.5. Case Study 4. Novel Framework for the Identification of Steady States	211

6.6. Chapter Summary	222
Chapter 7. Case Studies and Results: Part II. The Diagnostic and Predictive Analytics Modules	223
7.1. Chapter Overview.....	223
7.2. Case Study 5. Fault Detection as Part of the Diagnostic Analytics Module	224
7.3. Case Study 6. Fault Identification as Part of the Diagnostic Analytics Module	232
7.4. Case Study 7. Predictive Analytics Module	245
7.5. Case 8. MA framework	256
7.6. Chapter Summary	274
Chapter 8. Discussion and Conclusions	276
8.1. Chapter Overview.....	276
8.2. Generated Novelties.....	276
8.3. Original Contribution to the Industry.....	280
8.4. Assumptions and Challenges.....	281
8.5. Recommendation for Future Work	282
8.6. Fulfilment of Aim and Objectives	290
8.7. Chapter Summary	295
Bibliography	296
Appendix A. SWOT Analysis of Maintenance Activities.....	330
A.1. Strengths	330
A.2. Weaknesses	330
A.3. Opportunities.....	331
A.4. Threats	333

Appendix B. Literature Review Summary Tables	334
Appendix C. Bibliometric and Taxonomy Results of Anomaly Detection Studies .	346
C.1. Bibliometric Analysis.....	346
C.2. Taxonomy of Anomaly Detection Studies.....	355
Appendix D. Main Results of Case Study 8. MA Framework.....	379
D.1. Main Engine Water Cooling System	379
D.2. Main Engine Cylinder 1	398
D.3. Main Engine Cylinder 2	417
D.4. Main Engine Cylinder 3.....	436
D.5. Main Engine Cylinder 4.....	454
D.6. Main Engine Cylinder 5.....	472
D.7. Main Engine Cylinder 6.....	490
D.8. Turbocharger (TC).....	508

List of Figures

Fig. 1.1. Most significant technologies (aside from low/zero-carbon R&D) that will drive change in the shipping industry over the next five years (adapted from Lloyd's List, 2021).....	3
Fig. 1.2. Best investment opportunity for shipping in 2021 (adapted from Lloyd's List, 2021).....	3
Fig. 1.3. Merchant vessels in accidents by nature of accident when considering UK merchant vessels (≥ 100 gt) throughout 2020 (adapted from MAIB, 2020).	5
Fig. 1.4. Interconnections between objectives and approaches/actions to be taken..	12
Fig. 2.1. Areas of investigation addressed in the literature review.....	18
Fig. 2.2. Maintenance analytics summary.	24
Fig. 2.3. Relationship and summary diagram of the modules that constitutes PHM.	26
Fig. 2.4. Summary of the SWOT analysis of maintenance activities.....	32
Fig. 3.1. Summary of the novelties introduced in this thesis.	80
Fig. 3.2. Graphical representation of the MA framework architecture.	84
Fig. 4.1. Graphical representation of the comparative methodology of univariate and multivariate imputation techniques.....	89
Fig. 4.2. Time series cross-validation.....	94
Fig. 4.3. Graphical representation of the hybrid data imputation framework.	115
Fig. 4.4. Graphical representation of framework developed for the analysis of LSTM-based VAE regressor for data imputation.	129
Fig. 4.5. Diagram of the VAE-based regression model modified to include LSTM layers in both encoder and decoder.	132
Fig. 4.6. Novel framework for the identification of steady states.	136
Fig. 4.7. Graphical representation of the 4-connected neighbourhood.....	139
Fig. 4.8. Connected component analysis phase representation.....	139
Fig. 4.9. Graphical representation of the post-processing phase.....	140

Fig. 5.1. Overview of the diagnostic and predictive analytics modules.	143
Fig. 5.2. Graphical representation of the fault detection framework.	145
Fig. 5.3. VAE architecture.	146
Fig. 5.4. LSTM cell architecture.	148
Fig. 5.5. Graphical representation of the proposed methodology.	152
Fig. 5.6. Graphical representation of the predictive analytics module.	157
Fig. 5.7. Graphical representation of the Markov-CNN architecture for regression.	159
Fig. 5.8. Graphical representation of the 1D-CNN architecture.	160
Fig. 6.1. Time series plot of the main engine rotational speed.	165
Fig. 6.2. Time series plot of the main engine power.	166
Fig. 6.3. Time series plot of the main engine fuel flow rate.	166
Fig. 6.4. Time series plot of the scavenging air pressure of the scavenge air receiver system.	167
Fig. 6.5. Time series plot of lubrication oil inlet pressure.	167
Fig. 6.6. Time series plot of the jacket water cooling system inlet pressure.	168
Fig. 6.7. Time series plot of the turbine lubrication oil inlet pressure of the turbocharger.	168
Fig. 6.8. Heatmap plot of Pearson's correlation coefficient matrix.	170
Fig. 6.9. Heatmap plot of Spearman's rank correlation coefficient matrix.	170
Fig. 6.10. Comparison between observed values of the main engine power parameter and imputed values through the implementation of the machine learning and time series forecasting models.	174
Fig. 6.11. Time series plot of the main engine rotational speed.	191
Fig. 6.12. Time series plot of the main engine power.	192
Fig. 6.13. Time series plot of the main engine fuel flow rate.	192
Fig. 6.14. Time series plot of the scavenging air pressure.	193
Fig. 6.15. Large gaps imputation of the main engine rotational speed.	195
Fig. 6.16. Large gaps imputation of the main engine power.	196
Fig. 6.17. Large gaps imputation of the main engine fuel flow rate.	196

Fig. 6.18. Large gaps imputation of the scavenging air pressure of the scavenge air receiver.	197
Fig. 6.19. Time series plot of the diesel generator power.	199
Fig. 6.20. Parameters' selection of the GMMs.	200
Fig. 6.21. Histograms of the monitored features (P1-P14).	201
Fig. 6.22. Comparison between observed values and imputed values of the diesel generator power (the line refers to the observed values, whereas the scatter points refer to the imputed values).	210
Fig. 6.23. Comparison between observed values and imputed values of the diesel generator power when the mean imputation is applied, and the missing ratio is 0.15 (the line refers to the observed values, whereas the scatter points refer to the imputed values).	210
Fig. 6.24. DG1 power parameter time series plot.	211
Fig. 6.25. DG2 power parameter time series plot.	212
Fig. 6.26. DG3 power parameter time series plot.	212
Fig. 6.27. Steady states' identification for the DG1 power parameter.	214
Fig. 6.28. Steady states' identification for the DG2 power parameter.	214
Fig. 6.29. Steady states' identification for the DG3 power parameter.	214
Fig. 6.30. Steady states' identification for DG1 power parameter sequence based on the results obtained from the proposed methodology.	216
Fig. 6.31. Steady states' identification for DG2 power parameter sequence based on the results obtained from the proposed methodology.	216
Fig. 6.32. Steady states' identification for DG3 power parameter sequence based on the results obtained from the proposed methodology.	216
Fig. 6.33. Estimation of the Silhouette and Davies-Boulding indices for DG1, DG2, and DG3 power parameter, respectively.	217
Fig. 6.34. Steady states' identification for DG1 power parameter sequence based on the results obtained from the <i>k</i> -means application.	218

Fig. 6.35. Steady states' identification for DG2 power parameter sequence based on the results obtained from the <i>k</i> -means application.	218
Fig. 6.36. Steady states' identification for DG3 power parameter sequence based on the results obtained from the <i>k</i> -means application.	218
Fig. 6.37. Selection of the number of components for DG1, DG2, and DG3 power parameter, respectively.	220
Fig. 6.38. Steady states' identification for DG1 power parameter sequence based on the results obtained from the GMMs with EM algorithm application.	220
Fig. 6.39. Steady states' identification for DG2 power parameter sequence based on the results obtained from the GMMs with EM algorithm application.	221
Fig. 6.40. Steady states' identification for DG3 power parameter sequence based on the results obtained from the GMMs with EM algorithm application.	221
Fig. 7.1. Time series plot of the fourteen monitored parameters.....	226
Fig. 7.2. Steady states' identification for the diesel generator power parameter.....	226
Fig. 7.3. Average NRMSE with CI 95% of the analysed parameters with injected anomalies in parameter 4.	229
Fig. 7.4. NRMSE with CI 95% of parameters 0, 4, and 9 divided by instance with injected anomalies in parameter 4.....	229
Fig. 7.5. Average NRMSE with CI 95% of the analysed parameters.	230
Fig. 7.6. NRMSE with CI 95% of parameters 0, 4, and 9 divided by instance.	230
Fig. 7.7. Image generated for instance 0 and simulation 0.....	231
Fig. 7.8. Time series plot of the cooling air temperature monitored parameter.....	233
Fig. 7.9. Original sequence that has been altered to simulate the distinct abnormal operational sequences.....	235
Fig. 7.10. Simulated sequence with (a) a point anomaly, (b) two-point anomalies, (c) multiple point anomalies, (d) collective anomalies, (e) degradation, and (f) transition occurrences between steady operational states.	236
Fig. 7.11. Example of a normal sequence encoded into an image.....	238

Fig. 7.12. Example of images generated that contain (a) a point anomaly, (b) two-point anomalies, (c) multiple point anomalies, (d) collective anomalies, (e) degradation, and (f) transition occurrences between steady operational states.	238
Fig. 7.13. Confusion matrix for multi-class classification with n classes. The estimation of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) is presented when considering a class k ($0 \leq k \leq n$).	240
Fig. 7.14. Time series plot of the turbocharger exhaust gas outlet temperature monitored parameter.....	245
Fig. 7.15. (a) Example of an original subsequence plot (b-f) Example of a synthetic sequence plot.....	249
Fig. 7.16. (a) Example of a synthetic sequence (b-f) Example of a degradation simulation.	251
Fig. 7.17. RUL prediction for the turbocharger exhaust gas outlet temperature. ...	252
Fig. 7.18. Graphical representation of the cylinder 1 exhaust gas outlet temperature parameter.....	256
Fig. 7.19. Identification of the operational states for the monitored parameter.	257
Fig. 7.20. Example of normal sequences.	259
Fig. 7.21. Example of collective anomalies.	260
Fig. 7.22. Example of degradation patterns.....	261
Fig. 7.23. Histogram of the reconstructed errors of the normal sequences (test set).	263
Fig. 7.24. Histogram of the reconstructed errors of the sequences with collective anomalies.	263
Fig. 7.25. Histogram of the reconstructed errors of the sequences with degradation patterns.	264
Fig. 7.26. Example of normal reconstructed sequences (test set).	265
Fig. 7.27. Example of reconstructed sequences with collective anomalies.....	266
Fig. 7.28. Example of reconstructed sequences with degradation patterns.	267
Fig. 7.29. Image of a normal sequence.....	268

Fig. 7.30. Images with collective anomalies.	269
Fig. 7.31. Images with degradation patterns.....	270
Fig. 7.32. Examples of condition indicator prediction.....	273
Fig. C.1. Co-occurrence of keywords.	353
Fig. C.2. Frequencies of the most implemented algorithms throughout the years (2010-2020) based on the Keyword Plus indicator.	355
Fig. C.3. Studies classified by industrial sector.....	359
Fig. D.1.1. Graphical representation of the Cooling Fresh Water (CFW) inlet pressure parameter.....	380
Fig. D.1. 2. Identification of the operational states for the monitored parameter. .	380
Fig. D.1.3. Example of normal sequences.	382
Fig. D.1.4. Example of sequences with collective anomalies.....	383
Fig. D.1.5. Example of sequences with degradation patterns.....	384
Fig. D.1.6. Histogram of the reconstructed errors of the normal sequences (test set).	385
Fig. D.1.7. Histogram of the reconstructed errors of the sequences with collective anomalies.	385
Fig. D.1. 8. Histogram of the reconstructed errors of the sequences with degradation patterns.	386
Fig. D.1.9. Example of normal reconstructed sequences (test set).	387
Fig. D.1.10. Example of reconstructed sequences with collective anomalies.....	388
Fig. D.1.11. Example of reconstructed sequences with degradation patterns.	390
Fig. D.1.12. Images with collective anomalies.....	391
Fig. D.1.13. Images with degradation patterns.....	392
Fig. D.1.14. Examples of condition indicator.....	393

Fig. D.2.1. Graphical representation of the exhaust gas outlet temperature parameter.	398
Fig. D.2.2. Identification of the operational states for the monitored parameter....	399
Fig. D.2.3. Example of normal sequences.	400
Fig. D.2.4. Example of sequences with collective anomalies.....	401
Fig. D.2.5. Example of sequences with degradation patterns.....	403
Fig. D.2.6. Histogram of the reconstructed errors of the normal sequences (test set).	403
Fig. D.2.7. Histogram of the reconstructed errors of the sequences with collective anomalies.	404
Fig. D.2.8. Histogram of the reconstructed errors of the sequences with degradation patterns.	404
Fig. D.2.9. Example of normal reconstructed sequences (test set).	405
Fig. D.2.10. Example of reconstructed sequences with collective anomalies.	407
Fig. D.2.11. Example of reconstructed sequences with degradation patterns.	408
Fig. D.2.12. Images with collective anomalies.....	409
Fig. D.2.13. Images with degradation patterns.....	410
Fig. D.2.14. Examples of condition indicator.....	412
Fig. D.3.1. Graphical representation of the exhaust gas outlet temperature parameter.	417
Fig. D.3.2. Identification of the operational states for the monitored parameter....	418
Fig. D.3.3. Example of normal sequences.	419
Fig. D.3.4. Example of sequences with collective anomalies.....	420
Fig. D.3.5. Example of sequences with degradation patterns.....	422
Fig. D.3.6. Histogram of the reconstructed errors of the normal sequences (test set).	422
Fig. D.3.7. Histogram of the reconstructed errors of the sequences with collective anomalies.	423

Fig. D.3.8. Histogram of the reconstructed errors of the sequences with degradation patterns.	423
Fig. D.3.9. Example of normal reconstructed sequences (test set).	424
Fig. D.3.10. Example of reconstructed sequences with collective anomalies.	426
Fig. D.3.11. Example of reconstructed sequences with degradation patterns.	427
Fig. D.3.12. Images with collective anomalies.	428
Fig. D.3.13. Images with degradation patterns.	429
Fig. D.3.14. Examples of condition indicator.	431
Fig. D.4.1. Graphical representation of the exhaust gas outlet temperature parameter.	436
Fig. D.4.2. Identification of the operational states for the monitored parameter.	437
Fig. D.4.3. Example of normal sequences.	438
Fig. D.4.4. Example of sequences with collective anomalies.	439
Fig. D.4.5. Example of sequences with degradation patterns.	440
Fig. D.4.6. Histogram of the reconstructed errors of the normal sequences (test set).	441
Fig. D.4.7 Histogram of the reconstructed errors of the sequences with collective anomalies.	441
Fig. D.4.8. Histogram of the reconstructed errors of the sequences with degradation patterns.	442
Fig. D.4.9. Example of normal reconstructed sequences (test set).	443
Fig. D.4.10. Example of reconstructed sequences with collective anomalies.	444
Fig. D.4.11. Example of reconstructed sequences with degradation patterns.	445
Fig. D.4.12. Images with collective anomalies.	447
Fig. D.4.13. Images with degradation patterns.	447
Fig. D.4.14. Examples of condition indicator.	449

Fig. D.5.1. Graphical representation of the exhaust gas outlet temperature parameter.	454
Fig. D.5.2. Identification of the operational states for the monitored parameter....	455
Fig. D.5.3. Example of normal sequences.	456
Fig. D.5.4. Example of sequences with collective anomalies.....	457
Fig. D.5.5. Example of sequences with degradation patterns.....	459
Fig. D.5.6. Histogram of the reconstructed errors of the normal sequences (test set).	459
Fig. D.5.7. Histogram of the reconstructed errors of the sequences with collective anomalies.	460
Fig. D.5.8. Histogram of the reconstructed errors of the sequences with degradation patterns.	460
Fig. D.5.9. Example of normal reconstructed sequences (test set).	461
Fig. D.5.10. Example of reconstructed sequences with collective anomalies.....	462
Fig. D.5.11. Example of reconstructed sequences with degradation patterns.	463
Fig. D.5.12. Images with collective anomalies.....	465
Fig. D.5.13. Images with degradation patterns.....	465
Fig. D.5.14. Examples of condition indicator.....	467
Fig. D.6.1. Graphical representation of the exhaust gas outlet temperature parameter	472
Fig. D.6.2. Identification of the operational states for the monitored parameter....	473
Fig. D.6.3. Example of normal sequences.	474
Fig. D.6.4. Example of sequences with collective anomalies.....	475
Fig. D.6.5. Example of sequences with degradation patterns.....	476
Fig. D.6.6. Histogram of the reconstructed errors of the normal sequences (test set).	477
Fig. D.6.7. Histogram of the reconstructed errors of the sequences with collective anomalies.	477

Fig. D.6.8. Histogram of the reconstructed errors of the sequences with degradation patterns.	478
Fig. D.6.9. Example of normal reconstructed sequences (test set).	479
Fig. D.6.10. Example of reconstructed sequences with collective anomalies.	480
Fig. D.6.11. Example of reconstructed sequences with degradation patterns.	481
Fig. D.6.12. Images with collective anomalies.	482
Fig. D.6.13. Images with degradation patterns.	483
Fig. D.6.14. Examples of condition indicator.	485
Fig. D.7.1. Graphical representation of the exhaust gas outlet temperature parameter.	490
Fig. D.7.2. Identification of the operational states for the monitored parameter.	491
Fig. D.7.3. Example of normal sequences.	492
Fig. D.7.4. Example of sequences with collective anomalies.	493
Fig. D.7.5. Example of sequences with degradation patterns.	494
Fig. D.7.6. Histogram of the reconstructed errors of the normal sequences (test set).	495
Fig. D.7.7. Histogram of the reconstructed errors of the sequences with collective anomalies.	495
Fig. D.7.8. Histogram of the reconstructed errors of the sequences with degradation patterns.	496
Fig. D.7.9. Example of normal reconstructed sequences (test set).	497
Fig. D.7.10. Example of reconstructed sequences with collective anomalies.	498
Fig. D.7.11. Example of reconstructed sequences with degradation patterns.	499
Fig. D.7.12. Images with collective anomalies.	501
Fig. D.7.13. Images with degradation patterns.	501
Fig. D.7.14. Examples of condition indicator.	503

Fig. D.8.1. Graphical representation of the exhaust gas outlet temperature parameter.	508
Fig. D.8.2. Identification of the operational states for the monitored parameter....	509
Fig. D.8.3. Example of normal sequences.	510
Fig. D.8.4. Example of sequences with collective anomalies.....	511
Fig. D.8.5. Example of sequences with degradation patterns.....	512
Fig. D.8.6. Histogram of the reconstructed errors of the normal sequences (test set).	513
Fig. D.8.7. Histogram of the reconstructed errors of the sequences with collective anomalies.	513
Fig. D.8.8. Histogram of the reconstructed errors of the sequences with degradation patterns.	514
Fig. D.8.9. Example of normal reconstructed sequences (test set).	515
Fig. D.8.10. Example of reconstructed sequences with collective anomalies.....	516
Fig. D.8.11. Example of reconstructed sequences with degradation patterns.	517
Fig. D.8.12. Images with collective anomalies.....	518
Fig. D.8.13. Images with degradation patterns.....	519
Fig. D.8.14. Examples of condition indicator.....	520

List of Tables

Table 2.1. Summary of the results obtained from the benchmarking implementation.	28
Table 2.2. Guidelines and services provided by regulatory bodies	29
Table 2.3. Summary of the implemented data pre-processing steps in identified studies related to the application of data-driven methodologies for the implementation of predictive maintenance of marine systems. The data pre-processing steps have been structured as follows: (1) Feature Selection, (2) Data Smoothing, (3) Outlier detection, (4) Normalisation, (5) Resampling, (6) Data Imputation, (7) Feature Extraction, and (8) Value correction.	39
Table 2.4. Summary of identified gaps divided by MA phase.....	75
Table 6.1. Main engine system monitored parameters.....	164
Table 6.2. Descriptive statistics of the monitored features.	169
Table 6.3. Correlation matrix of the monitored features.	172
Table 6.4. Imputation results of the main engine rotational speed parameter.	183
Table 6.5. Imputation results of the main engine power parameter.....	184
Table 6.6. Imputation results of the main engine fuel flow rate parameter.....	185
Table 6.7. Imputation results of the inlet pressure parameter of the lubrication oil system.....	186
Table 6.8. Imputation results of the inlet pressure parameter of the jacket cooling water system.....	186
Table 6.9. Imputation results of the turbine lubricating oil inlet pressure parameter of the turbocharger.	186
Table 6.10. Imputation results of the scavenging air pressure of the scavenge air receiver.	187

Table 6.11. Advantages and disadvantages of the implemented imputation techniques.	188
Table 6.12. Pearson’s correlation coefficient matrix.	193
Table 6.13. Large gaps of missing values imputation results.	197
Table 6.14. Parameters of the diesel generator considered for the case study.	198
Table 6.15. Descriptive statistics of the monitored features.	204
Table 6.16. Pearson’s correlation coefficient (absolute values).	205
Table 6.17. Imputation results of the LSTM-VAE-based regressor.	207
Table 6.18. Imputation results of the k -NN imputer.	207
Table 6.19. Imputation results of the application of Forward Fill and, subsequently, Backward Fill algorithms.	208
Table 6.20. Imputation results of the mean imputation technique.	208
Table 6.21. Descriptive statistics of the monitored parameters.	212
Table 6.22. Sequence selection for visual analysis.	215
Table 7.1. Descriptive statistics of the monitored parameters.	225
Table 7.2. Range of values analysed for hyperparameter optimisation.	227
Table 7.3. Descriptive statistics of the monitored parameter.	233
Table 7.4. Classification metrics results for performance evaluation of the multi-fault classification task.	245
Table 7.5. Descriptive statistics of the monitored parameter.	246
Table 7.6. Descriptive statistics of an observed subsequence and its respective simulated subsequences.	249
Table 7.7. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.	254
Table 7.8. Descriptive statistics of the monitored parameter.	257
Table B.1. Literature review summary about data imputation techniques.	334

Table B.2. Advantages and limitations of the methodologies reviewed in the maritime industry context.....	339
Table B.3. Summary of fault identification of marine systems studies.	341
Table B.4. Summary of time series imaging approaches in fault identification task studies.....	344
Table C.1. The 10 most cited articles in the anomaly detection for maintenance analytics practices field.	347
Table C.2. Most frequent keywords.	353
Table C. 3. Taxonomy of anomaly detection studies.	361
Table D.1. 1. Descriptive statistics of the monitored parameter.	380
Table D.1.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.	394
Table D.2.1. Descriptive statistics of the monitored parameter.	398
Table D.2.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.	413
Table D.3.1. Descriptive statistics of the monitored parameter.	417
Table D.3.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models	432
Table D.4.1. Descriptive statistics of the monitored parameter.	436
Table D.4.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models	450

Table D.5.1. Descriptive statistics of the monitored parameter.	454
Table D.5.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.	468
Table D.6.1. Descriptive statistics of the monitored parameter.	472
Table D.6.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.	486
Table D.7.1. Descriptive statistics of the monitored parameter.	490
Table D.7.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.	504
Table D.8.1. Descriptive statistics of the monitored parameter.	508
Table D.8.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.	521

Acronyms

1D-CNN	1-Dimensional Convolutional Neural Network
AAKR	Auto Associative Kernel Regression
AE	Auto-Encoder
AI	Artificial Intelligence
ANN	Artificial Neural Network
APE	Absolute Percent Error
AR	Autoregressive
ARIMA	Autoregressive Integrated Moving Average
BP	Back Propagation
bPCA	Bayesian Principal Component Analysis
BR	Binary Relevance
CBM	Condition-Based Maintenance
CC	Classifier Chains
CDR	Correct Diagnosis Ratio
CFW	Cooling Fresh Water
CM	Condition Monitoring
CNN	Convolutional Neural Network
DARNN	Deep Attention Residual Neural Network
DBN	Deep Belief Network
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DDFF	Double Dynamic Forgetting Factors
DG	Diesel Generator
DL	Deep Learning
DNN	Deep Neural Network
DOS-ELM	Denosing Online Sequential Extreme Learning Machine
DTA	Decision Tree Analysis

EAI	Explainable Artificial Intelligence
EB	Expected Behaviour
EM	Expectation-Maximisation
EWMA	Exponential Weighted Moving Average
FCL	Fully Connected Layer
FD	Fault Diagnosis
FF-BF	Fill and Backward Fill
FN	False Negative
FNN	Feed-forward Neural Network
FP	False Positive
FPGA	Field Programmable Gate Array
GA	Genetic Algorithm
GAF	Gramian Angular Field
GMMs	Gaussian Mixture Models
GRU	Gated Recurrent Unit
HCM	Health Condition Monitoring
HDK	Human Domain Knowledge
HI	Health Indicator
HP	Health Prognosis
HS	Health Stages
IF	Isolation Forest
IIoT	Industrial Internet of Things
IMO	International Maritime Organisation
IoS	Internet of Ships
IoT	Internet of Things
k -NN	k -Nearest Neighbors
KPCA	Kernel Principal Component Analysis
KSC	Kernel Spectral Clustering
LASSO	Least Absolute Shrinkage and Selection Operator

LDA	Linear Discriminant Analysis
LOCF	Last Observation Carried Forward
LS-SVMs	Least Squares Support Vector Machines
LSTM	Long Short-Term Memory
MA	Maintenance Analytics
MAPE	Mean Absolute Percentage Error
MARSS	Multivariate Auto-Regressive State Space
MCAR	Missing Completely at Random
MCC	Mattheus Correlation Coefficient
MCL	Context and Linear Mean
MDM	Maintenance Decision Making
MedAE	Median Absolute Error
MI	Multiple Imputation
MICE	Multiple Imputation by Chained Equations
ML	Machine Learning
MLE	Maximum Likelihood Estimation
MLKNN	Multi-Label k -Nearest Neighbor
MLP	Multi-Layer Perceptron
MLTSVM	Multi-Label Twin Support Vector Machine
MSE	Mean Squared Error
MSLE	Mean Squared Logarithmic Error
MTS	Multi-variable Time Series
NAR	Nonlinear Auto-Regressive
NN	Neural Network
NRMSD	Normalized Root Mean Square Difference
NRMSE	Normalised Root Mean Square Error
O&M	Operations & Maintenance
OCSVM	One-Class Support Vector Machine
OEMs	Original Equipment Manufacturers

OLS	Ordinary Least Squares
PCA	Principal Component Analysis
PDF	Probability Density Function
PdM	Predictive Maintenance
PEM	Prediction Error Minimization
PHM	Prognostics and Health Management
PLS	Partial Least Squares
PMF	Probabilistic Matrix Factorization
PMM	Predictive Mean Matching
PMS	Planned Maintenance System
PPCA	Probabilistic Principal Component Analysis
PSO	Particle Swarm Optimisation
PvM	Preventive Maintenance
R^2	Coefficient of Determination
RBF	Radial Basis Function
RBFN	Radial Basis Function Network
RM	Reactive Maintenance
RMSE	Root Mean Square Error (RMSE)
RNN	Recurrent Neural Network
RUL	Remaining Useful Life
RVM	Relevance Vector Machine
SA	Sensitivity Analysis
SaaS	Software as a Service
SD	Standard Deviation
SGD	Stochastic Gradient Descent
SHM	Structural Health Monitoring
SMPAE	Symmetric Mean Absolute Percentage Error
SPRT	Sequential Probability Ratio Test
SRU	Simple Recurrent Unit

STL	Seasonal and Trend decomposition using Loess
SVM	Support Vector Machine
TC	Turbocharger
TC	Total Citations
	Temporal Convolutional Network iwth Residual Self-
TCN-RSA	Attention Mechanism
TN	True Negative
TP	True Positive
USS	Updating Selection Strategy
VAE	Variational Autoencoder
VAR	Vector Autoregression
VAR-IM	Vector Autoregressive Imputation Method
VIF	Variance Inflation Factor

Abstract

Prognostics and Health Management (PHM) approaches are gaining popularity within recent years due to the growing need for enhancements in ship automation and intelligence. Although PHM technologies have been widely investigated and achieved a certain level of maturity in industries such as aerospace, manufacturing, and railway, it is an undeniable fact that the shipping sector is still in its infancy in this regard and further research is required on the matter. For this reason, the main aim of this thesis is to enable Smart Maintenance within the shipping sector. Accordingly, this thesis presents a Maintenance Analytics (MA) framework for marine systems. This framework is primarily constituted by three modules: 1) data pre-processing, which ensures the data quality and integrity required in the subsequent modules, 2) diagnostic analytics, which determines the current state of marine systems, and 3) predictive analytics, which aims to predict the Remaining Useful Life (RUL), thus establishing the future health state of marine systems. In total, eight novelties have been ascertained to contribute towards the analysis and formalisation of Machine Learning and Deep Learning approaches to ensure the applicability of Artificial Intelligence within the shipping sector, thus facilitating implementation of better maintenance strategies. To analyse the performance of such a novel MA framework, a total of eight case studies of distinct marine systems are introduced. Results demonstrate the importance of exploring, analysing, and formalising novel holistic MA frameworks to assist with decision-making processes related to maintenance strategies

to guarantee the robustness and flexibility of O&M activities whilst facilitating both reliability and availability of marine systems, thus reducing downtime and operational cost and enhancing ship/company profitability. Through the implementation of the eight distinct case studies, the developed MA has demonstrated its high accuracy in detecting and identifying faults for determining the diagnosis and in predicting the RUL for the prognosis of marine machinery. For instance, the fault detection phase and fault identification phase of the diagnostic analytics module have achieved an average accuracy of 92.5% and 95%, respectively.

Keywords: prognostics and health management; maintenance analytics; diagnostic analytics; predictive analytics; remaining useful life; marine systems; shipping industry; artificial intelligence.

Chapter 1

Introduction

1.1. Chapter Overview

This chapter introduces the background information for the formation of the thesis in tandem with the presentation of the research direction and the thesis layout.

1.2. Maritime Industry and Digitalisation

The maritime industry has often been referred to as a sector slow to change unless regulations require the adoption of new technologies or there is a clear short-term financial benefit. Legacy issues, skills gaps, and market instability are major factors that have made most businesses resistant to moving towards digital transformation. According to a study performed by Gkerekos et al. (2019), which surveyed various maritime stakeholders (e.g., technology providers, industry, funding bodies, and academics) as part of the IN 4.0 project, there was unanimous agreement that the UK maritime industry needs modernisation to remain competitive in a sector that occupies a central position in the growth of many countries' economies. One of the main barriers identified in adopting technologies such as Internet of Things (IoT), cloud computing, blockchain, and Artificial Intelligence (AI) was the lack of knowledge and understanding around them, thus being unaware of the potential benefits that such technologies could contribute to the profitability of an organisation. Such a barrier

could lead the shipping sector into a competitive degradation in technology and innovation when compared with other industrial sectors, including aerospace, manufacturing, and railway, all of which have proven their willingness to lead by example.

However, an unfortunate and unprecedented situation changed the perspective of the maritime industry and digitalisation, leading to organisations to decide to helm their business strategies and progress technologically. Despite the undoubtedly devastating impact that COVID-19 is having in most areas, a new paradigm is revolutionising the maritime industry. According to Thetius-Inmarsat (2021), COVID-19 has accelerated the process of adopting digital processes due to the increase in the average daily data consumption per vessel. This indicates a significant increase in IT infrastructure investment.

Such a fact facilitated an inflection point with regards to the perception of digitalisation within the shipping sector. According to the Lloyd's List 2021 Shipping Outlook Forum, Big data & AI has been considered as the most significant technology, aside from low/zero-carbon R&D, that will drive the change in shipping over the next five years (see Fig. 1.1). Moreover, according to the same survey, digitalisation has been considered as the best investment opportunity for shipping in 2021 (see Fig. 1.2) (Lloyd's List, 2021).

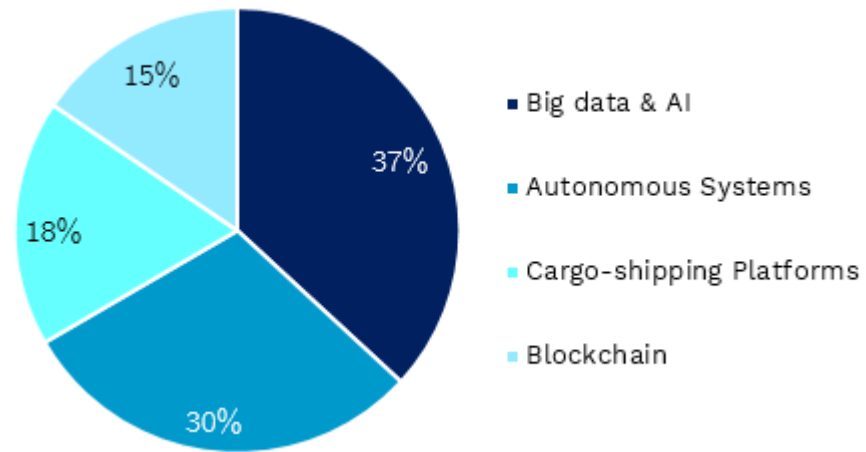


Fig. 1.1. Most significant technologies (aside from low/zero-carbon R&D) that will drive change in the shipping industry over the next five years (adapted from Lloyd's List, 2021).

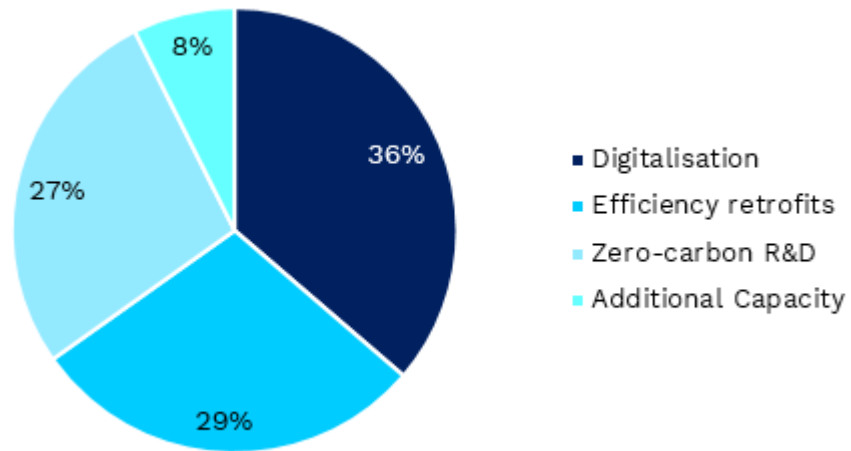


Fig. 1.2. Best investment opportunity for shipping in 2021 (adapted from Lloyd's List, 2021).

Unfortunately, this same headway could not be observed in a reduction of accidents (e.g., marine casualties and incidents). Although the first lockdown resulted in a

significant reduction in reportable accidents from March through to May of 2020, due to the decrease in maritime activity, the total number of reportable accidents for the year returned to normal levels (a total of 1,217 marine casualties and incidents were reported) due to a spike in reports of leisure craft accidents over the period June-September according to the 2020 Marine Accident Investigation Branch report. If such accidents are further analysed by considering their nature, it can be perceived that machinery is one the most frequent type of accidents. As per example, if UK merchant vessels (≥ 100 gt) are considered, it can be perceived that machinery is the second most frequent nature of accident, representing a 24% of the total (see Fig. 1.3) (MAIB, 2020).

If such accidents are also analysed by their respective costs, as introduced by the consortium composed of London Economics and NLA International in the Consultancy Research into the UK Maritime Technology Sector study, the total cost of maritime accidents between January 2012 and September 2019 was estimated to be £7.2 billion, £5.1 billion of which could have been prevented by considering automated mooring systems, autonomous vessels, on-board technology, and automated cargo handling. Technology that could avert the 72% of the total cost of maritime accidents over the analysed period (London Economics et al., 2021).

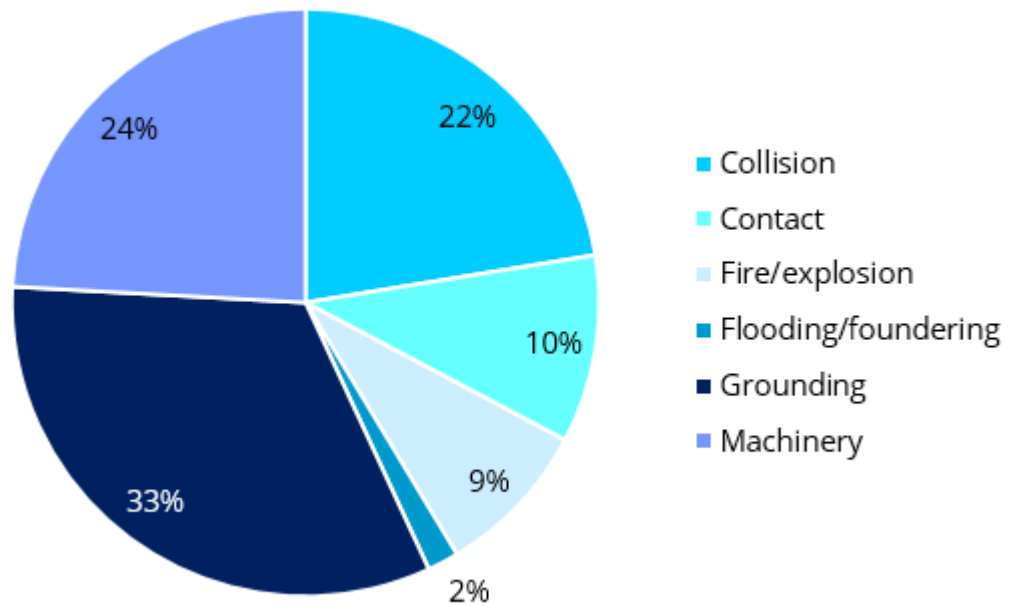


Fig. 1.3. Merchant vessels in accidents by nature of accident when considering UK merchant vessels ($\geq 100\text{gt}$) throughout 2020 (adapted from MAIB, 2020).

For this reason, there is an undeniable need to continue investing in technology within the maritime sector. The increasing level of information gathering and the improvement of communication technologies on ships by utilising sensors and AI can enable better coordination between ships and thus enhance the decision-making processes. An aspect that is fundamental in a sector in which 75-96% of accidents are attributed to human action due to fatigue or bad judgement (AGCS, 2018).

Accordingly, it is expected that there will be an increase in research related to the four main groups that define the smart shipping industry (smart port, autonomous

vessels, on-board technologies, and professional service technologies). Smart ports have already been an extensive research area that relies on automation, big data, AI software systems, and alternative energy. The port of Valencia is an example in which a method was employed to dynamically track the port's lighting system (Wang et al., 2021b). By contrast, while significant advancements have since been perceived, autonomous ships are not yet as well established as smart port technologies. Technology development as well as both the administrative and safety requirements that are needed for testing are examples of challenges that hinder the advancements of this smart technology. In spite of this, IMO level 3/4 autonomous ships are expected to be completed in the next 5 or more years (London Economics et al., 2021). Furthermore, numerous efforts by academia have been performed in the analysis of such technologies. An example of which is the project AUTOSHIP, which aims at speeding-up the transition towards a next generation of autonomous ships in the EU (AUTOSHIP, 2022). Examples of construction projects of autonomous ships are the Yara Birkeland and the Autonomous Spaceport Drone Ship (ASDS); projects that rely on Industry 4.0 technologies (Ichimura et al., 2022). Another critical area for enabling smart shipping is on-board technologies. These technologies assist in safe navigation, ship performance, maintenance, connectivity, and alternative propulsion through the implementation of AI and vessel optimisation systems. Examples of commercial technologies are Danaos Web Enterprise Suite (Danaos, 2022), Cassandra (DeepSea, 2022), Laros (Prisma Electronics, 2022), METIS ship connect (METIS, 2022), Mimic intelligent condition monitoring solution (James Fisher and Sons plc, 2022), and

Foresight Marine (European Space Agency, 2022). To the best of the author's knowledge, there is no real trend in the direction of technology development. However, certain efforts have been perceived in academia and some SMEs. Professional service technologies define the last group that characterises the smart shipping industry. Because of the emergence of new technologies and applications, a high demand for qualified operations and personnel has been perceived, which is expected to expand further in upcoming years.

Therefore, manufacturing and deployment of autonomous vessels; smart shipping sensor development and sensor integration services; smart shipping command and control systems and expertise; smart shipping data and intelligence services; smart shipping cyber security and risk management; and training in the adoption and utilisation of smart shipping technologies are considered the smart shipping technologies that the UK will opt for as a way of strengthening its position in market intelligence within the maritime industry in subsequent years (London Economics et al., 2021). By applying these technologies, it is expected that the UK shipping technology sector will be worth £13 billion per year by 2030, a sector that is currently a £4 billion industry (Department for International Trade, 2019).

Of all possible smart technologies being analysed in this respect, special attention will be given to smart maintenance in this study, as further research in ship operations

related to repairing and maintenance are needed. Although potential applications of AI within the shipping sector have been identified to perform a more efficient and economical maintenance (Department for Transport, 2019), which can also present a positive impact on both the safety and security of the personnel, there is not yet a clear technological solution for such a matter. Also, there is a lack of analysis and formalisation of AI methodologies for enabling smart maintenance within the shipping industry in academia, as, for instance, a limited number of studies related to fault detection, fault identification, and remaining useful life were identified (please see *Chapter 2. Literature Review*) Furthermore, according to the Inmarsat/Lloyd's List Digitalisation Uncovered Report, the reduction of operational cost and the creation of operational efficiencies have been considered as the most critical drivers for adopting digital solutions (Lloyd's List, 2021).

In addition, if Scopus database is used as a means to identify academic studies in relation to smart maintenance within the shipping sector, only 18 publications have been encountered, whereas in other analogous sectors such as railway, aerospace, and manufacturing a total of 204, 323, and 1,012 publications have been identified respectively. These results have been obtained by retrieving the results of the following query: ("Smart maintenance" AND "sector (e.g., shipping, railway, aerospace, and manufacturing)").

Thus, there is not only an unclear technological solution for addressing the challenges pertaining to ship operation regarding repair and maintenance, but there is also a lack of analysis and formalisation of such concepts. For all these reasons, the exploration of smart maintenance within the shipping industry is the primary motivation for conducting this research.

1.3. Research Direction

Despite the current challenges the shipping industry is currently facing, there is a great potential to explore emerging and innovative technologies to assist the O&M activities situation. In this regard, further research in smart maintenance within the maritime sector needs to be performed for a more secure, sustainable, and cost-effective O&M implementation. To contribute towards the analysis and formalisation of methodologies that assist in the establishment of Smart Maintenance strategies within the shipping sector, a main direction of research is proposed. Accordingly, both the main research question and the aim and objectives are presented within this section as an introductory part of the plan presented throughout the following paragraphs.

1.3.1. Research Question

The research question that aims to be answered throughout this thesis is expressed hereunder.

How can a novel Maintenance Analytics (MA) framework be enabled to enhance and advance O&M activities in the maritime sector?

1.3.2. Aim and Objectives

By answering the research question, the main aim is achieved. This is the formalisation and analysis of data-driven methodologies for the diagnosis and prognosis of marine systems through the development of a holistic maintenance framework comprised of novel components that relate to the data pre-processing, and both diagnostic and predictive analytics phases in order to employ Smart Maintenance within the shipping sector.

Accordingly, the objectives are established. These are as follows:

1. The identification of the current gaps within the shipping sector with regards to data pre-processing, and both diagnostic and predictive analytics through the application of a critical literature review.
2. The development of a maintenance analytics framework fuelled by Machine Learning and Deep Learning (DL) algorithms to generate the novelties ascertained based on the gaps identified. Contributions to the analysis and

formalisation of data pre-processing, diagnostic analytics, and predictive analytics are expected.

3. The assessment of the performance of the developed methodology and the demonstration of its effectiveness with regards to the generation of the novelties based on the gaps identified through the implementation of a total of 8 case studies.
4. The introduction of future research guidelines to continue contributing to the establishment of Smart Maintenance in the shipping industry based on the discussion of the obtained results.

The interconnections between the four main objectives and the potential actions needed to achieve such objectives are summarised and graphically represented in Fig. 1.4. The first objective aims to identify the main gaps related to the implementation of data-driven methodologies for the diagnosis and prognosis of marine systems. This step is of paramount importance in order to determine potential opportunities and novelties that can be addressed within the sector. To identify them, an AS-IS analysis and a critical literature review are expected to be conducted. By determining these gaps and opportunities, the novelties can be ascertained. The generation of such novelties are expected to be achieved in objective two, which aims to develop a holistic MA framework through the application of novel AI, ML, and DL approaches. This framework is comprised of novel components related to data pre-processing, and both diagnostic and predictive analytics. The developed holistic MA framework needs to

comply with certain quality criteria. Therefore, the third objective is also expected to be conducted, which aims to assess the performance and demonstrate the effectiveness of the MA framework through the implementation of 8 case studies. To finalise, based on the discussion of the results, future research guidelines to continue contributing towards the employment of Smart Maintenance in the shipping industry are expected to be established. This refers to the fourth and final objective.

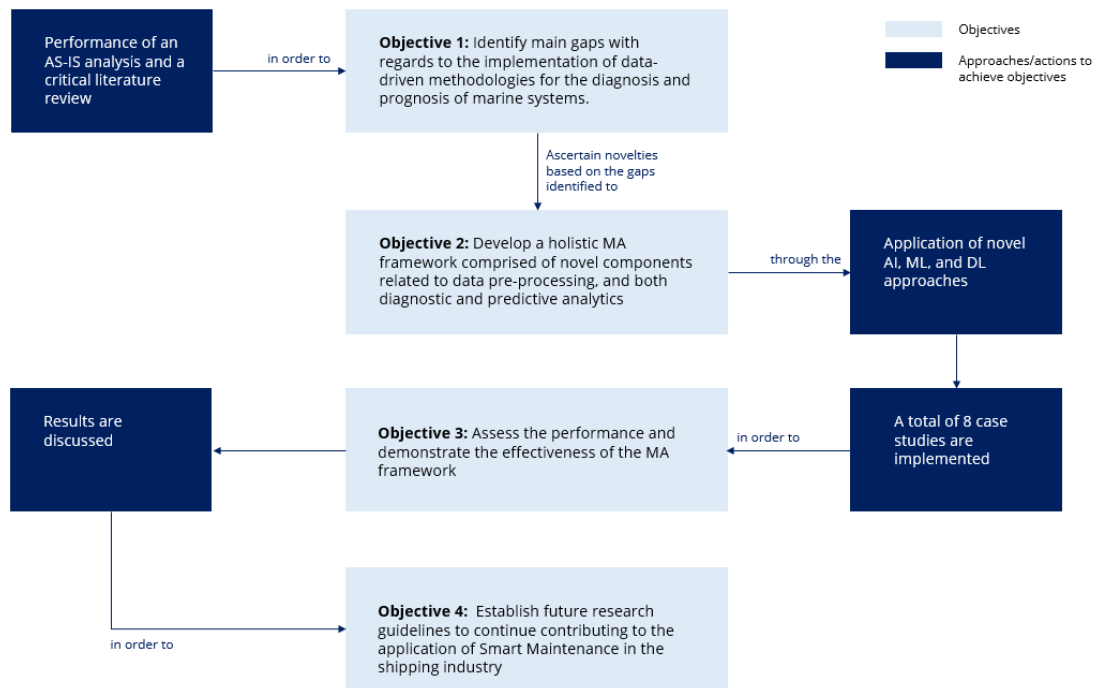


Fig. 1.4. Interconnections between objectives and approaches/actions to be taken.

1.4. Thesis Layout

This thesis has been structured in a total of 8 chapters. Brief descriptions of such chapters are presented hereunder.

Chapter 1. Introduction

This chapter refers to the introduction section and aims to provide the information that relates to the situation that the maritime industry is currently experiencing with regards to digitalisation. Through the provision of such information, a better understanding of the importance of the research presented in this thesis is expected.

Chapter 2. Literature Review

Chapter 2 aims to introduce the path from reactive to smart maintenance based on the literature review performed in order to provide a better understanding of the current practices within the sector and what to expect next. Additionally, chapter 2 includes the critical review performed to adequately identify the gaps presented in relation to the implementation of data-driven methodologies for enabling Smart Maintenance within the shipping industry. The structure of this literature review chapter is aligned with the definition of MA.

Chapter 3. Methodology: Part I. Introduced Novelties and Overview of the Developed Holistic MA Framework

Chapter 3 aims to introduce the novelties to be generated in this research and the overview of the Maintenance Analytics (MA) framework.

Chapter 4. Methodology: Part II. The Data Pre-processing Module

Chapter 4 provides a comprehensive description of the individual components that comprise the data pre-processing module.

Chapter 5. Methodology: Part III. The Diagnostic and Predictive Analytics Modules

Chapter 5 provides a comprehensive description of the individual components that comprise the diagnostic and predictive analytics modules.

Chapter 6. Case Studies and Results: Part I. The Data Pre-processing Module

Chapter 6 introduces case studies performed and a discussion of the main results to both validate and highlight the individual components of the data pre-processing module.

Chapter 7. Case Studies and Results: Part II. The Diagnostic and Predictive Analytics Modules

Chapter 7 presents cases studies performed and the discussion of the main results to validate and highlight the individual components of the diagnostic and predictive

analytics modules. Accordingly, the effectiveness of the MA framework and its anticipated functionality are expected to be demonstrated.

Chapter 8. Discussion and Conclusions

Chapter 8 introduces an in-depth summary based on the key learning outcomes and conclusions achieved throughout the development of the research. Accordingly, an analysis of the objectives' accomplishments, the shortcomings identified while developing and implementing the framework, and the future work guidelines to consider for continuing this line of research are presented.

1.5. Chapter Summary

A snapshot of the digitalisation being experienced within the maritime sector is presented in this chapter. Furthermore, the research question, and the aim and objectives are also established. The chapter is finalised by presenting the research output and the thesis layout.

Chapter 2

Literature Review

2.1. Chapter Overview

This chapter aims to introduce the critical literature review performed to identify the current gaps present within the sector so that the novel contributions of this PhD can be established. The structure of the literature review chapter aims to provide the path from reactive to smart maintenance to provide a better understanding of the current practices within the sector and what to expect next. Accordingly, the current maintenance practices within the shipping industry, and both the definition and applicability of Smart Maintenance are established.

Also, this chapter aims to introduce the critical literature review performed to identify the current gaps presented within the sector so that a novel contribution can be established. As perceived in Fig. 2.1., the structure of this literature review deepens the maintenance analytics concept and how this has been implemented within the shipping sector. To provide a clear and intuitive structure, the sections of this chapter are aligned with both the research direction and the definition of maintenance analytics. As defined in this chapter, a MA framework is usually constituted by four main modules: 1) descriptive, 2) diagnostic, 3) predictive, and 4) prescriptive. Specifically, the research will be mainly focus in both diagnostic and predictive

analytics, as the main aim is to enhance the current practices in relation to the determination of the current and future health of marine systems through the application of data-driven methodologies. Accordingly, both descriptive and prescriptive modules will not be further analysed throughout the thesis. As data-driven methodologies will be considered, special attention will be also given to the data pre-processing step to ensure data integrity and quality. For each of these phases, the most relevant papers identified are discussed to understand the current trends, and to determine their disadvantages and limitations that are yet to be addressed. By analysing such disadvantages and limitations, the main gaps are expected to be identified so that they can be addressed with the generation of novel innovative solutions. A graphical representation of the areas of investigation addressed in this literature review is presented in Fig. 2.1.

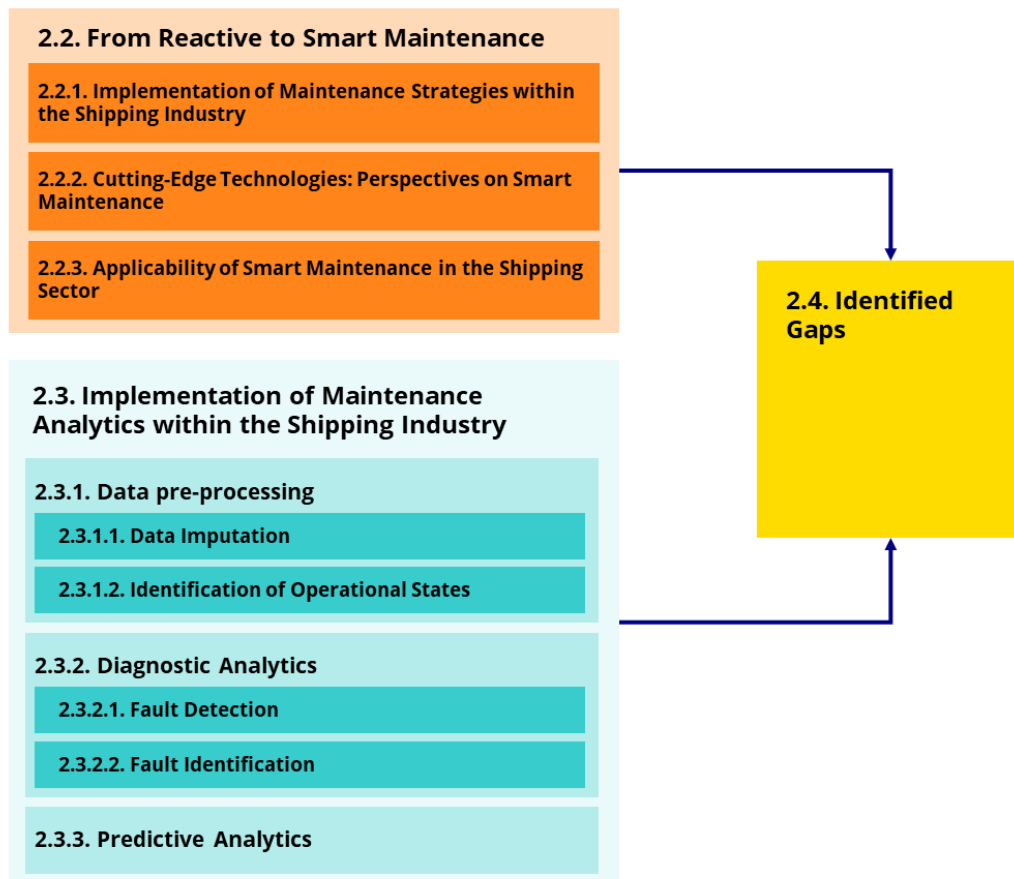


Fig. 2.1. Areas of investigation addressed in the literature review.

2.2. From Reactive to Smart Maintenance

2.2.1. Implementation of Maintenance Strategies within the Shipping Industry

The current maintenance routines on ships follow either a Reactive Maintenance (RM) or a Preventive Maintenance (PvM) approach (Han et al., 2021).

2.2.1.1. Reactive Maintenance (RM)

RM applies a run-to-failure approach. Accordingly, either repairing or replacing actions are performed when a failure in machinery occurs. This means that the machinery operate until a failure is presented so that the useful life of the asset can be exploited. The main advantage of such an approach is its simplicity in both planning and scheduling the maintenance actions, as well as both the personnel and spare parts required. However, not only devastating economic and security consequences can be expected due to the possible caused damage, but also the sub-optimal management of both personnel and assets, as these need to be available at any time, thus being inefficient and not cost-effective. Furthermore, the failure occurred can cascade in other systems and the selection of maintenance actions may be complex and inaccurate, as no information of the failure originated is available. For these preceding reasons, RM is usually only recommended for non-critical auxiliary systems and when the risk of failure is low (Cheliotis, 2020; Oikonomou, 2021).

2.2.1.2. Preventive Maintenance (PvM)

In tandem with RM, PvM is another widely used routine within the shipping industry. This routine aims to apply the maintenance actions at predetermined intervals. Such a strategy is also usually applied when a certain criterion is met (a threshold level is achieved, for instance). By applying this routine, the probability of failure of machinery is expected to be reduced. Therefore, in comparison with RM, PvM aims

to perform the maintenance actions before the machinery enters into a failure state (Jaramillo Jimenez et al. 2020).

Although the approach is considered as structured and organised, as well as aims the minimisation of failures while applying a more efficient asset management, P_vM maintenance can either over-maintain or under-maintain the considered machinery, can increase operational costs, and relies heavily on empirical knowledge (Cheliotis, 2020). Moreover, failures can still occur, as it has been identified that approximately a total of 77% of failures within marine systems occur in a stochastic manner (DNV GL, 2014). Other failures can also be introduced due to a bad decision or judgement of the personnel that performs the maintenance activities.

2.2.2. Cutting-Edge Technologies: Perspectives on Smart Maintenance

The infancy character that the shipping industry presents when considering emerging technologies and maintenance strategies facilitates the enablement of opportunities due to the need of exploring new concepts of maintenance based on Smart Maintenance, which aims to modernise the current maintenance activities. By considering the evolution of the shipping sector, PdM seems to be the next step. The increase in the utilisation of data within the sector also provides higher opportunities in relation to PdM to anticipate forthcoming failures in maritime machinery, as due

to the prosperity of Condition-Based Maintenance (CBM), the volume of data accessible to implement instant data-driven decision-making strategies for enhancing operations and maintenance activities is growing expeditiously. Accordingly, analysis can empower both the development and implementation of new PdM solutions.

CBM is a maintenance strategy based on the condition monitoring of assets in order to reduce the number of failures associated with machinery (Lazakis et al., 2016; Lazakis et al., 2018b; Raptodimos and Lazakis, 2018; Raptodimos and Lazakis, 2019), as the condition of the asset is considered to be one of the main drivers utilised for the determination whether a maintenance action is required or not (Jaramillo Jimenez et al. 2020). Accordingly, a large number of sensors are installed alongside the most critical components and around the environment where these assets are operating in order to implement Condition Monitoring (CM) effectively by utilising Industrial Internet of Things (IIoT). By monitoring the machinery, it is possible to detect at an early stage any possible failure that may occur, thus averting either the development of loss of functionality or any potential breakdown through the application of any required maintenance action in the instant that any abnormal behaviour has been detected. CM has proven to increase safety and reduce risk. Furthermore, CM could also increase the efficiency, reliability, profitability, and performance of the vessel, thus also facilitating the emissions reduction during its operational lifetime (Cheliotis et al., 2019, Lazakis et al., 2018a). In addition, if data collected by the monitoring systems is further analysed, conclusions with regards to the diagnosis and prognosis

of the asset can be performed in order to obtain a more proactive maintenance approach, thus enabling the implementation of predictive maintenance (PdM). Despite of all the advantages of CBM described above, this approach is highly costly, making its implementation only considered for critical assets only. It is probably due to this level of expenditure that only 2% of classed ships present a condition monitoring scheme (Jaramillo Jimenez et al. 2020).

PdM implements data analysis techniques to both identify and predict possible defects in machinery so that maintenance activities can be planned optimally prior to the occurrence of the failure. Multiple advantages have been identified from the implementation of PdM. Examples of such are the increased component operational life and availability, the pre-emptive corrective actions on non-critical items allowability, the decrease in equipment downtime and unexpected breakdown, and the reduction in unnecessary maintenance costs. However, its disadvantages cannot be diminished, as, for instance, initial costs of deployment can be expensive, additional skills and training are required for analysing the monitored data, and proper infrastructure is needed in order to apply a functional predictive strategy (Raptodimos, 2018b).

Various configurations of PdM have been proposed for the development of a holistic framework. For instance, Karim et al. (2016) and Jasiulewicz-Kaczmarek and Gola

(2019) introduced the concept of Maintenance Analytics (MA). MA is constituted by four interconnected time-line phases (maintenance descriptive analytics, maintenance diagnostic analytics, maintenance predictive analytics, and maintenance prescriptive analytics). MA aims to promote maintenance actions by enhancing the understanding of both data and information. Maintenance Descriptive Analytics is the first phase and aims to summarise the data collected from various maintenance sources. By summarising these data, measures and visualisation can be provided. The subsequent phase is Maintenance Diagnostic Analytics, which combines the outcome of the preceding phase with reliability data. This phase is comprised of three distinct sections: 1) fault detection, 2) fault isolation, and 3) fault identification. Fault detection aims to detect faults and malfunctions. Fault isolation implements root cause analysis. Fault identification provides the description of the fault type and its nature. By combining these phases, the current health of the marine machinery can be determined. By contrast, Maintenance Predictive Analytics aims to determine the future health of the machinery. Accordingly, the Remaining Useful Life (RUL) is predicted. RUL is predicted by considering current marine machine conditions concurrently with past operation profiles. Finally, the highest level of maturity is presented in the Maintenance Prescriptive Analytics phase, which transforms the outcomes obtained in the preceding phases into actions to optimise, among other aspects, Operations & Maintenance (O&M) management, personnel management, and asset management (see Fig. 2.2).

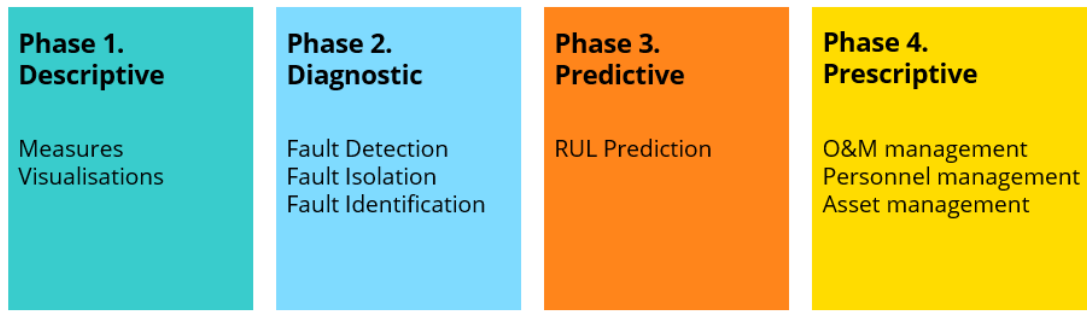


Fig. 2.2. Maintenance analytics summary.

An analogous concept widely investigated in aerospace and manufacturing is Prognostics and Health Management (PHM). According to Zhang et al. 2022, a viable PHM is constituted by four distinct modules: 1) Health Condition Monitoring (HCM), 2) Fault Diagnosis (FD), 3) Health Prognosis (HP), and 4) Maintenance Decision Making (MDM). The relationship between modules can be perceived in Fig. 2.3. HCM aims to monitor in real-time the analysed systems while obtaining ship condition parameters and performing condition assessment and anomaly detection. Accordingly, HCM can be divided into three distinct sections: 1) data acquisition, 2) data processing, and 3) condition monitoring. Data acquisition involves the process of collecting data on-board of navigation, system condition perception, and environmental data. This process is achieved through the application of intelligent and modern sensing technology. This allows the collection of vast amounts of ship data. Subsequently, to ensure the data's accuracy and integrity of the data collection to extract and deduce valuable and meaningful quality data for health management, data processing needs to be performed. The last section of HCM relates to condition

monitoring. This section applies a system of early warning, alarm, and anomaly detection according to online data. Accordingly, based on the abnormal issues detected by HCM, FD aims the identification of the failure modes and their causes so that a relationship between the monitoring data and the fault condition can be established. By considering the analysis performed in both HCM and FD modules, HP establishes the degradation model to complete the life prediction, and thus estimate the future behaviour of the system to realise the failure risk assessment and modify the control strategy accordingly. HP is usually comprised of Health Indicator (HI) construction, Health Stages (HS) division, and RUL prediction. HI is defined as either a statistical or quantitative indicator that is utilised to determine the health condition of the system. An adequate construction of HI facilitates a clear degradation and monotonic trend reflection throughout the indicator, aspect that is critical to effectively predict the RUL of the system. As usually the degradation trend is constituted by different changes in accordance with distinct fault severities throughout the life cycle, the health condition of the system can be categorised into different stages. Then, distinct RUL prediction methods are applied for each degradation category. According to the results obtained in the phases HCM, FD, and HP, MDM is implemented. MDM aims to describe the O&M vision of how the assets may need to be maintained throughout their respective lifespan. If PHM is compared with MA, it can be perceived that PHM is a more end-to-end approach, as not only considers the data analysis to establish decisions according to the current and future health of the identified system, but also aims to encompass the process of data acquisition and condition monitoring. By

contrast, as perceived in Fig. 2.2, MA focuses on the implementation of data analysis methodologies to drive conclusions that will lead the optimisation of the elements involved in ensuring the adequate functioning of the system. However, its challenges cannot be diminished, such as the increase of IT infrastructure, and cyber security issues.

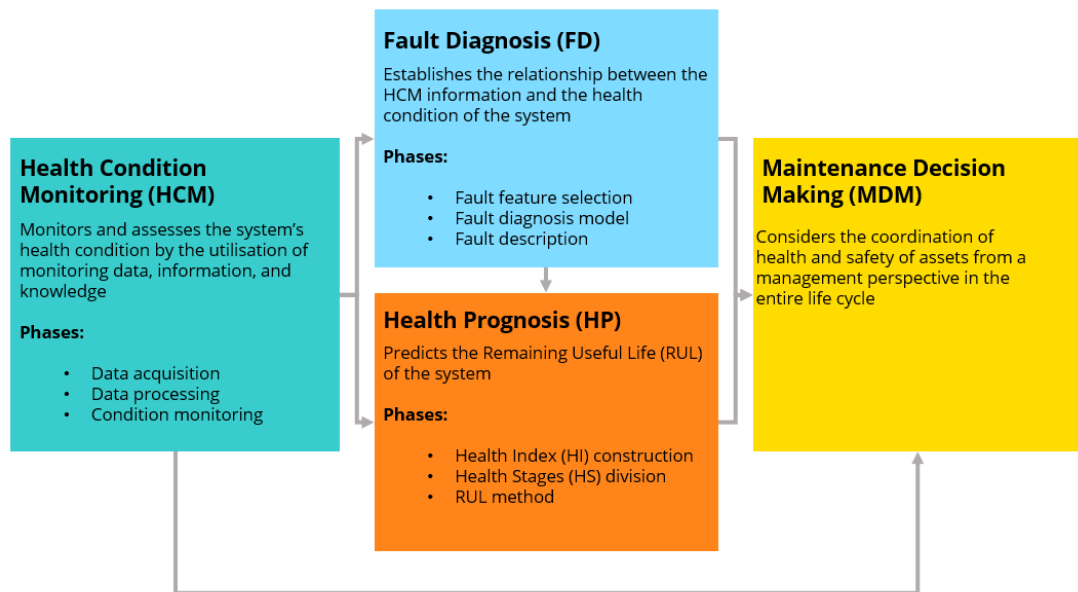


Fig. 2.3. Relationship and summary diagram of the modules that constitutes PHM.

2.2.3. Applicability of Smart Maintenance in the Shipping Sector

The modernisation of the current O&M activities industry through the application of Smart Maintenance is critical to address several challenges that the shipping industry is currently experiencing. Examples of which are ship breakdowns, marine systems sub-optimal performance, and ship-generated emissions (MAIB, 2019). Although it

can be perceived an increase in the performance of data acquisition through ship monitoring systems, the maritime sector is still based on Planned Maintenance System (PMS) guidelines (Jimenez et al., 2020).

After performing market research, it could be identified that several solutions aim to address the ship operational efficiency aspect by employing both data acquisition systems and analytics. Danaos Web Enterprise Suite, Cassandra, Laros, Metis ship connect, and Mimic intelligent condition monitoring solution are some of the major solutions currently identified. All of them presented interesting features, such as maintenance support, automated reporting, data acquisition as a service, and hardware provision (Sensor-as-a-Service). However, none of them can be considered a Smart Maintenance solution, as there is a lack of a comprehensive analysis of diagnosis and prognosis of the ship systems, element that is critical when considering these types of solutions. Table 2.1 presents a more comprehensive description of the features and gaps identified based on 1) the utilisation of real-time sensor data, 2) the implementation of data analytics, 3) the application of integrated frameworks, and 4) the consideration of artificial for diagnosis and prognosis. This benchmarking has been performed based on public information, as the author did not have access to the analysed systems.

Table 2.1. Summary of the results obtained from the benchmarking implementation.

Solution				Main features	Gaps
Danaos	Web	Enterprise	Suite	Systems and machinery definition, maintenance support, crew management, KPI's definition, maintenance planning, reporting and dashboards, and spare parts management.	AI is not implemented to provide diagnostic and predictive analytics. In an analogous manner, there is no evidence that data pre-processing is employed.
Cassandra				Utilisation of IoT, data collection, data pre-processing, data analytics, integration framework, diagnostics, systems and machinery definition, and reporting and dashboards.	There is no indication that novel data-driven methodologies are implemented for the application of data pre-processing steps, such as data imputation and identification of operational states. Analogously, it seems the software does not perform predictive analytics for the prediction of the RUL.
Laros				Automated reporting, regulatory compliance, hull and propeller monitoring, main engine condition monitoring, voyage parameter monitoring and optimization.	To the best of the author's knowledge, there is no evidence that Laros applies innovative solutions for the data pre-processing of data collected from sensors that are coupled to marine systems.
METIS	Ship		Connect	Data acquisition as a service.	The main purpose of this tool is to provide the data acquisition process as a service. Thus, maintenance analytics is not employed.

Mimic Intelligent Condition Monitoring Solution (https://www.james-fisher.com/services/inspection-and-monitoring/)	Vibration monitoring, performance monitoring, efficiency, manager, turbocharger monitoring.	fluid asset fleet There is no evidence that prognosis is performed by this solution. This also applies to the data pre-processing step.
---	---	--

When considering the applicability of maintenance practices within the shipping sector, it is of paramount importance the consideration of guidelines and regulatory bodies that set the minimum requirements and create the basic structure for the application of maintenance (Cheliotis et al., 2020). The main rules and applications for the application of CBM and predictive maintenance are summarised in Table 2.2.

Table 2.2. Guidelines and services provided by regulatory bodies

Regulatory body	Guidelines / Services
Lloyd’s Register	Condition monitoring of marine machinery
RINA	RINACube Applications
DNV-GL	DNVGL-SE-0439 Certification of condition monitoring DNVGL-CG-0052 Survey arrangement for machinery condition monitoring DNVGL-CG-0508 Smart vessel
American Bureau of Shipping (ABS)	ABS 0224: Guidance Notes on Equipment Condition Monitoring Techniques
Bureau Veritas	NR674 R00: Condition monitoring systems
Nippon Kaiji Kyokai	CBM Guidelines (Edition 2.0)
IACS	Z27 Condition Monitoring and Condition Based Maintenance

Based on the conclusions obtained from the benchmarking and the analysis of the shipping sector with regards to the implementation of smart technologies and their

implementation to enhance O&M activities, a SWOT analysis was performed. As strengths, it can be observed a significant transition from reactive to proactive approaches, as it has been identified that reactive strategies are unfeasible for critical equipment. Moreover, as PMS is mandatory according to the International Maritime Organisation (IMO), maintenance activities processes (planification, performance, and documentations) are standardised, thus complying with Classification Societies and manufacturers requirements. However, there are significant weaknesses that cannot be diminished. For instance, as until recently maintenance has not been considered of paramount importance within the maritime industry, there is a lack of implementation of CBM, thus precluding the application of more sophisticated and novel technologies. Furthermore, the shipowners may not have in-house competences or resources to implement such systems and they usually present a lack of interest and understanding of predictive maintenance methodologies within the maritime domain. If the personnel are also considered, it can be perceived they are usually overburdened with day-to-day operation routines, and thus their willingness to cooperate to ensure data quality and standards is minimal. Such identified weaknesses facilitate the practically inexistence of emerging and innovative technologies to assist O&M activities. Fortunately, the shipping industry is experiencing a digital transformation to maintain its competitiveness, transformation that has been accelerated due to the challenges presented by COVID-19. This opportunity enables an interest increase in the development and operations of autonomous ships, the development of digital twins, the utilisation of innovative inspection techniques (e.g., remotely operated vehicles),

the applicability of Internet of Ships (IoS), and the modernisation of modern maritime communications. Aspect that intrinsically enable the utilisation of real-time data to develop real-time intelligent systems. Nevertheless, although a certain advancement can be perceived, the maritime industry is still considered a conservative market with regards to analogous industrial sectors. Threat that may negatively impact the regularisation of innovative systems. Additionally, insurance claims due to the biased prediction of innovative systems that lead to potential incidents are yet to be further analysed. Cybersecurity is also a preeminent element that is starting to be considered one of the major concerns of stakeholders due to an increase in technology utilisation. If the personnel are again considered, it can be perceived that there is a lack of trained personnel within the shipping sector, implying a greater complexity in emerging technologies implementation. A summary of this SWOT analysis is presented in Fig. 2.4. A more comprehensive descriptive of such a SWOT is introduced in *Appendix A. SWOT Analysis of Maintenance Activities.*

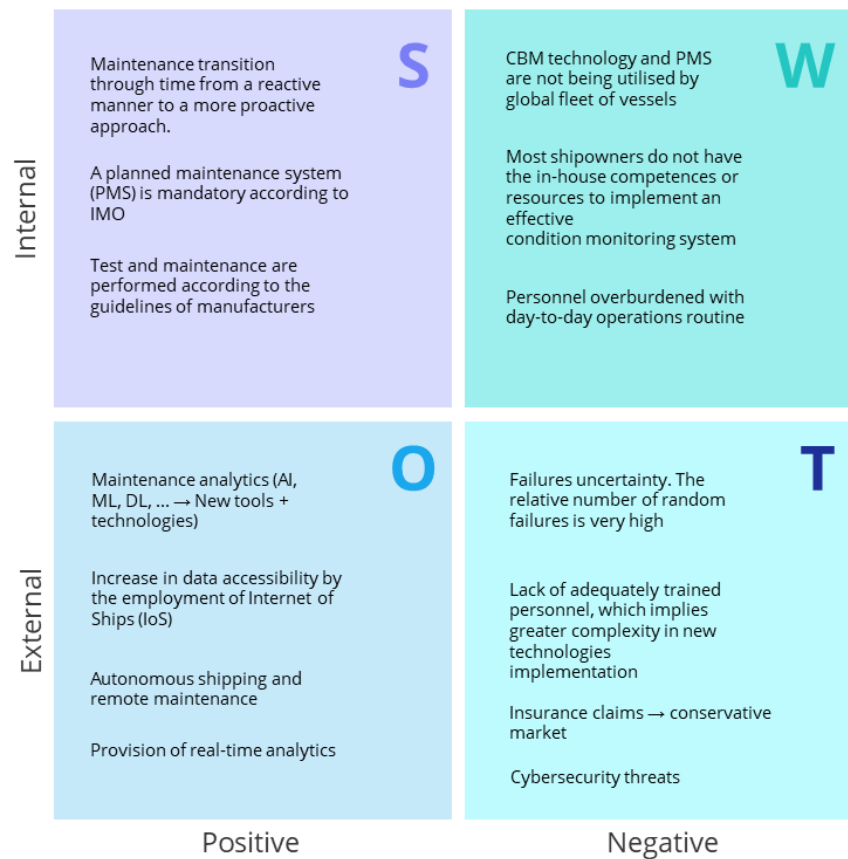


Fig. 2.4. Summary of the SWOT analysis of maintenance activities.

2.3. Implementation of Maintenance Analytics within the Shipping Industry

2.3.1. Data Pre-processing

Recent studies have highlighted the importance of utilising sensors, thus applying advanced monitoring techniques to provide both an enhancement of the vessel efficiency, and intrinsically reducing fuel oil consumption (Gkerekos et al., 2019b), but also reduce the number of failures associated with machinery. Condition Monitoring

(CM) has also proven to increase safety and reduce risk and, if adequately implemented, increment the efficiency, reliability, profitability, and performance of the vessel during its operational lifetime (Cheliotis et al., 2019; Lazakis et al., 2018a).

Sensors are installed alongside the most critical components and around the environment where these assets are operating to implement CM effectively by utilising IIoT. By employing IIoT, real-time data collection can be performed with the utilisation of smart sensors, reliable communications, and seamless integration to enable predictive maintenance through the provision of relevant information (Aheleroff et al., 2022). Therefore, diagnostic and predictive analytics can be performed to evaluate both the current and future state of marine machinery, and thus assist the decision-making processes to optimise, among other aspects, O&M activities, personnel management, and spare parts stocks.

Although the undeniable benefits that IIoT presents towards the enablement of advanced monitoring task that facilitate the performance of diagnosis and prognosis, several challenges need to be addressed. These include unreliable outcomes caused by certain anomalies and missing values that are originated by device/sensor failure, network collapse, and human error (Balakrishnan and Sangaiah, 2018; Izonin et al., 2019; Noor et al, 2014). Accordingly, data pre-processing needs to be performed. The quality of a data analysis is usually related with the quality of the data utilised. Thus,

an adequate data pre-processing performance is required to ensure the practical modelling in the real world (Dalheim and Steen, 2020). However, despite its importance, only two papers that presented a holistic data pre-processing framework for marine systems were identified. Dalheim and Steen (2020) presented a data pre-processing step comprised of a total of five phases: 1) feature selection, 2) time vector jumps and synchronisation, 3) outlier detection, 4) data validation, and 5) data extraction. Feature selection aims to identify the main parameters required for the adequate performance of the data analysis. The phase time vector jumps, and synchronisation is then implemented. Time vector jumps are defined as outliers in the first order differenced series of the time vector. Synchronisation is performed to ensure that all the selected signals are mapped to a joint time reference. The third step refers to outlier detection, which aims to detect the anomalous instances presented in the data. The authors divide this phase according to the characteristics of the outliers. The first group analyses obvious outliers, which can easily be detected through the implementation of physical constraints (e.g., set minimum and maximum thresholds). The second group of analysis refers to repeated values, which is only applied when continuous variables are considered. Repeated values usually refer to measurement that have been either rounded off to a few numbers of digits or pre-filtered. Drop-outs are the third type of outliers analysed, which usually appear as zero, a certain sensor dependent value, or NaN values. The fourth and last group of outliers analysed are the spikes, which are perceived as sudden changes, thus deviating from the rest of analogous points in a certain context. Once identified, the outliers are replaced

through the implementation of a simple linear interpolation, for instance. After this phase, the quality of the data needs to be evaluated. Accordingly, the fourth phase, data validation, is applied. To finalise, based on the type of analysis that needs to be performed, it is possible that data extraction is required. For instance, when analysing marine systems data, a general requirement is to sample the data under operational steady conditions.

The second study in relation to the development of a data pre-processing framework for marine systems was implemented by Masmoudi et al. (2021). The authors conducted a comparative study to analyse the effect of commonly utilised data cleaning, normalisation, and reduction techniques when performing ML predictions. Mean imputation, Expectation-Maximisation (EM) algorithm, min-max normalisation, and Principal Component Analysis (PCA) are just examples of analysed techniques in the study.

Some other studies that presented novel data-driven methodologies also introduced some analysis of data pre-processing methodologies. For instance, Cheliotis et al. 2020b implemented the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm to remove both outliers and transient states of operation. Michałowska et al. (2021) implemented resampling techniques, such as Lebesgue

sampling, to emulate an even sample rate and align data from distinct sensors. Data imputation was also applied by considering the last observation available.

Ellefsen et al. (2020) gave special attention to feature selection. Approaches such as Pearson correlation analysis and Human Domain Knowledge (HDK) were introduced. Analogous studies were performed in such a matter by Ellefsen et al. (2019) and Cheng et al. (2019). Ellefsen et al. (2019) studied the implication of feature selection in the model accuracy by analysing three distinct approaches: 1) all input features are considered for training the model, 2) HDK, and 3) Sensitivity Analysis (SA).

Cheng et al. (2019) also analysed correlation analysis and SA-based feature selection techniques. Moreover, Cheng et al. (2019) introduced moving average as part of data smoothing to reduce noise. Although the authors mentioned that both outlier detection and data imputation were performed, there is no clear indication of the methodologies they performed. An analogous practise can be perceived in Makridis et al. (2020), in which both Forward Fill and Backward Fill were implemented as data imputation techniques, although no discussion about the selection of such models for performing data imputation was presented. That said, the data pre-processing performed by Makridis et al. (2020) is probably the most complete of the studies identified that implement data-driven methodologies for the performance of predictive maintenance of marine systems. The authors implemented outlier detection by

considering the Mean + 3 Standard Deviation (SD) approach. Rolling mean was also implemented as data smoothing for either reducing or eliminating short-term volatility in data. To finalise, scaling and normalisation techniques were applied, and Principal Component Analysis (PCA) was performed as part of feature extraction.

Other studies introduced at some extent information in relation to the pre-processing steps performed prior to the implementation of the novel method, although their contribution in such a matter is not significant. For instance, Tan et al. (2021) only introduced data normalisation, as the data they utilised to validate the proposed methodology was already pre-processed and presented in analogous studies. Tan et al. (2020) presented in a similar manner the preceding aspect. However, special attention to feature selection was also given.

Table 2.3 summarises the pre-processing steps analysed in the identified studies with regards to the implementation of data-driven methodologies for the implementation of predictive maintenance of marine systems. As perceived, data normalisation is the most implemented step due to the characteristics of the features analysed, which present distinct range of values, so that all of them can influence the model in an equal manner. Moreover, some deep learning methodologies require the features to present a certain range of values. Although feature selection is not mentioned in most of the analysed studies, it is used in all cases at some extent. The least analysed steps

but critical to ensure data quality are both data imputation and the identification of operational conditions. Although certain efforts have been performed in the implementation of data imputation techniques, the selection and implementation of such techniques have been neither justified nor analysed. In addition, in most of the cases data imputation have not been implemented due to the lack of real-world studies, as most of the case studies performed to validate the novel methodologies are based on simulated data or data that have been already pre-processed in a preceding study. However, if real-world conditions are considered, it can be highlighted the criticality of adequately implementing data imputation, as preceding analysis have identified that datasets collected from marine machinery systems usually contain from 4.4% to 26% missing values (Cheliotis et al., 2019). With regards to the identification of operational states, only one study was identified, even though usually the data collection process takes place over an extended period and, therefore, non-operational states, such as transient and idle states, are also recorded, which need to be adequately identified and discarded (Cheliotis et al., 2020b).

For these preceding reasons, the remaining paragraphs of this data pre-processing section aims to identify the current practices and potential gaps identified with regards to data imputation and the identification of operational states.

Table 2.3. Summary of the implemented data pre-processing steps in identified studies related to the application of data-driven methodologies for the implementation of predictive maintenance of marine systems. The data pre-processing steps have been structured as follows: (1) Feature Selection, (2) Data Smoothing, (3) Outlier detection, (4) Normalisation, (5) Resampling, (6) Data Imputation, (7) Feature Extraction, and (8) Value correction.

Reference	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Cheliotis et al. (2022)			•					•
Cheng et al. (2019)	•	•	•	•		•		
Ellefsen et al. (2019)	•			•				
Ellefsen et al. (2019)b				•				
Ellefsen et al. (2020)	•			•				
Han et al. (2021)				•				
Han et al. (2021)b				•				
Makridis et al. (2020)		•	•	•		•	•	•
Michałowska et al. (2021)			•		•	•		•
Tan et al. (2020)	•			•				
Tan et al. (2021)				•				
Tang et al. (2020)	•			•			•	
Raptodimos and Lazakis (2019)				•				

2.3.1.1. Data Imputation

Data imputation is considered a crucial step in sensor data pre-processing due to the need of dealing with incomplete data (Liu et al., 2020b; Bashir and Wei, 2018).

For instance, Azimi et al. (2019) determined that the implementation of modern techniques is of paramount importance when dealing with missing values, as traditional methods, such as mean imputation and deletion methods, may lead to bias in the estimated due to their lack of accuracy, and thus leading to a decrease of data quality. By contrast, the implementation of modern techniques, such as Multiple

Imputation (MI), leads to accurate results and considers statistical uncertainty by adding an error term in the regression equations and, therefore, increasing the quality of the source sensor data with incomplete values (Hegde et al., 2019).

Nevertheless, the study of data imputation methods within the shipping sector is still inconsistent, although there is a significant increase in the development of novel data-driven methodologies for enabling smart maintenance. In total, only one study that proposed a novel data imputation framework for dealing with missing values of sensor data collected from marine machinery was identified. Such a study relates to Cheliotis et al. (2019), which developed a hybrid imputation method combining k -NN and Multiple Imputation by Chained Equations (MICE) algorithms with first-principle knowledge. The proposed hybrid framework was compared with k -NN and MICE algorithms by estimating the Absolute Percent Error (APE), the Mean Absolute Percentage Error (MAPE), and the standard deviation of the error metrics. Accordingly, the three distinct methods were applied to time-series data collected from a total of 8 sensors coupled to the turbocharger and to the main engine of a chemical tanker. Results demonstrated that the proposed hybrid imputation model outperformed k -NN and MICE methods. Despite these results, there are several limitations that cannot be diminished. For example, the two algorithms that constitute the hybrid framework (k -NN and MICE) are multivariate imputation methods, and thus, if the predictors are not highly correlated with the response, the imputation may not be accurate. Also, it is possible that there are not available predictors for a specific

feature. With regards to the k -NN algorithm, the number of neighbours, k , needs to be estimated, which may lead to either underfitting or overfitting if k is not optimally selected. Accordingly, further studies in this respect need to be performed, as the current sensor data collected from marine machinery present a significant number of missing values, which may lead to under-utilisation of data that can yield biased results.

Such a scenario is completely different in other industrial sectors or application domains, in which more efforts have been made to both formalise and analyse data imputation techniques. For instance, Pratama et al. (2016) presented a review study of conventional imputation methods (ignoring, deletion, and mean/mode imputation) and of more modern imputation procedures (hot and cold deck imputation, and multiple imputation that included autoregressive models, genetic algorithm optimisation-based methods, Support Vector Machines (SVMs), interpolation, maximum likelihood, fuzzy-rough set, and similarity measurements imputation methods). The study concluded that genetic algorithms, fuzzy c-means, and autoregressive methods were considered the best in terms of flexibility among the other imputation methods that were analysed.

Kim et al. (2022) presented a method for estimating missing values in ship principal data. Specifically, a regression model-based computation combined with the correction

using domain-knowledge was employed. A case study on principal data from 6278 container ships was conducted to highlight the proposed methodology. Results demonstrated that the proposed method presented an enhancement performance of 15.6% with regards to preceding methods.

Chong et al. (2016) presented a comparative study of five imputation methods (linear regression, weighted k -NN, SVM, mean imputation, and replacing incomplete values with zero). Time-series data collected from sensors installed in a floor of a community centre were utilised to evaluate the imputation performance of each analysed method. In addition, to study how the performance of the analysed techniques was affected by the ratio of missing values within the sample, four distinct datasets with different percentage of incomplete values were evaluated (5%, 10%, 15%, and 20%) by estimating the Normalized Root Mean Square Difference (NRMSD). Linear regression, k -NN, and SVM provided more accurate results than mean imputation or replacing incomplete values with zero. Moreover, it was also concluded that linear regression led better accuracy when a linear relationship between the outcome and the predictors was presented. Accordingly, the study suggested implementing SVM when the relationship between features was not linear.

Noor et al. (2014) introduced two data imputation methods (linear interpolation and mean imputation) to evaluate their imputation accuracy by assessing five randomly

simulated missing data patterns divided into three degrees of complexity (small percentage of incomplete values (5%), medium percentages of incomplete values (15% and 25%), and large percentage of incomplete values (40%)). To evaluate their performance three distinct metrics were considered: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and coefficient of determination (R^2). Results indicated that linear interpolation presents more accurate results, as mean imputation may distort the distribution of the feature and, therefore, the relationship between variables may result in a degradation of the performance.

Balakrishnan and Sangaiah (2018) presented an automated framework based on the Context and Linear Mean (MCL) method to impute missing values. To assess its performance, a temperature sensor was utilised. Results presented a minor performance enhancement when implementing the MCL model.

Azimi et al. (2019) proposed a MI approach by considering data variability, and thus considering context information, as the methods implemented were selected based on the characteristics of the data and the auxiliary information type. Accordingly, short-term historical data was utilised, as it is strongly correlated with the missing values when both the individual condition and the context situation are constant. Autoregressive models were considered to deal with short-term data. Both context data and lifestyle were also utilised. To evaluate the performance of the proposed

approach a case study was performed where 20 pregnant women were remotely monitored for seven months. Furthermore, a comparative study was conducted, in which four existing data imputation techniques (k -NN, Autoregressive, Maximum Likelihood Estimation (MLE) (Logistic), and SVM) were also analysed. RMSE and C-Index were the indices estimated to evaluate each of the models. Results demonstrated that the proposed methodology led to less accurate results than both autoregressive and k -NN models when the incomplete window was small. By contrast, it obtained the best results when the incomplete window was large. In addition, it was also concluded that the bias of the estimates was minimised when a high correlation existed between context and incomplete values.

Priya Stella Mary and Arockiam (2017) presented a methodology that assumed that data collected from sensors presented a high spatial and temporal correlation. Accordingly, sensors that were close geographically presented a strong relationship during a certain amount of time. The proposed methodology estimated n proximate sensors by using geographical coordinates through the Haversine formula. Then, the linear relationship between selected sensor data and the sensor data that requires imputation was estimated. If a strong correlation was identified, the incomplete values were imputed by utilising the correlated sensor occurrences at the corresponding time. The performance of this methodology was evaluated through the performance of a comparative study, in which a total of four distinct widely utilised imputation methods were considered (mean imputation, median imputation, mode imputation, and MICE).

The RMSE was estimated to quantify their accuracy. Results indicated that the proposed model led to higher accuracy only when the relationship between the sensors was strong. Fekade et al. (2018) introduced an analogous approach by presenting a probabilistic method to impute missing values from IoT devices by utilising data from similar sensors. The k -means clustering algorithm was considered to divide the sensors into distinct clusters. Once such clusters were formed, a Probabilistic Matrix Factorization (PMF) was implemented inside each partition to impute missing values. The proposed methodology was compared with two existing algorithms (Support Vector Machines (SVM) with linear and Radial Basis Function (RBF) kernels, and Deep Neural Networks (DNNs) with two and three hidden layers) by estimating RMSE. The proposed methodology was more efficient than SVM and DNN in terms of accuracy, as the existing methods were designed for classification purposes. Moreover, it was also perceived that the proposed method yields better results when the number of clusters was higher, as clusters with smaller groups would facilitate lesser differences between sensor measurements that belonged to the same group.

Bashir and Wei (2018) developed a new algorithm by considering a combination between Vector Autoregression (VAR) and Prediction Error Minimization (PEM) with an Expectation-Maximization (EM) algorithm. Such an algorithm was recognised as Vector Autoregressive Imputation Method (VAR-IM). 10% and 20% Missing Completely at Random (MCAR) data were created to assess the performance of VAR-IM alongside five other existing methods (listwise deletion, linear regression

imputation, Multivariate Auto-Regressive State Space (MARSS) model, and EM algorithm). In all cases, the implementation of VAR-IM led to the best results. However, it was perceived as a limitation that VAR-IM did not present more accurate results than any other analysed imputation techniques when the percentage of missing values was relatively small (approximately 5%).

Izonin et al. (2019) introduced another novel method based on the use of the Ito decomposition and the AdaBoost algorithm. MAPE, RSME, MAE, and Symmetric Mean Absolute Percentage Error (SMPAE) indicators were estimated to assess the performance of the established approach. The developed regressor yielded more accurate results than other analysed algorithms, such as SVR and Stochastic Gradient Descent (SGD) regressor. Nevertheless, a decrease in the imputation performance of the novel technique was perceived when either the data presented anomalies, or the sample size was not adequately large to train the model.

Liu et al. (2020b) established a univariate imputation method based on Seasonal and Trend decomposition using Loess (STL) to recover large gaps of missing data. The method tried to predict the missing values by implementing pattern discovery, thus decomposing the time series into trend, seasonal, and remainder components. Thus, the repeated patterns could be learned from the time series so that the imputation could be performed by combining the estimated components of each gap. As STL

decomposition required a complete dataset, the prior implementation of linear interpolation was required. The introduced approach, named Itr-MS-STLDecImp, outperformed the other analysed methods when dealing with large gaps, although its accuracy decreased when the time series did not present trend and seasonality.

Bokde et al. (2018) presented imputePSF, another univariate data imputation method that presented an adjustment of the Pattern Sequence based Forecasting (PSF) algorithm. Firstly, k -means algorithm was performed to cluster the time series into different partitions, and then the resulting clusters were utilised as data input for the PSF model. Such an approach led to more accurate results than the other analysed imputation methods, such as bayesian Principal Component Analysis (bPCA). Despite this fact, imputePSF was not recommended when the time series presented either noisy trends, or non-cyclical patterns.

Hegde et al. (2019) introduced a comparison of Probabilistic Principal Component Analysis (PPCA) and MICE. Accordingly, 116 dental variables, in which incomplete values were generated at random, were considered. PCA was utilised for dimensionality reduction so that a lower dimensional space of the dataset could be established. Such a property was accordingly utilised to impute the missing values, as these values were recovered from the compressed information distribution estimated by the PCA. Then, EM algorithm was applied to calculate the MLE of an incomplete

dataset in an iterative manner. By contrast, MICE imputed the missing values multiple times by using regression models, and thus contemplating the statistical uncertainty in the imputations. For this study, logistic regression was utilised for nominal categorical variables (2 levels), polytomous logistic regression for nominal categorical variables (>2 levels), and Predictive Mean Matching (PMM) for continuous variables. RMSE was estimated to assess the performances of both PPCA and MICE, which determined that PPCA led to more efficient results in comparison with MICE.

Hadeed et al. (2020) introduced an evaluation process to assess existing imputation methods. Such methods were sectioned into two groups: univariate imputation techniques (mean, median, Last Observation Carried Forward (LOCF), Kalman Filter, random, and Markov), and multivariate time series (PMM, and row mean method). The performance of the analysed methods was evaluated through the estimation of a total of five error metrics: 1) absolute bias, 2) percentage absolute error in means, 3) coefficient of determination, 4) RMSE, and 5) MAE. Accordingly, a total of 20 household with complete 24-h monitoring data for PM_{2.5} were considered. The results suggested the Markov technique as the most promising approach. Kalman filter also performed exceptionally well in data with strong trends. However, multivariate imputation techniques presented a lower performance due to the significant differences between households.

Chivers et al. (2020) established a two-step approach combining a binary classification step in tandem to regression analysis. Binary classification was considered to identify the unbalance samples of rain and no-rain, whereas regression analysis was implemented so that the magnitude of rain samples could be quantified. This analysis encompassed both the utilisation and comparison of commonly machine learning techniques, which included gradient boosting, bagged decision trees, neural networks, and SVM. Accordingly, a case study of a network of weather stations and a network of rain gauges in the UK was applied. The results concluded that the introduced technique outperformed a surface fitting technique for the recovery of missing precipitation data at 30-min resolution.

A summary of the analysed references mentioned within this section is expressed in Appendix B Table B.1, so that a brief description of the methodology implemented together with its utilisation and its limitations can be easily identified. As represented, various univariate and multivariate imputation methods have been either analysed or developed to lead to more accurate estimates of missing values. Therefore, it was highlighted the need of actively analysing this pre-processing step, as if missing values are not addressed appropriately, the resulting data analysis may be unreliable and inaccurate, thus yielding bias in succeeding steps, and, therefore, leading to poor models being implemented to assist decision-making processes.

If the studies analysed are divided by sector, it can be observed that fewer studies were introduced in the shipping industry in comparison to other sectors, such as environmental and healthcare sectors, for which a total of 4 and 5 studies have been analysed respectively. This indicates a lack of analysis and formalisation of data imputation frameworks within the shipping industry, although data-driven decision-making processes are increasingly popular within the sector. One of the current practices that is being implemented in the shipping sector for such a matter is the deletion of those instances in the dataset that contain missing values. Hence, those analysis that apply the deleting approach to deal with missing values may be biased, as most of the instances may be deleted for those datasets that contain many parameters, and thus result in a small dataset that led to poor data-driven models that assist decision-making processes.

Furthermore, in the preceding paragraphs it has also been observed that data imputation techniques usually are divided into two main groups: univariate and multivariate imputation techniques. Univariate approaches impute values of a parameter by only analysing its instances. By contrast, multivariate methods impute values of a parameter through the consideration of predictors, which are parameters that correlate with the response. The number of studies that contributed to the analysis of multivariate imputation techniques are nearly equal to the ones that

considered univariate imputation techniques, as of the total of 32 machine learning and time series forecasting models analysed, 53% referred to univariate imputation techniques and 47% to the multivariate imputation methods. However, only one third of the presented univariate imputation methods were considered as novel imputation algorithms. Therefore, most of the univariate imputation techniques considered were conventional imputation approaches, such as mean imputation or LOCF models. Although such conventional methods were easy to interpret and implement, they presented some major limitations that could not be diminished. Examples of which were the distortion of the parameter and the disruption of the relationship between the predictors and the response variable when implementing mean imputation and the statistical deficiencies of the LOCF model (Lachin, 2016).

As indicated in the preceding paragraphs, none of the studies so far considered the option of imputing incomplete values in real time to assist instant data-driven decision-making strategies. Furthermore, the applications of such algorithms were very specific, as they did not analyse whether the proposed methodology could work when dealing with different types of datasets or not. Although several studies have been performed to provide a formal approach for data imputation, only one approach has been suggested in the shipping industry to the best of the author's knowledge.

2.3.1.2. Identification of Operational States

Raw data collected from marine machinery usually contain non-operational states that adversely alter the results outlined when implementing data-driven tasks to determine the current and future health of marine machinery. Although the marine engines typically run under steady-state conditions, fluctuations may occur due to, for instance, environmental conditions or variations in the operating status (Theotokatos et al., 2020). Therefore, if such states are not adequately addressed, a decrease in both computational efficiency and model effectiveness can be perceived. However, despite its preeminent importance and the efforts made to enhance the current practices in relation to O&M activities within the shipping sector, the analysis of this pre-processing step is minimal, thus indicating that this step is not yet widely formalised.

Of a total of 6 identified publications that consider the steady states' identifications as a pre-processing phase, only 2 of them implemented data-driven models. Perera and Mo (2016) implemented Gaussian Mixture Models (GMMs) with an EM and PCA to both classify and analyse frequent operating regions of marine engines. Dalheim and Steen (2020b) introduced a new computationally efficient method to identify those parts of the time series that refer to steady states by assuming that the underlying system behaviour could be modelled by a deterministic linear trend model.

Although the results of such methodologies demonstrated promising results, the case studies that were implemented to analyse their performance only considered different engine loads, although there are states other than engine operating regions, such as idle states. Moreover, GMM with EM require the selection of its hyperparameters prior to the selection of the different states. Such a matter may be unfeasible when deploying this pre-processing step in real time, as new operating regions may arise. Accordingly, the hyperparameters that determine the number of clusters need to be updated, thus yielding an increase of the computational time. Furthermore, various studies are still considering non-data-driven models to address this matter (e.g., manually identifying the steps through the implementation of HDK), which is more time consuming due to the need for human resources; thus, increasing the probability for human error. For instance, Ellefsen et al. (2020) manually divided the engine loads into five distinct operating conditions to perform multi-regime normalisation. Nevertheless, the need for automating such a process was also indicated in this study, as new operating conditions may be encountered in real-life systems.

2.3.2. Diagnostic Analytics

As introduced precedingly and defined by Karim et al. (2016) and Jasiulewicz-Kaczmarek and Gola (2019), Diagnostic analytics is the second phase of a four interconnected time-line process, conceptualised as Maintenance Analytics (MA), and aims to determine the current health of marine systems. Accordingly, a total of three

phases are considered: 1) fault detection, 2) fault isolation, and 3) fault identification. Fault detection aims to encounter faults and malfunctions, whereas fault isolation implements root cause analysis and fault identification describes the fault type and its nature. Due to the identified potential enhancement that can be established within the shipping sector, fault detection and fault identification are comprehensively analysed in the subsequent paragraphs.

2.3.2.1. Fault Detection

Fault detection has evolved from anomaly detection, which aims to detect data patterns that deviate significantly from normal operation behaviour. Its implementation has been identified as being of paramount importance due to its extensive application domains (Erhan et al., 2021), in such areas as manufacturing (Ducharlet et al., 2020; Alaoui-Belghiti et al., 2019; Morariou et al. 2020), railway (Oliveira et al., 2019; Xue et al., 2019; Shi et al., 2019), and aerospace (Roy et al., 2018; Li et al., 2019; Imbassahy et al., 2020).

However, when the shipping sector is considered, only 7 studies related to data-driven methodologies for the application of anomaly detection of marine systems have been identified. Aslam et al. (2020) provided a comprehensive survey of the Internet of Ships (IoS) paradigm as well as its key elements and its main characteristics. The paper introduced a review of automatic fault detection methodologies in the shipping

industry and provided evidence of the need of applying automatic and intelligent method to both detect and report faults.

Lazakis et al. (2018) proposed a methodology for the monitoring and detection of operating anomalies in ship machinery based on a one-class SVM (OCSVM). The model was trained by using data that corresponded to the normal behaviour of a diesel generator under varying operating conditions. Abnormal data was simulated in the form of a sensitivity analysis. The proposed approach was effective for identifying anomalies. However, it does not consider the characteristics of time-series data, and the normal operating conditions may be considered anomalous if the training set does not contain similar conditions. The following may also need to be considered: it may lead to poor performance if the sample contains noise, and it is inconvenient if the time series is large.

Brandsæter et al. (2017) presented a cluster-based anomaly detection methodology. This was based on an original methodology that was divided into two main steps: signal reconstruction, through the implementation of Auto Associative Kernel Regression (AAKR), and residual analysis, by performing Sequential Probability Ratio Test (SPRT). The methodology was then modified to include two new steps: cluster analysis, through the utilisation of the k -means algorithm, and the selection of a set of closest points per cluster, which would replace the original dataset as training set

to reduce the computational cost. The proposed approach was assessed by analysing sensor signals on a marine diesel engine. Fault data were simulated to be implemented as the test set, as no fault data were available. The technique demonstrated to be successful in detecting anomalies and the computation time was reduced in relation to the original methodology. However, the reconstruction method does not consider the characteristics of time series data, such as its successiveness, and there is no evidence that this methodology can be used to estimate anomalies in real time, as, for instance, there is no indication on how to estimate the clusters and the dimensions of the hyperrectangles, which refers to different clusters, dynamically. The proposed methodology is expanded in Brandsæter et al. (2019), which introduced a comprehensive description of the generalisations and modifications performed in the original methodology. As mentioned, cluster analysis was applied to replace the original dataset with rectangular boxes that referred to different clusters. In addition, the distance measure was altered to treat the variables differently based on the credibility of the signal and to distinguish between explanatory and response signals. Credibility estimation was also performed.

Cheng et al. (2019b) implemented a denoising filter based on Field Programmable Gate Array (FPGA) to apply fault feature extraction in gearbox vibration signals that contain strong noise. Specifically, a 50-stage low-pass filter design was implemented and proved to denoise gear fault vibration signal to diagnose the gearbox fault. This is an interesting approach that can be used as an effective pre-processing step when

applying vibration signal diagnosis as a denoise step that can be implemented in real time. However, its application is specific to vibration analysis, and thus cannot be considered as a holistic anomaly detection framework.

Ellefsen et al. (2019) reviewed four well-established deep learning techniques applied in PHM systems: Deep Belief Network (DBN), Auto-Encoder (AE), LSTM, and Convolutional Neural Network (CNN). Also, some of the benefits and challenges to be faced in relation to PHM based on DL were introduced. In relation to the benefits, the authors suggested that the provision of high-speed broadband connections to ships at sea would enable online PHM systems based on DL, which could facilitate autoships without onboard maintenance personnel and achieve zero-downtime performance. Hence, it was thought that when PHM systems based on DL were introduced, they could contribute to reduce errors occurring due to personnel, as systems were less dependent on prior knowledge and human influence. However, there is still a lack of confidence and trust in “black-box” systems within the maritime industry and a lack of run-to-failure data of components and subcomponents. In addition, due to the implementation of real-time PHM systems, such systems would have to provide automatic pre-processing and dimensionality reduction schemes due to continuous flows of data. This may induce cyberattacks, and thus threaten safe maritime operations.

Ellefsen et al. (2020) also proposed a fault-type independent spectral anomaly detection algorithm for marine diesel engine degradation in autonomous ferries. The VAE was utilised as DNN, and thus trained on pre-processed normal operation data, the engine loads of which were merged into one context by applying a multi-regime normalization technique. Then, the trained VAE was used to estimate the velocity and the acceleration of the anomaly score at each time step in three fault types with different natures of degradation. Both the velocity and the acceleration were estimated dynamically to detect faults automatically when the estimations exceeded the threshold limits. The proposed methodology achieved an accuracy of 97.66% when the acceleration was used as the fault detector, proving that the algorithm could be used to detect degradations of different natures. However, the proposed methodology cannot be applied automatically in real time, as the engine loads were divided into distinct operating conditions manually. In addition, a hybrid power lab was utilised to collect the data sets, and thus the authors do not refer to some of the challenges that researchers need to deal with when using real data, such as the lack of synchronisation between sensors, the appearance of missing values, and the emergence of anomalous data in the training set. Analogously, an unsupervised reconstruction-based fault detection algorithm was also presented in Ellefsen et al. (2019), as supervised classifiers are highly complex to be implemented within the maritime industry due to the lack of fault labels. Hence, VAE was selected as a reconstruction model, which was implemented in two data sets of real-operational data from a marine diesel engine. The reconstruction error was used as an anomaly score function, which

may have led to inaccuracies due to the characteristics of sensor data of marine machinery. Thus, due to the different engine operational loads, feature selection was applied, and thus three scenarios were presented: all input features were included in the reconstruction, feature selection was implemented based on human knowledge, and feature selection was performed by applying SA. Findings demonstrated that the most accurate result was achieved when SA was being applied as feature selection, as irrelevant and redundant input features were not considered in the training process. The presented framework presents analogous advantages and disadvantages of the algorithm presented in Ellefsen et al. (2020), which suggests that the algorithm presented in Ellefsen et al. (2020) was adjusted to include some of the enhancements that were not contemplated in Ellefsen et al. (2019b), such as the consideration of several types of faults and the necessity to merge the different engine loads into one context.

Karatuğ and Arslanoğlu (2022) aimed to propose a condition-based maintenance model for ship machinery. Specifically, the study introduced an Artificial Neural Network (ANN) to determine the engine performance. The case study was based on the analysis of a 489-day dataset. The fault diagnosis system consisted of a total of six elements (measured real data, performance model output, constraint information, user interface, FMEA approach, and fault prediction).

Cheliotis et al. (2020b) combined Expected Behaviour (EB) models with the Exponential Weighted Moving Average (EWMA) for fault detection. Four different regression models were assessed: Ordinary Least Squares (OLS) single linear regression, multiple linear ridge regression, OLS single polynomial regression, and multiple polynomial ridge regression. Multiple polynomial ridge regression was identified as the most accurate to detect faults manifesting in both the main engine cylinder exhaust gas temperature and the main engine scavenging air pressure. As the collected data represented fault-free operating conditions, a total of four different fault cases were simulated in the form of a sensitivity analysis. The estimated residuals were analysed in an EWMA control chart that contained upper and lower control limits to detect faults. It was concluded that the proposed approach could successfully detect imminent faults by analysing the residuals from the recorded and expected occurrences. Data preparation was of paramount importance in this study due to the characteristics of the raw data and the models that were implemented. DBSCAN algorithm was applied effectively to remove outliers and transient states of operation, and thus induce its applicability when dealing with these types of data. However, there is no evidence that this framework can be implemented in real time. Also, although the authors were sceptical about "black-box" models, there is no doubt whatsoever about its accurate performance to detect anomalous data. A comprehensive comparative study including both type of models could be implemented to support the utilisation of "black-box" models, and thus reverse the conception of these approaches within the maritime industry.

A summary of the analysed references precedingly described within this section is shown in Appendix B Table B.2, in which a brief description of the methodologies is characterised together with the representation of their respective advantages and limitations. Although all the methodologies demonstrated their accuracy in the case studies implemented, there is no evidence that they can be applied in real time for diverse faults and marine machinery. In addition, most of the analysed frameworks do not consider the characteristics of time-series data, albeit the features considered in the case studies are indexed by time. Consequently, methodologies that apply distance-based approaches may yield inaccurate results, as anomalous data points in time-series data may be considered as normal data points since its numerical value is within normal operational thresholds.

Moreover, due to the importance and extensive research performed in this area, both a bibliometric analysis and a taxonomy of anomaly detection papers have also been performed. Results indicated that DL is the type of methodology that is being more widely applied. Autoencoders and LSTM networks are examples of the most common models implemented in the analysed studies. Clustering algorithms, such as k -means and GMM, were also widely applied within this context. A comprehensive description of the main results of both the bibliometric analysis and taxonomy are presented in *Appendix C. Bibliometric and Taxonomy results of anomaly detection studies.*

2.3.2.2. Fault Identification

Fault identification, which can be also referred to as fault classification, is the last stage of the diagnostic analytics phase. The main aim of fault identification is to provide a description of any considered fault type and its nature. Despite the importance of this phase and its criticality to define the maintenance strategy, it is still an unexplored area due to the lack of fault data within the maritime industry. Specifically, only a total of six studies that address the fault classification task for marine systems have been identified.

Wang et al. (2020) presented a fault diagnosis framework constituted by an unsupervised and supervised phase. The unsupervised phase was introduced through the application of the k -means algorithm, the main aim of which was to cluster data based on different operating conditions. The supervised phase implemented Back Propagation (BP) neural network to identify the running state of the system. PCA was also applied to optimise the fault diagnosis scheme. To validate the performance of the framework, a case study on a marine diesel engine was performed, as it is considered a critical system. This is because the working environment of diesel engines have been identified as decayed, thus presenting a high failure rate. Furthermore, due to its complexity, the implementation of maintenance is usually complicated and, therefore, its implementation requires a substantial amount of time. Hence, if maintenance is executed inadequately, serious accidents may occur, thus facilitating a

negative economical and environmental impact, and endangering the lives of seafarers. The data considered for the case study was running under five common working conditions (normal, carbon deposition of the injector nozzle, air leakage of the exhaust valve, wear of the high-pressure oil pump and damage of the piston ring). Also, a total of eight parameters were considered, such as mean effective pressure, scavenging pressure, and rotation speed. The established fault diagnosis scheme demonstrated high accuracy under both working and high-pressure oil pump wear exhaust valve leakage conditions, although the diagnostics of both the nozzle carbon deposition and piston ring damage conditions required an enhancement.

Cai et al. (2017) introduced another fault diagnosis framework for marine diesel engines. The first step was the structuring of the diesel engine system into subsystems to reduce the complexity of the analysis. Accordingly, the 1) fuel, 2) lubrication, 3) intake and exhaust, and 4) cooling systems were identified. Then, a classification model based on SVM was established to perform operating state monitoring and fault diagnosis. To finalise, the association rule mining algorithm was considered to analyse the relationships among the fault characteristics at distinct levels. A historical fault database was implemented for such a purpose. Results demonstrated that by performing a hierarchical analysis the fault diagnosis phase was simplified and an increase in the classification accuracy was perceived. For instance, an accuracy of 96% was obtained when the diesel engine fuel system fault dataset was considered. Such a

dataset presented a total of 75 groups of prediction data in 5 working conditions with 15 groups of prediction data of each working state.

Hou et al. (2020) also proposed a fault diagnosis framework for marine diesel engine. Specifically, the fuel oil supply system of the engine was considered. Analogous to Cai et al. 2017, the classification performance of the SVM model was analysed. Moreover, PCA was applied prior to the training of the model to reduce computational complexity. To optimise the performance of the SVM, a three-dimensional Arnold mapping introduced into the Particle Swarm Optimisation (PSO) algorithm was considered. Accordingly, an enhancement in the generalisation capability was expected. As a case study, a total of ten faults presented in the fuel oil supply system were considered. A comparative study between the established methodology and the multi-layer perceptron (MLP) with BP algorithm added momentum factor and adaptive learning, and Radial Basis Function Network (RBFN) was also introduced. An outperformance of the established approach when considering the Correct Diagnosis Ratio (CDR) was perceived, obtaining an average of 93.9%. However, regarding the execution time, it could be perceived a significant difference, as the proposed approach presented an execution time of 78.27 seconds, whereas the RBFN and MLP methods presented an execution time of 0.98 and 0.53 seconds respectively.

Senemmar and Zhang (2021) developed a new deep learning-based framework for fault detection, classification, and location identification simultaneously in shipboard power systems. A total of three distinct methodologies were introduced: 1) deep neural network, 2) gated recurrent unit, and 3) LSTM. To implement the case study fault data from an 8-bus shipboard power system were simulated. A 99% accuracy was obtained, determining the GRU-based model as the most effective DL model. The DNN model was the one that presented less accurate results. To consider potential real-world scenarios, the impact of load variation and noisy inputs of the model performance was analysed. It was perceived that the decrease in the performance of the framework when such elements were introduced were minimal, thus determining the robustness of the proposed approach.

Tan et al. (2020) investigated the performance of the following one-class classifiers: One Class Support Vector Machine (OCSVM), Support Vector Data Description (SVDD), Global k-Nearest Neighbors (GKNN), Local Outlier Factor (LOF), Isolation Forest (IF), and Angle-Based Outlier Detection (ABOD). To that end, a real-data validated numerical simulator developed for a Frigate characterised by a combined diesel-electric and gas propulsion plant was utilised for a case study implementation. Based on the outlined results, the authors sorted the performance of the six analysed algorithms as follows: $ABOD > OCSVM \approx SVDD > GKNN > IF \approx LOF$. Tan et al. (2021) presented an analogous comparative study, although the topic of study in this case was multi-label classification for simultaneous fault diagnosis. The comparative

study consisted of analysing a total of five models: 1) Binary Relevance (BR), 2) Classifier Chains (CC), 3) multi-label k-nearest neighbour (MLKNN), 4) Binary Relevance k-nearest neighbour (BRKNN), and 5) multi-label twin support vector machine (MLTSVM). Analogous to Tan et al. (2020), a dataset generated from a real data simulator of a Frigate was considered for the performance of the case study. Based on the outlined results, the authors sorted the performance of the five analysed algorithms as follows: $BR > CC > BRKNN > MLKNN > MLTSVM$.

Of all the methodologies implemented, four of the six identified studies presented an analysis of a version of SVM. The remaining two studies referred to DL approaches, in which the application of either deep neural networks or recurrent neural networks have been considered at some extent. However, although analogous industries have exploited the potential of powerful methods of image processing and time series imaging for fault detection and diagnostics (Zio, 2022), there is no evidence that such practices have been analysed and formalised within the shipping sector.

For instance, Fahim et al. (2021) proposed a self-attentive weight-sharing capsule network (WSCN) to perform both fault detection and classification in the transmission line domain. Prior to the implementation of WSCN, the authors encoded the time-series signal into an image by implementing the Gramian Angular Field (GAF) algorithm. The authors highlighted that transforming the time-series signal into an

image is significant in revealing certain fault features and patterns that cannot be extracted from the original time-series signal. A Western-System-Coordinating-Council WSCC 9-bus and 3-machine test model modified with the series capacitor was analysed to determine the robustness of the self-attention WSCN.

Fahim et al. (2021b) introduced a unified unsupervised learning framework for short circuit fault analysis of a power transmission line. Analogous to Fahim et al. (2021), GAF was applied to transform the time-series oscillographs into images. A stacked denoising-autoencoder was integrated and modelled to guarantee the robustness of the framework against noise. Field data was considered for a case study with three types of fault classification results. Fahim et al. (2021c), Fahim et al. (2020), and Fahim et al. (2020b) also implemented GAF for image representation of sampled signals. Such images would then be considered as inputs of the proposed model.

Yao et al. (2020) proposed a framework for fault diagnosis with Full-scope Simulator based on the State Information Imaging (FDFSSII). FDFSSII aimed to construct a series of grey images that presented the operating transient (both normal and fault condition) according to the real time monitoring data. The image feature was extracted by implementing Kernel Principal Component Analysis (KPCA). Then, such image features were classified by the designed classifiers. Specifically, a total of five typical classifiers were considered: 1) SVM, 2) k-NN, 3) Linear Discriminant

Analysis (LDA), 4) Decision Tree Analysis (DTA), and 5) logistic regression. A case study based on the nuclear plant-wide fault diagnosis system was presented.

Kiangala and Wang (2020) developed a classification model based on time-series imaging and CNN. The image representation was obtained by implementing GAF. PCA was also performed to apply feature extraction, and thus reduce the dimensions of the dataset considered into two channels. A case study based on data collected from the conveyor system of a small manufacturing was presented.

Of all the studies identified about time series imaging applied to fault classification, only two distinct approaches could be perceived: 1) GAF, and 2) FSFSSII. Specifically, all studies except one, which implemented the FSFSSII approach, considered GAF for encoding the signals into images. Thus, there is a need for exploring new methods for image representation from time series, as time series imaging has demonstrated their ability of discovering fault features and patterns that cannot be obtained from the original version of the time series. A summary of the analysed references precedingly described within this section are shown in Appendix B Tables B.3 and B.4.

2.3.3. Predictive Analytics

Predictive analytics aims to implement prognostic-based maintenance, which is a novel approach that aims to predict the RUL of machinery based on historical and

on-going degradation trends. This is achieved by predicting likely fault cases through the condition monitoring of marine machinery (Lei et al., 2018). This approach is usually triggered by the preceding phase, which is diagnostic analytics. Once a fault has been detected, isolated, and identified, fault prognosis is implemented by estimating the Remaining Useful Life (RUL) of the asset, and thus predicting the progression of the identified fault.

RUL can be defined as the length of time that an asset is likely to operate prior to the requirement of either repair or replacement (Tang et al., 2020). Estimation methods for RUL are largely divided into four categories: 1) physical model-based approach, 2) statistical model-based approach, 3) AI-based approach, and 4) hybrid approaches (Zeng et al., 2021). Physics model-based approaches describe degradation processes through the consideration of mathematical models and the bases of the failure mechanisms. Statistical model-based approaches, a.k.a. empirical model-based approaches, implement statistical models based on observed evidence, generally representing the RUL prediction as a conditional Probability Density Function (PDF). AI-based approaches, meanwhile, implement artificial intelligence to learn the machinery degradation patterns from available observations. The fourth and final category refers to hybrid approaches, which integrate different methodologies to address the limitations of the preceding categories (Lei et al., 2018).

In recent years, RUL approaches have been widely developed and implemented in various sectors. For instance, Zhang et al. (2021) proposed a systematic method for degradation modelling and remaining useful life prediction based on uncertain process for degradation recovery phenomenon. Accordingly, uncertain process was initially adopted for modelling degradation and accounting for epistemic uncertainty. Subsequently, a novel similarity based-uncertain weighted least squares estimation method was applied to update the model parameters with real-time monitoring data. A denoising method was then also utilised to deal with noises caused by recovery phenomenon while RUL was calculated by uncertain simulation. A case study on real lithium-ion battery degradation dataset was performed. Li et al. (2021) presented a wiener-based RUL prediction method utilising improved Kalman filtering and an adaptive modification algorithm. An experimental bearing dataset was introduced to validate the methodology. Xiao et al. (2021) proposed a new RUL prediction method where noise was intentionally added into a LSTM network. Correlation analysis was also conducted to construct new degradation features as the input of the network. The C-MAPSS aero-engine lifetime dataset was utilised to validate the effectiveness of the proposed methodology. Ellefsen et al. (2019c) investigated the effect of unsupervised pre-training in RUL predictions utilising a semi-supervised setup. A Genetic Algorithm (GA) approach was presented to tune the number of diverse hyper-parameters in deep architectures. The proposed approach was verified in a similar way by using on the C-MAPSS dataset. Berghout et al. (2020) presented a new Denoising Online Sequential Extreme Learning Machine (DOS-ELM) with Double Dynamic

Forgetting Factors (DDFF) and Updated Selection Strategy (USS). The proposed methodology was also tested on the C-MAPS dataset. Xiang et al. (2020) presented a new type of LSTM with weight amplification. Monitoring data of a gear life cycle test was introduced to assess the effectiveness of the proposed methodology. All studies highlighted the importance of accurate degradation modelling to identify the underlying component degradation processes, and thus avert any possible failure, as components usually presents a degradation pattern before the total failure occurs.

Similarly, the application of deep learning methodologies in this context has also been widely considered (Yao et al., 2021). Kang et al. (2021) implemented a multilayer perceptron neural network (MLP). Djedidi et al. (2021) utilised auto-regressive neural (NAR) network to model the trend of the drift. Shi et al. (2020) presented an estimation method of RUL for drop system based on both the principal component analysis and the Bayesian inference methods. Yao et al. (2021) proposed an improved one-dimensional convolution neural network (1D-CNN) and a Simple Recurrent Unit (SRU) network. Zeng et al. (2021) applied a Deep Attention Residual Neural Network (DARNN). Cao et al. (2021) introduced a new deep learning framework, a.k.a. temporal convolutional network with residual self-attention mechanism (TCN-RSA). Ramadhan and Hassan (2021) integrated a Laplacian score, random search optimisation, and LSTM. Agrawal et al. (2021) implemented LSTM and Gated Recurrent Unit (GRU) models and compared the obtained results with a proposed genetically trained neural network. Tan and Teo (2021b) proposed a Multi-variable

Time Series (MTS) focused approach to prognostics that implements a lightweight Convolutional Neural Network (CNN) with attention mechanism.

In the shipping industry, however, the study of such approaches has not been extensively discussed due to the challenges the industry needs to address in relation to incomplete, unreliable, and missing data, and the lack of faulty data. Despite these challenges, some efforts have been made for the RUL prediction task. For instance, Han et al. (2021) presented a data-driven model for fault prognostics of a marine diesel engine. Accordingly, data augmentation was utilised to increase the generalisation of the network. This was defined with two LSTM layers, two feed-forward neural network (FNN) layers, one dropout layer, and a fully connected output layer. The prediction performance was verified through the utilisation of data sampled from a hybrid power lab. Run-to-failure data of two independent fault-types were collected from a total of two profiles. Tang et al. (2020) analysed various computational techniques for both the monitoring and estimation of RUL of individual energy assets. Relevance Vector Machine (RVM) was implemented for the prediction of RUL, while a k -NN was proposed for the prognostics of state of charge of back-up lead-acid batteries. Gribbestad et al. (2021) predicted the RUL of air compressors by exploring three different DL techniques (FNN, LSTM, and CNN) for predicting RUL as well as transfer learning.

Therefore, as perceived precedingly, due to several challenges within the shipping industry (e.g., lack of fault data, computation and communication complexities, and inaccurate, incomplete, and unreliable data (Aslam et al., 2020)), there is a lack of analysis and formalisation of approaches that address the RUL prediction task. Methods, such as Long-Short Term Memory (LSTM) neural networks (Gribbestad et al., 2021), and Relevance Vector Machine (RVM) (Tang et al., 2020) have been analysed. However, to the best of the author’s knowledge, time series imaging has not yet been considered for such a task, despite time series imaging demonstrated promising results when applying forecasting (Li et al., 2020b). Analogously, results in Makridis et al. (2020) demonstrated that ensemble models can be more stable than individual models. Efforts have also not been made to simulate degradation data despite the lack of degradation data perceived.

2.4. Identified Gaps

As described throughout the literature review chapter, the utilisation of data-driven models for the development of a holistic diagnostic and predictive analytics framework is still unexplored. In fact, to the best of the author’s knowledge, there is no evidence that such a framework has been proposed in the existing literature. Nevertheless, there are select studies which incorporate fault detection in their research that have analysed, to a limited extent, some of the phases introduced in these analytics. However, most of these studies did not address the challenges of using real data or did

not analyse the potential of deploying novel technologies in real time. Furthermore, the application of DL approaches has not yet been widely formalised and analysed, as only two studies were identified that addressed such an aspect.

A comparable experience is also presented when considering the fault identification task, as only six studies could be identified. Of all these studies, four of them implemented a version of SVM. The remaining two studies introduced either deep neural networks or recurrent neural networks, which are considered DL methodologies. This indicates once again the lack of exploration of more sophisticated approaches that have yielded promising results in analogous studies. The consideration of time series imaging, the analysis of ensemble methods, the need of simulating degradation features for validation purposes due to the lack of fault data, and the need of introducing novel data pre-processing techniques for enhancing data quality are just a few examples. A more comprehensive description of the gaps identified and addressed in the subsequent thesis chapters is summarised in Table 2.4.

Table 2.4. Summary of identified gaps divided by MA phase.

Phase	Gaps	Opportunities
Overall MA framework	<ul style="list-style-type: none">• There is no evidence in the academic literature that a holistic framework constituted by diagnostic analytics (fault detection and identification) and predictive analytics (remaining useful life prediction) has been formalised.	<ul style="list-style-type: none">• Introduce new frameworks to enhance the current maintenance strategies. Investigate the potential that novel data-driven methodologies may have when implementing diagnostic and predictive analytics.
Data pre-processing	<ul style="list-style-type: none">• Data imputation techniques have not been extensively analysed, although it has been determined that current datasets present between 4.4% and 26% missing values. Only one study has been identified that introduced a novel hybrid framework for performing data imputation.• Although the identification of operational states is a fundamental pre-processing step due to the existence of non-operational states within marine systems' datasets, this step is still unexplored. Some analyses have been performed, in which GMM with EM have been introduced, for instance. However, the studies did not assess their performance when the dataset included other states instead of engine operating regions. Moreover, GMM with EM requires the selection of hyperparameters prior to the detection of the different states. This aspect hinders the automation of this step, which is of preeminent importance when the framework needs to be deployed in real time.	<ul style="list-style-type: none">• Implement comparative studies of widely utilised data imputation methods.• Explore time series imaging for the identification of operational states.• Introduce DL methodologies within the shipping sector.• Create new hybrid data imputation frameworks to assess their potential in enhancing the imputation performance.

Diagnostic analytics

- None of the studies have analysed the potential of a real-time deployment. Also, although the features analysed are in the form of time series, most of the analysed frameworks did not consider temporal dependencies. DL methodologies that perform fault detection are still unexplored within the shipping industry, as only VAEs have been introduced.
- Fault identification has not been extensively analysed due to the lack of fault data within the sector. Only SVM have been extensively analysed. Although two main DL methodologies have been explored, it is still an unexplored area. For instance, the potential of time series imaging and DL for image classification has not been discussed.

- Analyse DL methods for the detection of faults.
- Introduce time series imaging approaches and image classifiers for the implementation of fault identification.

Predictive Analytics

- Challenges such as the lack of fault data and inaccurate, incomplete, and unreliable data precluded the formalisation and analysis of prognostic-based elements. Regarding the RUL prediction, only four studies have been identified. Certain potential enhancements have not been considered for this matter, such as the simulation of degradation features for both training and validation purposes, the analysis of DL methodologies, the consideration of time series imaging approaches, and the application of ensemble methods.

- Explore ways for simulating degradation patterns due to the lack of fault data.
- Introduce DL methods for performing the RUL prediction task.
- Study the potential of ensemble modelling for enhancing the prediction performance.

2.5. Chapter Summary

This chapter presented a comprehensive literature and critical review that facilitated the determination of the current novel approaches introduced for the data pre-processing, diagnostic analytics (fault detection and identification), and predictive analytics for the analysis of marine systems sensor data. Such a determination enabled the identification of the current research gaps observed in the sector so that these can be addressed accordingly in the forthcoming chapters of this thesis.

Chapter 3

Methodology: Part I. Introduced

Novelties and Overview of the Developed

Holistic MA Framework

3.1. Chapter Overview

The proposed novelties and an overall description of the MA framework are presented in this chapter. The novelties are presented in section *3.2. Novelties*, and have been defined based on the main gaps identified in section *2.4. Identified Gaps*. In addition, section *3.3. Overview of the Developed Frameworks* is introduced. This section aims to provide the structure of the Holistic MA Framework and its main modules and describe each of them in a general manner. Each of its modules and their respective novelties are then described in more detail in *Chapter 4. Methodology: Part II. The Data Pre-processing Module*, and *Chapter 5. Methodology: Part III. The Diagnostic and Predictive Analytics Modules*.

3.2. Novelties

Having identified the distinct gaps that currently need to be addressed within the shipping sector with regards to the application of innovative maintenance strategies, the novelties are determined and specified in Fig 3.1. In total, eight novelties have

been ascertained, which are categorised into a total of four groups. The first group refers to novelty 1 and aims to develop a holistic MA framework for the diagnosis and prognosis of marine systems through the application of data-driven methodologies. To the best of the author's knowledge, there is no evidence that such a framework has been introduced within the sector.

The second group of novelties refers to the individual components of the data pre-processing module. A total of four are introduced in this group (3 referring to the data imputation task, and 1 relating to the operational states' identification). The first study refers to a comparative study of widely used Machine Learning (ML) and time series forecasting algorithms. If this comparative methodology is integrated into a more holistic approach whereby the steady state identification process is analysed in more detail, the second approach is proposed. Due to the lack of analysis of DL methodologies, a third approach is considered, in which a variational autoencoder is analysed. With regards to the operational states' identification task, a new approach has been formalised. This new approach also deals with an additional novel element: the analysis of time series imaging approaches through the implementation of the first-order Markov chain for image generation.

The third group refers to the diagnostic analytics, and thus a total of two individual components are introduced: 1) fault detection, and 2) fault identification. For the fault

detection component, a novel DL model has been introduced in tandem with image thresholding for the detection of anomalies. With regards to the second component, a methodology for the identification of anomalies of marine systems is presented. Such a methodology aims to analyse the implementation of time series imaging together with DL models for the application of image classification tasks.

The fourth and final group relates to the predictive analytics module, in which a RUL prediction component is introduced. This component also deals with the lack of both fault data availability and analysis and formalisation of DL methodologies within the sector.

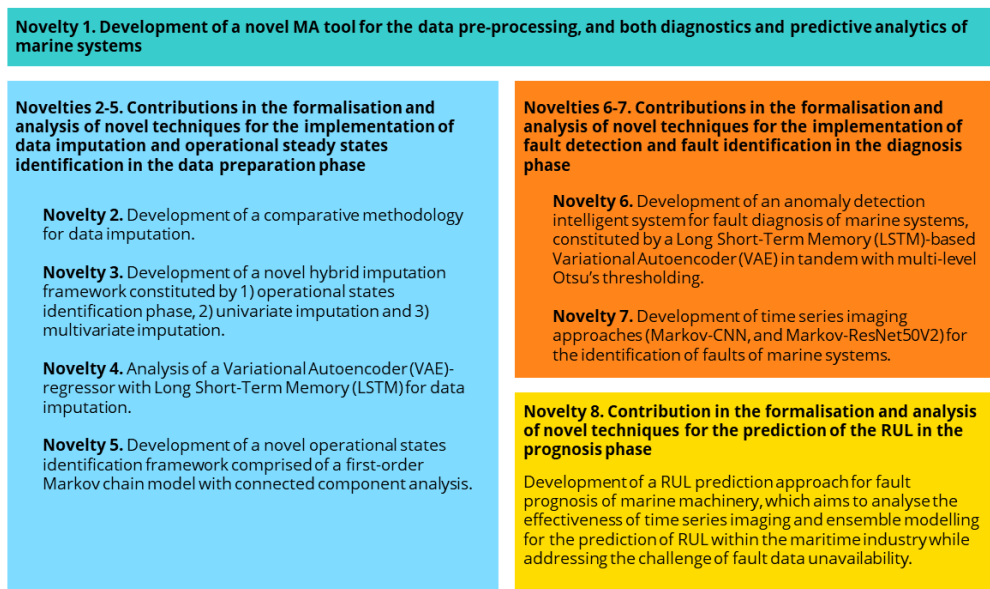


Fig. 3.1. Summary of the novelties introduced in this thesis.

The determined novelties can be summarised as follows:

- Development of a holistic MA framework for the diagnosis and prognosis of marine systems.
- Development of four individual components for the analysis and formalisation of the data imputation and the operational states' identification tasks as part of the data pre-processing module.
- Development of a diagnostic analytics module comprised of fault detection and fault identification components through the introduction of state-of-the-art data-driven methodologies.
- Development of a predictive analytics module based on the prediction of the RUL task.

3.3. Overview of the Developed Frameworks

Based on the determined novelties, the contributions are introduced. The main contribution of this thesis, which relates to novelty 1, is the proposal of an overall MA architecture for the diagnosis and prognosis of marine systems. This architecture is graphically represented in Fig. 3.2. As observed, it is constituted by a total of three main stages: 1) data pre-processing, a.k.a data preparation, 2) diagnostic analytics, and 3) predictive analytics.

A total of four distinct frameworks have been introduced in the first stage of the MA framework. These frameworks aim to introduce novel techniques for dealing with two critical data pre-processing tasks: data imputation and steady states' identification. The first framework developed under the data imputation section relates to novelty 2 and aims to introduce a comparative methodology of data imputation approaches in order to determine the most appropriate approach based on the characteristics of both the missing values and the dataset (please see a comprehensive description of this framework in *4.2. Comparative Methodology of Univariate and Multivariate Data Imputation Techniques*). The second framework, which is related to novelty 3, aims to combine distinct data imputation techniques to enhance the overall imputation performance (please see a comprehensive description of this contribution in *4.3. Hybrid Data Imputation Framework*). The third and last framework of the data imputation section relates to novelty 4. This framework contributes to the formalisation of data imputation techniques through the application of DL in the shipping sector (please see a comprehensive description of this contribution in *4.4. Analysis of LSTM-based Variational Autoencoders for Regression for Data Imputation*). With regards to the identification of operational states, a framework comprised of time series imaging and connected component analysis is introduced. This framework relates to novelty 5 and is comprehensively described in *4.5. A Novel Framework for the Identification of Steady States*.

Two additional contributions are introduced for implementing the diagnostic analytics module. The first contribution aims to develop a fault detection approach, and thus a framework based on deep learning methodologies and image thresholding is introduced. This contribution relates to novelty 6 and is comprehensively described in *5.3.1. Fault Detection*. The second framework relates to novelty 7 and aims to implement fault identification by combining time series imaging, deep learning, and image classifiers (please see a comprehensive description in *5.3.2. Fault Identification*). The last contribution aims to implement predictive analytics and relates to novelty 8. Accordingly, a RUL prediction framework is presented (please see a comprehensive description in *5.4.1. Remaining Useful Life (RUL) Prediction Framework*). The data considered for the implementation of each of these modules refer to sensors coupled to distinct marine systems (S_1, S_2, \dots, S_i).

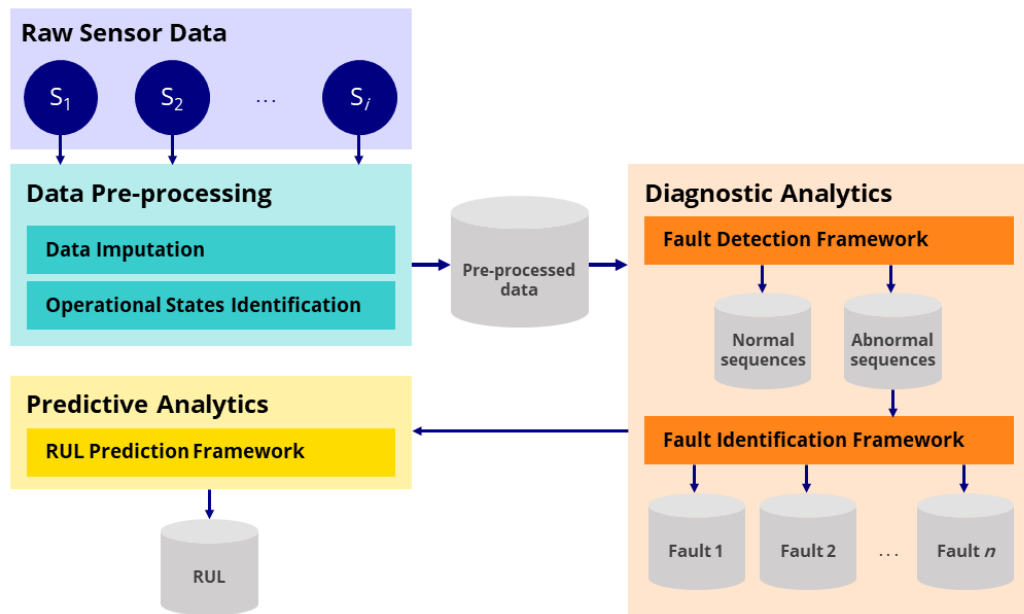


Fig. 3.2. Graphical representation of the MA framework architecture.

For a better understanding of the flow of data please refer to Fig. 3.2. As it can be perceived, the unique source of data is a raw database with sensor data. Thus, the data analysed is time series data, which is recorded over time intervals and has been collected from sensors coupled to marine machinery/systems. Due to the criticality of the machinery and/or systems analysed (e.g., main engine, and diesel generators), the data is usually collected in between 1 and 10-minute frequency basis (high-frequency data). In addition, this data usually present quality issues that need to be addressed, such as outliers, missing values, and lack of synchronisation between sensors. The required transformations, imputations, and quality checks are performed within the data pre-processing phase. However, due to the novelties introduced in this thesis, special attention is given to two data pre-processing tasks: data imputation, and operational states' identification. The pre-processed data are presented as input in the

fault detection phase. In such a phase, an initial classification is performed between normal and abnormal sequences. Appropriately, any sequence with faults should be labelled as abnormal sequences, whereas the normal sequences should be labelled as normal sequences by the algorithm. Only those sequences that have been considered as abnormal are presented as input in the following step, fault identification. In this subsequent step, a better description of the nature of the fault is expected, and thus a more comprehensive classification of the distinct faults (e.g., fuel leakage or air in fuel system). The sequences which are considered to be following a degradation pattern are set as critical and are presented as input in the last phase of the MA framework, RUL prediction. The final output of the MA is the predicted RUL that can be utilised for determining when the system will fail so that any preventive action can take place.

The subsequent two chapters aim to describe each of the introduced contributions for addressing the identified gaps and apply the determined novelties.

3.4. Chapter Summary

This chapter presented the novelties that are expected to be generated through the development and implementation of a MA framework. This proposed framework has been comprehensively described in this chapter to provide a better understanding of the novel data-driven methodologies that are presented in this thesis.

Chapter 4

Methodology: Part II. The Data Pre-processing Module

4.1. Chapter Overview

The subsequent paragraphs of this section are structured to comprehensively describe the four novelties introduced in the data pre-processing module. Thus, section 4.2. *Comparative Methodology of Univariate and Multivariate Imputation Techniques* presents the methodology considered for the development of a comparative methodology for data imputation. Section 4.3. *Hybrid Data Imputation Framework* introduces a novel hybrid imputation framework constituted by 1) operational states' identification phase, 2) univariate imputation and 3) multivariate imputation. Section 4.4. *Analysis of LSTM-VAE-based Regressor Analysis* introduces a DL methodology for data imputation. Specifically, the model VAE-regressor with LSTM is analysed. Finally, section 4.5. *A Novel Framework for the Identification of Steady States* presents a novel operational states' identification framework comprised of a first-order Markov chain with connected component analysis.

4.2. Comparative Methodology of Univariate and Multivariate Imputation Techniques¹

A methodology that provides a comparative study of a total of 20 widely used ML and time series forecasting algorithms is introduced in this section to determine the most appropriate data imputation framework based on the characteristics of both dataset and missingness.

The selection of the models has been performed based on widely known time series forecasting models (Hyndman and Athanasopoulos (2020), Kotu and Deshpande (2019)) and ML models (Kuhn and Johnson (2016)) so that their imputation performance can be analysed.

The comparative methodology introduced in this section is graphically represented in Fig. 4.1. Short-term data collected from sensors installed on critical marine machine systems are stored in a database for further processing. As raw sensor data is being considered, certain steps need to be applied prior to the imputation of missing values. The first one refers to data preparation. This needs to be implemented so that the data presents the quality required for fitting the distinct analysed models. To evaluate each of the imputation models, missing values are generated. This refers to step 2.

¹ The data imputation framework presented in this section has been already converted to a journal paper, and has been published in the Ocean Engineering journal (Velasco-Gallego and Lazakis, 2020).

Time-series cross-validation technique is applied in step 3. Steps 4 and 5 refer to the implementation of both univariate and multivariate imputation techniques, respectively. The section that applies univariate imputation techniques include mean imputation, time series decomposition techniques, exponential smoothing methods, and ARIMA models. By contrast, the section that introduces multivariate imputation techniques encompasses linear regression (Partial Least Squares regression, LASSO regression, Ridge regression, and ElasticNet regression), k -NN, support vector machines for regression (with linear and RBF kernel), neural networks (with 1, 2, and 3 hidden layers), Vector Autoregressions (VARs), decision tree regressors, and ensemble methods (Bagged trees (with SVR and k -NN regressors), random forests, and AdaBoost). Multivariate imputation is only implemented for those parameters that present predictors. The models considered in this framework have been implemented through the utilisation of the Python libraries Scikit-Learn and Statsmodel. The last step of this comparative methodology refers to the evaluation process, in which the seven metrics utilised are defined. These seven metrics refer to the Mean Squared Error (MSE), Mean Squared Logarithmic Error (MSLE), the Root Mean Square Error (RMSE), the Mean Absolute Percentage Error (MAPE), the Median Absolute Error (MedAE), and the Max Error.

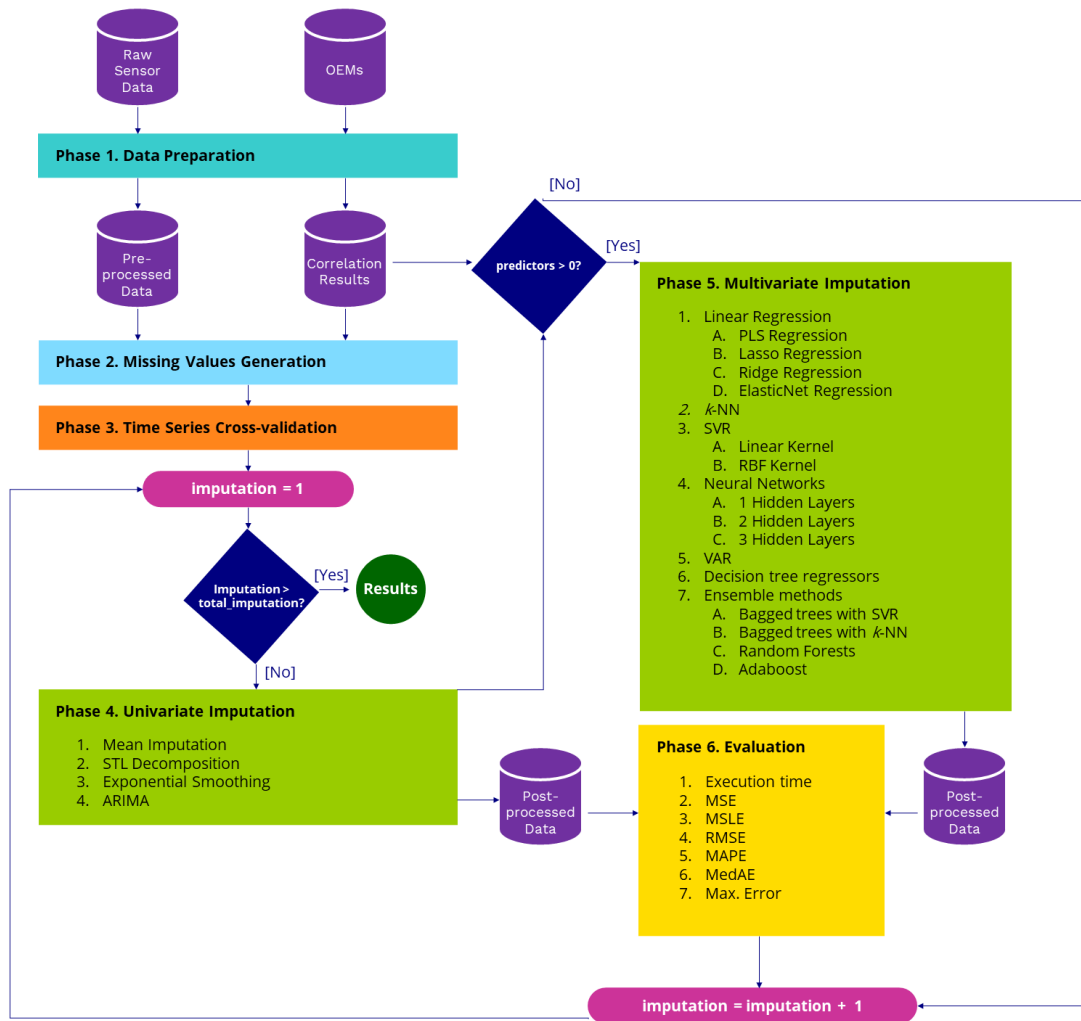


Fig. 4.1. Graphical representation of the comparative methodology of univariate and multivariate imputation techniques.

4.2.1. Data Preparation

Due to the characteristics of the data, context, and model, for this part of the methodology a total of three phases are implemented in the data preparation step. These are 1) machinery transient states identification, 2) data transformation, and 3) correlation analysis.

Machinery Transient States Identification

Datasets may contain non-operational states, such as manoeuvring and transient states of machinery, that need to be adequately identified and discarded. Such states have been identified manually through the consideration of the Original Equipment Manufacturers (OEMs) of the systems being analysed.

Data Transformation

The transformations proposed by Box and Cox are applied to remove distributional skewness (Eq. 1).

$$x' = \begin{cases} \frac{x^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0, \\ \log(x) & \text{if } \lambda = 0 \end{cases}, \quad (\text{Eq. 1})$$

where λ is a parameter that is determined empirically by training the data and applying Maximum Likelihood Estimation (MLE). Based on the value of λ various widely used transformations can be identified from this family of transformations, such as the square transformation ($\lambda = 2$), the square root transformation ($\lambda = 0.5$), and the inverse transformation ($\lambda = -1$). This transformation is only valid to transform values greater than 0.

Additionally, the data is also standardised to ensure that all the predictors contribute equally to the model. Hence, all features are centred by subtracting the mean from all values and scaled by dividing the features by their respective standard deviations (Eq. 2).

$$z = \frac{x' - \bar{x}'}{s'} \quad (\text{Eq. 2})$$

As a result of the standardisation, the standardised features present a mean of zero and a standard deviation of one.

Correlation Analysis

The Pearson's correlation coefficient is determined in order to identify the relationship between two variables. Considering two features, x and y , the Pearson's correlation coefficient results from the standardisation of each feature and subsequently estimates the mean after multiplying them (Eq. 3):

$$\rho = \frac{1}{n-1} \sum_{i=1}^n \frac{x_i - \bar{x}}{s_x} \frac{y_i - \bar{y}}{s_y} \quad (\text{Eq. 3})$$

If Eq. 3 is rewritten by considering the covariance term, which is defined as the mean of the product of the deviations, Eq. 4 is obtained.

$$\rho = \frac{\text{Cov}(x, y)}{s_x s_y} \quad (\text{Eq. 4})$$

Pearson's correlation coefficient always lies between -1 and +1. The strength of the relationship is identified through the magnitude obtained. If the resulting coefficient is either -1 or 1 it indicates that the features are perfectly correlated. Conversely, if Pearson's correlation coefficient is 0 it means that a linear relationship does not exist between them.

Pearson's correlation coefficient presents some limitations, such as its sensitiveness to outliers and its inability to capture non-linear relationships. For this reason, the Spearman's rank coefficient is also estimated. This coefficient is based on the rank of the data. Its robustness to outliers and its ability to capture certain non-linear relationships are its main advantages in comparison with the Pearson's correlation coefficient.

4.2.2. Missing Values Generation

To adequately compare the distinct analysed models, missing values need to be generated. Accordingly, missing values are generated completely at random. The ratio of missingness is set to 0.3, as this ratio encompasses the range of missing values (4.4-26%) identified in analogous datasets (Cheliotis et al., 2019).

4.2.3. Time Series Cross-validation

Certain models present hyperparameters that need to be selected. Such hyperparameters are critical due to their control in the complexity of the model. Moreover, they cannot be estimated from the sample and an inadequate selection of these may lead to either under-fitting or over-fitting of the applied models. Accordingly, certain techniques need to be selected to optimally select these hyperparameters.

In this study time series cross-validation is implemented. This is a cross-validation technique that defines the training set by only considering the instances that precede the instances that constitute the test set (Fig. 4.2). Therefore, when imputing the missing values, the evolution of the feature through time is examined, which enables the imputation of the missing values in real-time.

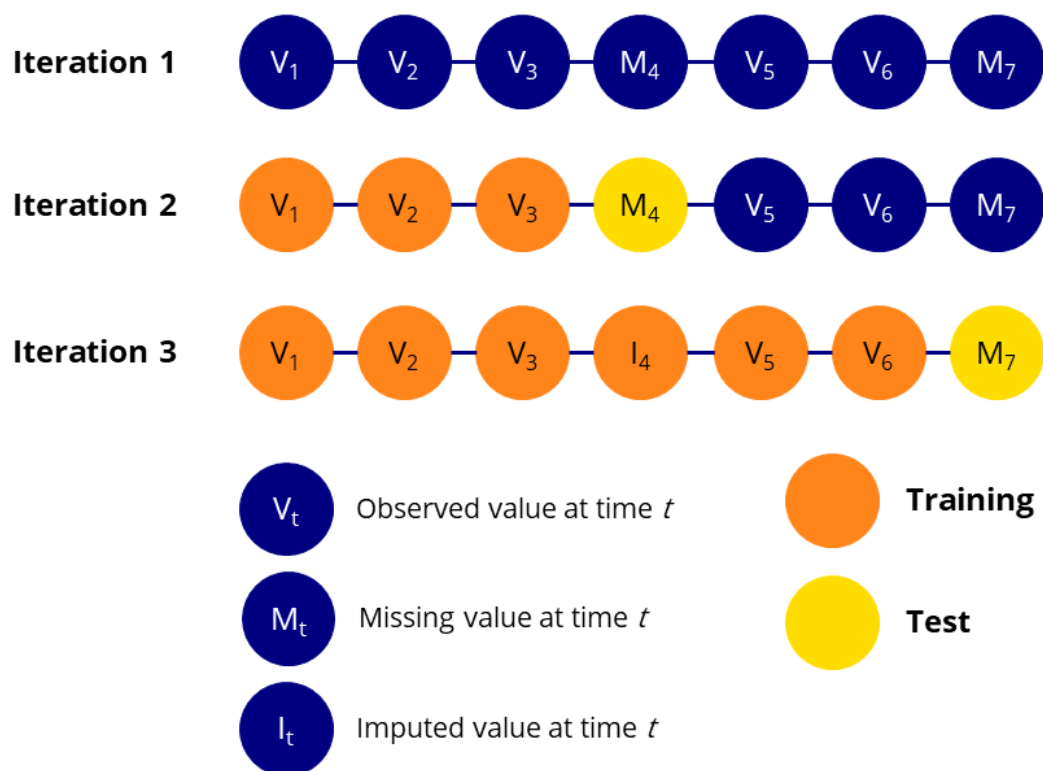


Fig. 4.2. Time series cross-validation.

4.2.4. Univariate Imputation

Mean Imputation

The mean imputation technique is a simple forecasting method. This technique imputes missing values by estimating the mean of the sample:

$$\hat{y}_t = \bar{y} = \frac{1}{t-1} \sum_{i=1}^{t-1} y_i, \quad (\text{Eq. 5})$$

where \hat{y}_t is the predicted value at the current time t , \bar{y} is the mean, and y_i is the sensor measurement at time i .

Seasonal and Trend Decomposition using Loess Method

The Seasonal and Trend decomposition using Loess (STL) method aims to decompose the original time series into three distinct components that capture trend, seasonal, and residual patterns. The trend component captures the evolution of the series through time. The seasonal component detects fluctuation patterns that are repeated along time due to seasonal factors. Those irregularities that do not correspond to either the trend or seasonal components are reflected in the residual component.

By determining these three components, the missing values can be imputed by considering an additive decomposition and by forecasting the time series components (Eq. 6).

$$\hat{y}_t = \hat{S}_t + \hat{T}_t + \hat{R}_t, \quad (\text{Eq. 6})$$

where \hat{S}_t is the forecasted seasonal component, \hat{T}_t is the forecasted trend component, and \hat{R}_t is the forecasted residual component. The forecast of the seasonal component is performed based on the assumption that the seasonal component is unchanging, or slightly changing, and thus its value at time t can be predicted by implementing a seasonal naïve method. Conversely, the seasonally adjusted component, defined as the

addition of forecasted trend and residual components, is estimated by applying a non-seasonal forecasting method, such as the simple exponential smoothing method. This method is presented in the subsequent section.

Exponential Smoothing methods

The simple exponential smoothing method is usually utilised when the time series does not present either trend or seasonality in a distinct manner. This results from the fact that the estimation of the predicted values is based on the weighted averages method. Thus, the most recent observations present the greatest values whilst observations addressed further back decrease exponentially in weight (Eq. 7).

$$\hat{y}_t = \sum_{i=0}^{t-1} \alpha (1 - \alpha)^i y_{t-i} + (1 - \alpha)^t l_0 \quad (\text{Eq. 7})$$

where α is the smoothing parameter, the value of which falls between 0 and 1 (inclusive), and l_0 is the least recent observation. Thus, less recent observations present a weight close to 0, and thus do not significantly influence the predicted values, whereas more recent observations have a weight near 1, which indicates that they have a major impact in the imputation of the missing values. Accordingly, the smoothing parameter, α , and the least recent observation, l_0 , need to be estimated by implementing cross-validation.

Holt's linear trend method is an extension of the simple exponential smoothing technique in which the trend of the time series is also considered to impute the missing values. The component form of this method is expressed hereunder.

$$\text{Forecast equation } \hat{y}_t = l_t + b_t \quad (\text{Eq. 8.1})$$

$$\text{Level equation } l_t = \alpha y_t + (1 - \alpha)(l_{t-1} + b_{t-1}) \quad (\text{Eq. 8.2})$$

$$\text{Trend equation } b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1}, \quad (\text{Eq. 8.3})$$

where l_t and b_t are the estimates of the series and of the trend of the series at time t respectively, and α and β are the smoothing parameters for the level and for the trend, which fall between 0 and 1 (inclusive).

Similarly, the seasonal component can also be captured by exponential smoothing methods, specifically through the Holt Winder's additive method, in which and additional seasonal equation is considered in the component form.

$$\text{Forecast equation } \hat{y}_t = l_t + b_t + s_{t-m} \quad (\text{Eq. 9.1})$$

$$\text{Level equation } l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1}) \quad (\text{Eq. 9.2})$$

$$\text{Trend equation } b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \quad (\text{Eq. 9.3})$$

$$\text{Seasonal equation } s_t = \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m}, \quad (\text{Eq. 9.4})$$

Autoregressive Integrated Moving Average (ARIMA) Models

To adequately implement ARIMA models, the time series is required to be stationary.

This is presented when both trend and seasonal components are not clearly identified.

both Kwiatkowski-Phillips-Schmidt-Shin (KPSS) and Augmented Dickey-Fuller

(ADF) unit root tests are considered for determining if the analysed time series is

stationary. Accordingly, a time series is recognised as stationary only if both tests

reject the hypothesis that the time series is non-stationary.

If the time series is identified as non-stationary then differencing is implemented (Eq.

10).

$$y'_t = y_t - y_{t-1} \tag{Eq. 10}$$

The first model analysed in this classification is the autoregressive models. These

models in tandem with moving average models form the ARIMA models. When

implementing autoregressive models, the missing value to be imputed is predicted by

applying linear combination of prior occurrences (Eq. 11).

$$y'_t = c + \phi_1 y'_{t-1} + \phi_2 y'_{t-2} + \dots + \phi_p y'_{t-p} + \varepsilon_t, \tag{Eq. 11}$$

where ε_t is white noise. Analogously, the linear combination of prior predicted errors, known as moving average models, may also be performed to impute missing values.

The expression of a moving average model of order q is described hereunder.

$$y_t = c + \varepsilon_t + \vartheta_1\varepsilon_{t-1} + \vartheta_2\varepsilon_{t-2} + \dots + \vartheta_q\varepsilon_{t-q}. \quad (\text{Eq. 12})$$

By combining differencing with autoregression and a moving average model, the non-seasonal ARIMA model is constituted (Eq. 13).

$$y'_t = c + \phi_1y'_{t-1} + \phi_2y'_{t-2} + \dots + \phi_p y'_{t-p} + \vartheta_1\varepsilon_{t-1} + \vartheta_2\varepsilon_{t-2} + \dots + \vartheta_q\varepsilon_{t-q} + \varepsilon_t. \quad (\text{Eq. 13})$$

The orders of the discussed models are determined by minimising the Bayesian information criterion.

4.2.5. Multivariate Imputation

Linear Regression

The linear relationship between a response feature (dependent variable) and one or more explanatory features (independent variables) is modelled through the implementation of linear regression methodologies. Thus, the missing values of the dependent variable can be imputed based on the independent variables' instances.

Simple linear regression is considered when only one independent variable or predictor is available. This technique imputes the response variable from the predictor based on the estimation of the regression line that best models the relationship between these two features:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x, \quad (\text{Eq. 14})$$

where \hat{y} is the predicted response, $\hat{\beta}_0$ is the intercept, $\hat{\beta}_1$ is the slope or regression coefficient, and x is the explanatory variable.

The best adjusted regression line is obtained through the implementation of the Ordinary Least Squares (OLS) regression method. OLS estimates the regression line that minimises the sum of squared residual values, being a residual value the difference between the observed and the predicted value ($e_i = y_i - \hat{y}_i$).

A generalisation of the simple linear regression model is the multiple linear regression model, where the response variable is related to k explanatory variables (x_1, x_2, \dots, x_k):

$$y = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k + e. \quad (\text{Eq. 15})$$

The predictors considered in the multiple linear regression may be strongly correlated, which implies both high variability and instability of the solution provided by the OLS. In such cases, the Partial Least Squares (PLS) is suggested. PLS seeks linear combinations between predictors, also named components, which are selected to not only summarise the variation of the predictors at maximum but also ensure that the estimated components present maximum correlation with the response. Hence, PLS is applied prior to linear regression model creation when the independent variables are highly correlated to obtain the components that will be utilised as predictors. To estimate the optimal number of components to be retained cross-validation is applied.

Collinearity between predictors can also be treated with biased models, as the entire MSE can be reduced when a trade-off between the bias and the variance is applied due to the fact that the variance can be reduced by increasing the bias slightly. These biased regression models can be created by adding a penalty, which regularises the parameter estimates, to the sum of the squared error. Ridge regression considers the addition of a second-order penalty on the parameter estimates (Eq. 16).

$$\text{SSE}_{L_2} = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^P \beta_j^2, \quad (\text{Eq. 16})$$

where λ is the penalty parameter. The greater this value is the more the method shrinks the estimates towards 0. An alternative to ridge regression is the Least Absolute Shrinkage and Selection Operator (LASSO) model, in which the absolute value of each parameter is added to regularise the parameter estimates (Eq. 17).

$$\text{SSE}_{L_1} = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^P |\beta_j|. \quad (\text{Eq. 17})$$

By applying this type of regularisation, some parameters may be set to 0, and thus these are not considered in the penalised regression model. Hence, LASSO regression method is not utilised to improve the model accuracy only, but also to implement feature selection.

Additionally, an extended version of Ridge and LASSO regression methods that combine the two penalties is the ElasticNet regression method, the SSE of which is expressed hereunder.

$$\text{SSE}_{\text{Enet}} = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \lambda_1 \sum_{j=1}^P \beta_j^2 + \lambda_2 \sum_{j=1}^P |\beta_j|. \quad (\text{Eq. 18})$$

The main advantage of this model is that the regularisation is enabled effectively by adding ridge penalty and the feature selection is applied by applying the LASSO penalty in a quality manner. As the entire penalised regression models considered in this study present hyperparameters, cross-validation is applied to achieve their optimal values.

k-Nearest Neighbors (k-NN)

k-Nearest Neighbors (*k*-NNs) imputes the missing values by considering *k*-closest records from the training set. Thus, the value of the response is obtained by estimating the mean of the *k*-nearest neighbors, which are selected by determining the Euclidean distance between the samples (Eq. 19).

$$d_{\text{Euclidean}} = \left(\sum_{i=1}^P (x_{1i} - x_{2i})^2 \right)^{\frac{1}{2}}, \quad (\text{Eq. 19})$$

where x_1 and x_2 are the analysed instances. The utilisation of the distance metric in this method can disrupt the prediction of the response if any predictor value is missing, as then the distance cannot be estimated. Hence, to address this problem, either the instances that contain missing values are excluded from the analysis or a univariate imputation method, such as mean imputation, is implemented prior to the *k*-NNs model to impute the missing values. However, to assess the accuracy of the model, it

is considered the predictors in the analysed sample do not contain missing values. Additionally, another aspect to be addressed in the k -NNs method is the number of neighbours to be considered, which, to avoid over-fitting, is estimated by the square root of the total number of occurrences of the sample.

An enhancement of this method is the weighted k -NNs, in which a weight is added to the selected neighbours to regularise their contribution into the response prediction. Thus, those neighbours that are closer present more weight, and thus contribute more to the response prediction, whereas those that present a greater distance contribute less.

Support Vector Machine (SVM)

The Support Vector Machine (SVM) for regression the model is formed by only those data points the residuals of which present an absolute difference greater than a given threshold, denoted as ϵ . The coefficients of the SVM regression model minimise

$$C \sum_{i=1}^n L_{\epsilon}(y_i - \hat{y}_i) + \sum_{j=1}^P \beta_j^2,$$

where $L_{\epsilon}(\cdot)$ is the ϵ -insensitive function, and C is the cost parameter, which penalises large residuals. The prediction function of the SVM can be written as

$$f(\mathbf{u}) = \beta_0 + \sum_{i=1}^n \alpha_i K(\mathbf{x}_i, \mathbf{u}), \quad (\text{Eq. 20})$$

where $K(\cdot)$ is the kernel function. Among the kernel functions that encompass non-linear functions of the predictors, the Radial Basis Function (RBF) is considered in this study (Eq. 21).

$$K(\mathbf{x}_i, \mathbf{u}) = \exp(-\sigma \|\mathbf{x}_i - \mathbf{u}\|^2) \quad (\text{Eq. 21})$$

In addition, the linear kernel is also considered (Eq. 22).

$$K(\mathbf{x}_i, \mathbf{u}) = \mathbf{x}_i' \mathbf{u} \quad (\text{Eq. 22})$$

This technique also presents hyperparameters, which are the cost parameter C and the threshold ϵ . Additionally, the RBF also presents the σ parameter, the optimal value of which also needs to be estimated. Thus, cross-validation is also applied in this case to achieve their optimal values.

Neural Networks (NNs)

Neural Networks (NNs) are another example of non-linear regression methods, inspired by biological neural networks, in which the response is modelled by hidden units. These hidden units are defined as an intermediary set of unobserved variables constituted by either linear or non-linear combinations of the predictors, the outcomes of which are combined again to either be utilised as inputs for the subsequent hidden layer or to predict the response.

To estimate the optimal parameters that constitute the combinations applied along the NN, the weight decay penalisation method is utilised to regularise the model by adding a penalty for large regression coefficients. Hence, by applying this regularisation over-fitting is moderated.

Vector Autoregressive (VAR) models

As ARIMA models, the VAR models are limited to be accurate only by those features that present to be stationary. This is due to the fact that the VAR models are a generalisation of the univariate autoregressive models, in which not only a linear combination of prior occurrences of the analysed feature is considered but also bi-directional relationships between features are also included, as the overall features are considered as endogenous. Hence, the VAR models are able to predict a vector of time

series iteratively by generating predictions for each feature included in the model, as it is expressed in (Eq. 23) for a VAR model of order 1 and dimension 2.

$$\hat{y}_{1, t} = \hat{c}_1 + \hat{\phi}_{11, 1}\hat{y}_{1, t-1} + \hat{\phi}_{12, 1}\hat{y}_{2, t-1} \quad (\text{Eq. 23.1})$$

$$\hat{y}_{2, t} = \hat{c}_2 + \hat{\phi}_{21, 1}\hat{y}_{1, t-1} + \hat{\phi}_{22, 1}\hat{y}_{2, t-1} \quad (\text{Eq. 23.2})$$

The order of the model is determined by minimising the Bayesian information criterion.

Decision trees regressors

The basis of the prediction of decision tree regressors is established by partitioning the feature space into subspaces in an iterative manner. The tree begins with the root node. This is constituted with the first partition that splits the data into disjoint sets, which in turn are divided into smaller partitions. Subsequent children nodes are then split until the optimal number of partitions is achieved.

For regression, from the overall sample, this methodology begins by partitioning the data into two groups, of which the sums of the squared errors are minimised (Eq. 24).

$$\text{SSE} = \sum_{i \in S_1} (y_i - \bar{y}_1)^2 + \sum_{i \in S_2} (y_i - \bar{y}_2)^2, \quad (\text{Eq. 24})$$

where \bar{y}_1 and \bar{y}_2 are the mean of the groups of data S_1 and S_2 respectively. Iteratively, the subsequent groups of data are split until the optimal number of partitions are achieved. To avoid over-fitting, a complexity parameter can be included to penalise the error rate by utilising the tree size (Eq. 25).

$$\text{SSE}_{c_p} = \text{SSE} + c_p n_{\text{terminal nodes}}, \quad (\text{Eq. 25})$$

where c_p is the complexity parameter.

Ensemble methods

The utilisation of single regression trees is likely to present sub-optimal predictive performance due to their limitations, such as their instability. For this reason, ensemble methods are suggested, as they tend to present better performance.

An example of which is the bootstrap aggregation trees, also referred as bagged trees, in which bootstrapping is implemented in tandem with a regression model. Hence, this ensemble method generates m samples, obtained by implementing bootstrapping from the original data. Then, an unpruned tree model is trained on each resulting sample, the predictions of which are averaged to obtain the resulting bagged model's prediction.

Random forest is another example of ensemble method that adds randomness into the learning process to reduce correlation among predictors. Analogous to bagged trees, the method generates m samples, obtained by implementing bootstrapping from the original data. However, the tree model on each sample is trained by applying random split selection, in which the tree is modelled by utilising a random subset of the top k predictors at each split in the tree. The resulting random forest model's prediction is obtained by estimating the average of the samples' predictions.

Additionally, boosting methods are another class of ensemble methods, the origin of which initially was to solve classification problems. The basis of these types of methods is the recursive modelling of compositions, in which each subsequent model learns by utilising the error information identified in the previous one. The adaptive boosting technique, also referred as AdaBoost, is an example of a boosting method, which implements weight adjustment procedures based on the errors of the current predictions. Hence, in each iteration, larger weights are assigned to more complicated predictions so that the succeeding tree can target them in more detail.

4.2.6. Evaluation

Seven metrics are considered to evaluate the performance of the imputation techniques.

Execution Time

The execution time is obtained by applying the difference between the function end time and the function start time.

Mean Squared Error (MSE)

MSE is obtained by estimating the mean of the sum of the squared errors, as defined in (Eq. 26).

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (\text{Eq. 26})$$

where n corresponds to the number of samples, and y_i and \hat{y}_i refers to the i -th occurrence of the observed and the predicted values, respectively. MSE is probably the most generally utilised loss function for regression, as if $\frac{1}{n}$ is discarded from the equation, the Least Square Errors function (L_2) is obtained. As the errors are squared, this metric penalises larger errors, which makes MSE sensitive to outliers.

Mean Squared Logarithmic Error (MSLE)

MSLE refers to the expected value of the squared logarithmic error (Eq. 27).

$$\text{MSLE} = \frac{1}{n} \sum_{i=1}^n (\ln(1+y_i) - \ln(1 + \hat{y}_i))^2 \quad (\text{Eq. 27})$$

where $\ln(x)$ refers to the natural logarithm of x . MSLE penalises under-predicted estimates more than over-predicted ones, and thus asymmetry is introduced in the error curve.

Root Mean Square Error (RMSE)

RMSE is a type of scale-dependent error, which indicates that the estimated errors are on the same scale as the observations, and its value is obtained by estimating the squared root of MSE (Eq. 28).

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (\text{Eq. 28})$$

Mean Absolute Percentage Error (MAPE)

Another type of error is the percentage error, which is given by (Eq. 29).

$$p_i = \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (\text{Eq. 29})$$

The Mean Absolute Percentage Error (MAPE), defined as the mean of the sum of the percentage errors (Eq. 30), is a widely used percentage error metric, which is also

computed for model evaluation, as it is unit-free. Conversely, one of its drawbacks is that its value is undefined if the observed value is 0.

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n p_i \quad (\text{Eq. 30})$$

Median Absolute Error (MedAE)

MedAE is computed by estimating the median of all absolute differences between the observed and the predicted occurrences (Eq. 31).

$$\text{MedAE} = \text{median}(|y_1 - \hat{y}_1|, \dots, |y_n - \hat{y}_n|) \quad (\text{Eq. 31})$$

Contrary to the other metrics introduced in this section, MedAE is robust to outliers due to the consideration of the median performance, which makes this regression metric particularly interesting.

Max Error

The maximum residual error is also computed to capture the worst-case error.

$$\text{Max Error} = \max(|y_i - \hat{y}_i|) \quad (\text{Eq. 32})$$

4.3. Hybrid Data Imputation Framework²

Once analysed some widely used ML and time series forecasting methods for data imputation in the preceding section, a hybrid data imputation framework is suggested, which brings together a number of features based on the gaps identified in the literature review section. These are related to the unavailability or incompleteness of the predictors' dataset, which are addressed through the implementation of the first-order Markov chain as a univariate imputation technique. Also, the lack of analysis and formalisation of a data imputation framework in the maritime industry is tackled by presenting a novel data imputation approach that can be introduced in a holistic predictive framework. Furthermore, parts of the comparative methodology presented in the preceding section are implemented as a multivariate imputation method to provide a general data imputation approach. A graphical representation of this framework is expressed in Fig. 4.3. The main contributions of this hybrid methodology are introduced hereunder.

- The development of a hybrid framework that combines the following phases:
 - 1) operational states' identification, 2) univariate imputation, and 3) multivariate imputation. To the best of the author's knowledge there is no evidence that such a hybrid approach has been proposed within the shipping

²The data imputation framework presented in this section has been already converted to a journal paper, and has been published in the *Ship and Offshore Structures* journal (Velasco-Gallego and Lazakis, 2021).

industry to address the challenges related to the identification of operational states and the imputation of missing values.

- The analysis and formalisation of the k -means algorithm for the identification of operational states. As presented in section 2.3.1.2. *Identification of Operational States*, there is no evidence to the best of the author's knowledge that this algorithm has been formalised within the shipping industry for the identification of steady states.
- The consideration of the first-order Markov chain model for the imputation of missing values. As presented in section 2.3.1.1. *Data Imputation*, there is no evidence to the best of the author's knowledge that this algorithm has been formalised within the shipping industry for the imputation of missing values when, for instance, there are not predictors available and univariate imputation needs to be performed.
- The consideration of a multivariate imputation approach, which is comprised of a comparative methodology in order to determine the most appropriate imputation approach based on the characteristics of both the dataset and missingness.

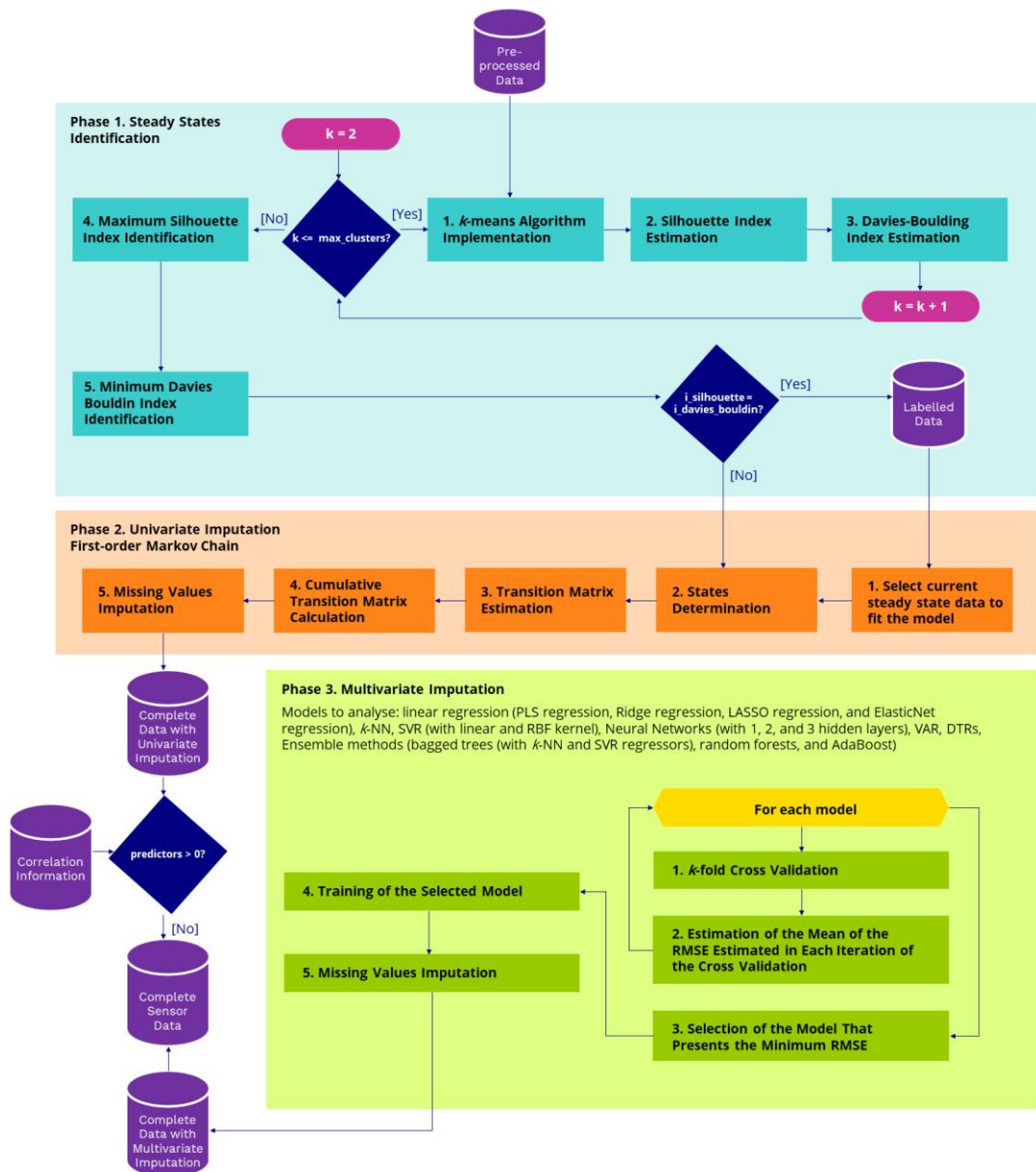


Fig. 4.3. Graphical representation of the hybrid data imputation framework.

4.3.1. Steady states' identification

When analysing sensor data collected from marine machinery, operational steady states are usually encountered. Such states are initiated after an abrupt change and persist over a certain period of time before another adjustment occurs. These abrupt

changes refer to either weather conditions or to small adjustments that are applied due to the contractual agreements between the charterer and the shipowner in relation to the vessel speed and the fuel oil consumption per day.

Clustering is a widely used technique implemented to identify substantial groups of data. The records of each group are analogous to one another but differ significantly with the ones clustered in the other groups. Hence, clustering can be applied to divide the time series automatically into the different steady states perceived in the sample. Then, when a missing value is detected, the current cluster is selected to be used as a feature to implement the first-order Markov chain. Among all the clustering techniques, the k-means clustering is utilised in this study due to its simplicity and scalability. One of the major limitations of this algorithm is the selection of the optimal number of clusters. Consequently, both the Silhouette (Starczewski and Krzyżak, 2015) and Davies-Boulding indices are applied to select the most appropriate number of clusters.

k-means Algorithm Implementation

k-means clustering splits the time series into *k* clusters by minimising the within-cluster sum of squares, which is the sum of the squared distances of each instance to the mean of its selected cluster. To implement *k*-means, it is necessary to determine the number of clusters, *k*, and the initial cluster centres. As the optimal number of

clusters is unknown, the algorithm is implemented for a range of a number of clusters, and, subsequently, different validity indices are applied to estimate the most appropriate one. The initial cluster centres are assigned randomly.

Once the number of clusters and the cluster centres are both determined, the following iterative process is performed:

- i. Clusters' assignment: each instance is assigned to the nearest cluster.
- ii. New clusters computation: the new clusters are computed based on the instance assignments.

This iterative process is implemented until the algorithm is converged. Then, each of the instances are assigned to a final cluster.

One of the major limitations of this algorithm is the selection of the optimal number of clusters. For this reason, k-means is implemented for different k values to select the most appropriate one through the utilisation of two validity indices, which are the Silhouette and the Davies-Bouldin indices.

Silhouette Index Estimation

Time series data of marine machinery parameters, such as the main engine power and the main engine rotational speed, may present steady states that are initiated after an abrupt change and persist over a certain period of time before another adjustment occurs. These abrupt changes refer to either weather conditions or to small adjustments that are applied due to the contractual agreements between the charterer and the shipowner in relation to the vessel speed and the fuel oil consumption per day.

The Silhouette index is implemented to validate the consistency within the estimated clusters, as it estimates the similarity between the instance and the cluster to which the instance was assigned in relation to the remaining clusters. This index is expressed in Eq. 33, where a is the mean intra-cluster distance and b is the mean nearest-cluster distance for each sample.

$$S = \frac{b - a}{\max(a, b)} \quad (\text{Eq. 33})$$

S always lies between -1 and +1. If the Silhouette value is closed to -1 it indicates that the sample is misclustered, whereas a value of +1 specifies that the sample is well-clustered. However, values near 0 may imply overlapping clusters.

Davies-Bouldin Index Estimation

Analogously, the Davies-Bouldin index is another index that is implemented to validate the consistency within the estimated clusters. This index is defined as the average similarity between clusters, the similarity being a measure that analyses the distance between clusters with their respective sizes. Thus, the similarity can be expressed as

$$R_{ij} = \frac{s_i + s_j}{d_{ij}} \quad (\text{Eq. 34})$$

where s is the average distance between each record and the centroid of its respective cluster, and d_{ij} is the distance between the cluster centroids. Then, the Davies-Bouldin index can be defined as

$$DB = \frac{1}{k} \sum_{i=1}^k \max_{i \neq j} R_{ij} \quad (\text{Eq. 35})$$

The minimum value of DB is 0. Values near 0 indicate better clustering.

Maximum Silhouette Index Identification

As indicated, one of the major limitations of k -means clustering technique is the selection of the optimal number of clusters. For this reason, a range of a number of

clusters is implemented for each parameter being analysed. Subsequently, both Silhouette and Davies-Bouldin indices are estimated once the clusters have been identified for each number of clusters. As a Silhouette value of 1 indicates that the partitions are well-clustered, the number of clusters that obtains the maximum Silhouette value is selected, as it corresponds to the most appropriate number of clusters for each parameter.

Minimum Davies-Bouldin Index Identification

Similarly, as a Davies-Bouldin value of 0 indicates that the partitions are well-clustered, the number of clusters that obtains the minimum Davies-Bouldin value is selected. To consider the selected clusters as the feature to implement the first-order Markov chain, the number of clusters identified as the most appropriate may obtain both the maximum Silhouette value and the minimum Davies-Bouldin value of the entire number of clusters analysed. Otherwise, the clusters are rejected and the entire time series is used as the feature to implement the subsequent step, which is the univariate imputation technique through the implementation of the first-order Markov chain.

4.3.2. Univariate Imputation. First-order Markov Chain

Once the steady states are identified, the univariate imputation is performed. Univariate techniques impute values of a feature by only considering the feature being

analysed. By contrast, multivariate methods impute values of a feature by considering other features that are correlated with the feature being analysed. Thus, to apply any multivariate method, the completeness of the predictors' dataset needs to be guaranteed. For this reason, the univariate imputation step is implemented prior to the multivariate imputation step to impute the missing values that the predictors of a feature may contain.

For that purpose, first-order Markov chain model, which considers that each subsequent state hinges only on the preceding state, is applied in this inquiry to assess its effectiveness when imputing missing values of sensor data of marine systems. The procedure is presented hereunder.

Select Current Steady State Data to Fit the Model

Before implementing the first-order Markov chain, it is necessary to establish the data that fits the model. To impute the missing values, the model is generated every time that an instance of the feature being analysed presents a missing value. Thus, it is possible to determine the current steady state of the time series based on the clusters identified in the previous step. Hence, the preceding instances of the current steady state are considered to fit the model if the steady states' identification process is converged. Otherwise, the clusters are rejected, and the entire preceding instances of the time series are considered.

States' Determination

A collection of occurrences, x_t , indexed by time, are considered to identify, and impute missing values. It is determined that the occurrence at time t just hinges on the previous value and not on all values at times before t . Such a conception is known as discrete time stochastic process, and more precisely Markov process.

Thus, the time series values need to be clustered in a finite number of states so that the first-order Markov chain transition matrix can be estimated. To determine these states, the next approach is followed:

- i. The standard deviation, s , and the mean, \bar{X} , are estimated.
- ii. The maximum and the minimum values of the time series are determined.
- iii. On the basis of the mean value, \bar{X} , k standard deviations, s , are added or subtracted until the maximum and the minimum values are achieved.

$$S_T = \bar{X} \pm ks \tag{Eq. 36}$$

Transition Matrix Estimation

A discrete time stochastic process, $(X_n)_{n \in \mathbb{N}}$, which takes values in a finite set S , is considered to have the Markov property if the probability distribution of X_{n+1} at time

$n+1$ only hinges on the previous state X_n at time n , and not on all the past values of X_k for $k \leq n - 1$. Thus,

$$\begin{aligned} \mathbb{P}(X_{n+1} = j | X_n = i_n, X_{n-1} = i_{n-1}, \dots, Z_0 = i_0) & \quad (\text{Eq. 37}) \\ &= \mathbb{P}(Z_{n+1} = j | Z_n = i_n) \\ &= p(i, j) \end{aligned}$$

where $i_0, i_1, \dots, i_n, j \in S$. The probability $p(i, j)$ indicates the probability that the previous state i is followed by the current state j . All the possible transition probabilities of a process can be collected in a $r \times r$ matrix, where each (i, j) entry P_{ij} is $p(i, j)$,

$$\mathbf{P} = (P_{ij})_{1 \leq i, j \leq r} = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,r} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,r} \\ \vdots & \vdots & \ddots & \vdots \\ p_{r,1} & p_{r,2} & \cdots & p_{r,r} \end{pmatrix} \quad (\text{Eq. 38})$$

and that satisfies

$$\begin{aligned} 0 \leq P_{ij} \leq 1, & \quad 1 \leq i, j \leq r, \\ \sum_{j=1}^r P_{ij} = 1, & \quad 1 \leq i \leq r. \end{aligned}$$

Cumulative Transition Matrix Calculation

The cumulative probability transition matrix needs to be encountered by using Eq. 39 so that the data imputation process can be applied.

$$P_{ir} = \sum_{j=1}^r P_{ij} \quad (\text{Eq. 39})$$

Missing Value Imputation

The following procedure is applied to impute missing values:

- i. The preceding state is considered as the initial state.
- ii. By using a uniform random number generator, a random value, ξ , which varies between 0 and 1, is established.
- iii. The ξ value achieved is compared with the elements of the current state row of the cumulative probability transition matrix. Thus, if ξ is higher than the cumulative probability of the preceding state but lesser than or equal to the cumulative probability of the successive state the new state is adopted.
- iv. Finally, the missing values are imputed by using Eq. 40.

$$V = V_{\min} + \xi(V_{\max} - V_{\min}) \quad (\text{Eq. 40})$$

4.3.3. Multivariate Imputation

After the first-order Markov chain model is generated for every instance of each predictor that contains a missing value to impute, the predictors' dataset is completed.

Thus, the multivariate imputation step can be applied to impute the missing values of that feature. To that end, a comparative study is performed to assess the accuracy of a total of 16 machine learning and time series forecasting models. These models include: linear regression (Partial Least Squares regression, LASSO regression, Ridge regression, and ElasticNet regression), k -Nearest Neighbors, Support Vector Machines for Regression (with linear and RBF kernel), Neural Networks (with 1, 2, and 3 hidden layers), Vector Autoregressions (VAR), Decision Tree Regressors, and ensemble methods (Bagged Trees (with SVR and k -NN regressors), Random Forests, and AdaBoost). The accuracy of each model is assessed by estimating the Root Mean Square Error (RMSE). The model that presents the minimum RMSE value after the validation process implemented by applying the k -fold cross-validation technique is selected to train the model and, finally, impute the missing values.

k-fold Cross-validation Step

To estimate which model is the most appropriate to impute the missing values of a feature, k -fold cross-validation algorithm is utilised as a resampling technique to estimate the models' performances. The algorithm for implementing this technique is described hereunder.

- i. The dataset is divided into k sets of approximately equal size in a random manner.

- ii. One k set is identified as the test set, and thus it is not considered to train the model.
- iii. The remaining sets are utilised to train the model.
- iv. n missing values are generated at random in the test set and, subsequently, are imputed by applying the trained model.
- v. A metric is estimated to assess the model performance.
- vi. k set is reassigned with a set that has not been considered as a test set yet.
- vii. The steps 3, 4, and 5 are implemented again.
- viii. This process is repeated until all the sets has been considered as test sets.
- ix. The estimated metrics for each repetition are combined to obtain the overall performance assessment of the model.

Estimation of the Mean of the RMSE Estimated in Each Iteration of the Cross-validation Step

To assess the model performance, the Root Mean Square Error (RMSE) is estimated according to section *4.2.6. Evaluation*. As the RMSE is estimated for each iteration performed in the validation process, they are combined by estimating the mean to obtain the overall performance assessment of the model.

Selection of the Model That Presents the Minimum RMSE

The k -fold cross-validation technique is applied for each model considered in the study. Thus, it is possible to identify which model is the most appropriate to impute the missing values by selecting the model that presents the minimum RMSE.

Training of the Selected Model

The model identified as the most appropriate one is trained to impute the missing values. In total, there are 16 machine learning and time series forecasting models that are candidates to be trained. A comprehensive description of each of these models is presented in section *4.3.3. Multivariate Imputation*.

Missing Values Imputation

Finally, the missing values of the feature are imputed, and the entire dataset is completed. The process is iterative, and thus the imputation is performed a certain number of cycles. To assess the final imputation, the RMSE is again estimated.

4.4. Analysis of LSTM-based Variational Autoencoders for Regression for Data Imputation³

To promote and enhance the application of data imputation within the shipping industry with regards to DL methodologies, this section of the thesis methodology chapter suggests a novel framework for the analysis of variational autoencoders for regression modified by adding Long Short-Term Memory (LSTM) layers in both the encoder and decoder to consider the characteristics of time series data.

The proposed methodology is graphically represented in Fig. 4.4. The first phase refers to data pre-processing. Subsequently, the LSTM-VAE-based regressor analysed is introduced. To assess the imputation performance of such an approach, several contexts and metrics are considered. Finally, to evaluate if the analysed methodology can enhance other imputation techniques implemented within the shipping industry, a comparative study is introduced. The main contribution of this method is presented hereunder.

- Analysis of LSTM-VAE-based regressor for data imputation of marine systems sensor data.
- Development and implementation of a Data Assessment Imputation Framework (DAIF) for the evaluation of data imputation techniques.

³ The data imputation framework presented in this section has been already converted to a journal paper, and has been published in the Journal of Ship Research (Velasco-Gallego and Lazakis, 2022c).

- Introduction of a comparative study to analyse other widely utilised data imputation techniques within the shipping sector.

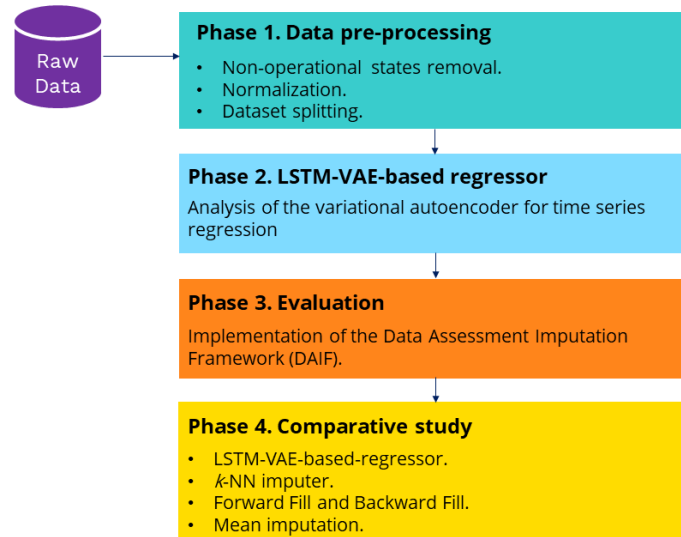


Fig. 4.4. Graphical representation of framework developed for the analysis of LSTM-based VAE regressor for data imputation.

4.4.1. Data Pre-processing

To avoid data pre-processing interfere significantly with the imputation performance, only critical data pre-processing steps are applied.

Non-operational states need to be adequately identified and discarded. Gaussian Mixture Models (GMMs) with Expectation-Maximization (EM) algorithm are analysed for such a purpose in this study. This probabilistic model considers that data

are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. EM is utilised for fitting the models and Bayesian information Criterion to evaluate the possible number of clusters. This step is performed through the application of the scikit-learn Python library (Pedregosa et al., 2011). Normalization is also applied so that the parameters lie between 0 and 1 values.

With regards to exploratory data analysis, correlation analysis is performed by estimating the Pearson's correlation coefficient to identify linear relationships between features. Data are split into training (80% of the entire dataset), validation (20% of the training dataset), and test (20% of the entire dataset) sets to avoid model overfitting. These values are considered as the standard ratios when training ML and DL methodologies.

4.4.2. LSTM-VAE-based Regressor Analysis

Variational autoencoders for regression are analysed in this study for performing the imputation task. This type of DL is an autoencoder that learns the parameters of a probability distribution. This which enables the model to be generative. The model is comprised of an encoder and a decoder. The encoder aims to learn both how to reduce the input dimensions and compress the inputs into an encoded representation. This compressed state, a.k.a latent space representation, presents the lowest possible dimensions of the inputs. By contrast, the decoder aims to learn how to reconstruct

the data contained in the latent space representation to reproduce the inputs as analogously as possible. The variational autoencoder is achieved by encoding the input as a distribution over the latent space. Thus, the autoencoder is regularised during the training process.

The methodology proposed by Zhao et al. (2019) is modified to enable the model to impute time-series data. Such methodology is constituted by an encoder with 2 intermediate layers of dimension (128, 32) with *tanh* as activation function. The resulting output is independently connected to two layers, the dimension of which is 8, to determine both the mean and the standard deviation of the latent representation. The regressor, which shared the intermediate layers of the encoder, is utilised to determine the mean and standard deviation for the predicted feature. Finally, the model is also constituted by the decoder. By utilising the latent representation as the input, the reconstruction is accomplished. Therefore, the architecture proposed considers a feedforward artificial neural network. Specifically, a multilayer perceptron, which does not deal with temporal dependencies, and thus does not consider the characteristics of time series data. Accordingly, the variational autoencoder based regression model is adapted to learn temporal dynamic behaviour through the implementation of LSTM, which is a type of recurrent neural network introduced by Hochreiter and Schmidhuber (1997). Fig. 4.5 presents a diagram of the VAE-based regression model highlighting the modification of both the encoder and decoder by adding LSTM layers. Specifically, the encoder is formed by 2 layers (128, 64) and *tanh*

activation function. Analogously, the decoder is constituted by 2 layers (64, 128) and \tanh activation function. The ratio of validation set has been set to 0.20. Adam optimizer has been applied to compile such a model. Subsequently, the model has been trained, setting the number of epochs to 100 and the batch size to 32. The prior hyperparameters have been defined based on prior experience and heuristic evaluation. This step is performed through the implementation of the Python libraries Tensorflow and Keras.

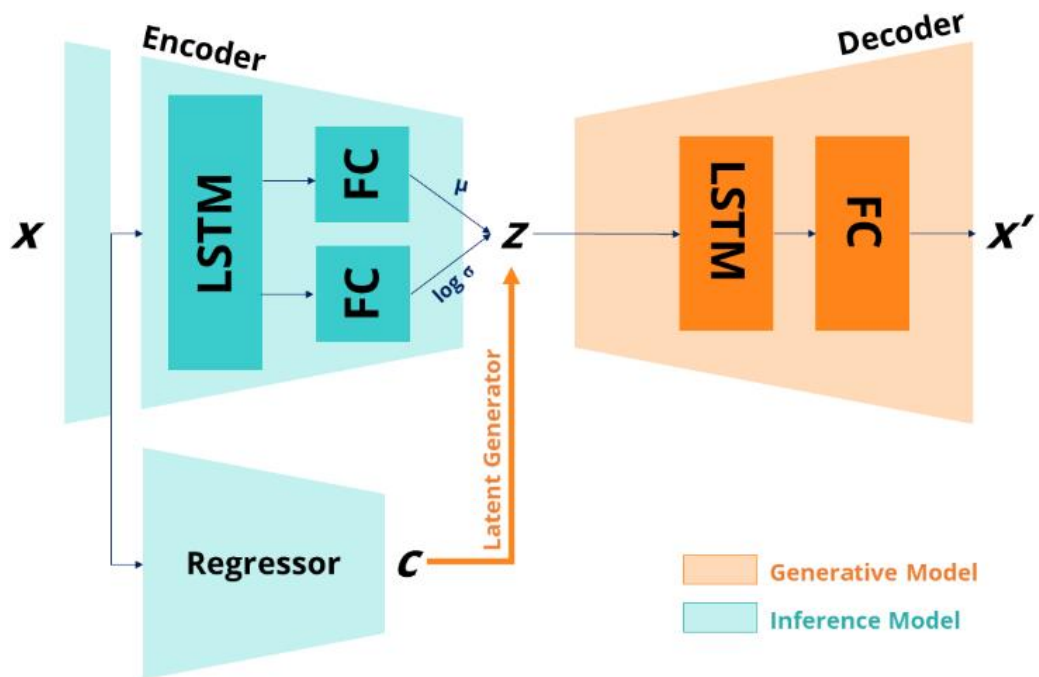


Fig. 4.5. Diagram of the VAE-based regression model modified to include LSTM layers in both encoder and decoder.

4.4.3. Evaluation

The procedure introduced for evaluating the imputation performance of the model is described hereunder.

- All the sequences of the target value obtained from the original dataset are considered as the input. The remaining features are considered as explanatory variables.
- n samples with different missing ratios (r_1, r_2, \dots, r_m) are generated. Each sequence contains values missing completely at random.
- The missing values are initially either masked or imputed to fit the VAE-regression model.
- The missing values are imputed by implementing the VAE-regression model.

Additionally, six metrics are estimated to determine the imputation performance of the imputation approach (RMSE, MSE, MedAE, MAE, Max. Error, and coefficient of determination (R^2)).

4.4.4. Comparative Study

A total of three models are considered for implementing a comparative study. This part of the methodology is utilised to validate the proposed method mainly introduced in section 4.4.2. *LSTM-VAE-based Regressor Analysis*. These models referred to imputation methodologies that have been considered precedingly by both academia

and industry. Thus, this section not only aims to validate the introduced methodology, but also analyse the performance of some of the most widely used imputation techniques within the shipping industry.

The first method considered is the mean imputation. This imputation method is usually implemented as it is easy to interpret, easy to apply, and the execution time is low. The second method is employed by Makridis et al. (2020) and consists of applying Forward Fill and, subsequently, Backward Fill algorithms. The last model analysed is the k -NN. k -NN has been precedingly analysed by Cheliotis et al (2019).

4.5. A Novel Framework for the Identification of Steady States⁴

The novel methodology is graphically represented in Fig. 4.6. The first step refers to the pre-processing of the input time series data, in which the overall time series is sectioned into sequences by applying the sliding window algorithm. Subsequently, each sequence is transformed into an image by estimating the transition matrix obtained from the implementation of the first-order Markov Chain. To adequately determine the different regions identified in each of the images, connected component analysis is

⁴ The operational states' identification framework presented in this section has been already converted to a journal paper, and has been published in the Applied Ocean Research journal (Velasco-Gallego and Lazakis, 2022e).

conducted and, consequently, post-processing is performed on the outcoming images to transform them into sequences. As the states are labelled per sequence, results from the preceding phase need to be also pooled to achieve the input time series with the resulting labels that specify the different steady states identified. The main contributions of this section are summarised as follows:

- Development of a time series approach for the identification of operational steady states of marine machinery.
- Application of the first-order Markov chain to encode time series data into images.
- Consideration of connected component analysis to identify steady and unsteady states.

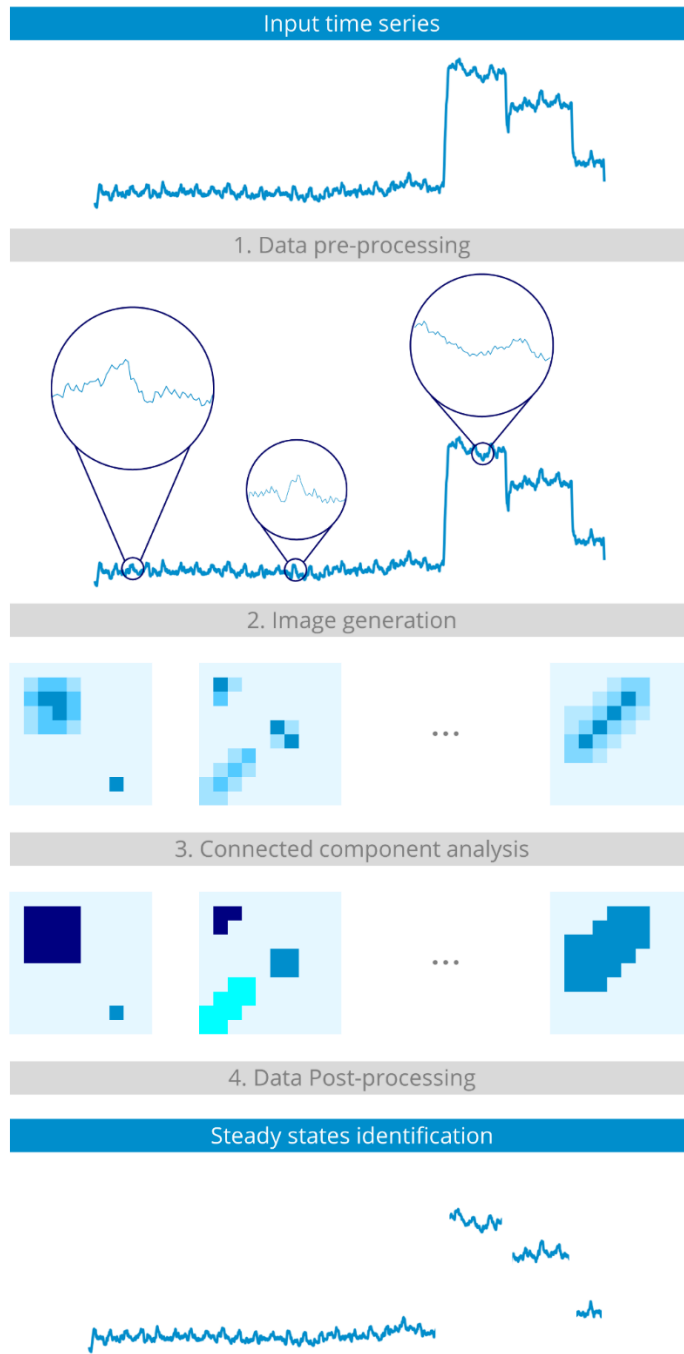


Fig. 4.6. Novel framework for the identification of steady states.

4.5.1. Data Pre-processing

Data imputation is performed due to the missing values that are usually encountered. Moreover, data denoising is also performed to assess if the identification of steady states is enhanced. Exponentially Weighted Moving Average (EWMA) is applied accordingly. Finally, the time series is divided into sequences through the application of the sliding window algorithm.

4.5.2. Image Generation

To adequately identify the different steady states all along the analysed data set, the input time series is transformed into an image by implementing the first-order Markov chain. The image generated refers to the transition matrix, which is estimated as represented in section *4.3.2. Univariate Imputation. First-order Markov chain.*

4.5.3. Connected Component Analysis

By considering the transition matrix estimated in the preceding step as a collection of discrete cells, a.k.a., pixels, the transformation from time series to image is achieved. Thus, each pixel is associated with a pixel value, which lies between 0 and 1 (inclusive) and refers to the probabilities formerly estimated.

In turn, to facilitate the implementation of connected component analysis, the image outlined is converted to a binary one with only two possible intensity values. Such a conversion is performed according to (Eq. 41), in which the binary image is obtained by classifying the different pixel values into either 0, if the probability associated with the pixel is equal to 0, or 1, otherwise. Thus, those pixels that present information about a transition between states can be efficiently identified.

$$P_{ij} = \begin{cases} 0, & \text{if } P_{ij}=0 \\ 1, & \text{otherwise} \end{cases} \quad (\text{Eq. 41})$$

By applying this conversion, the distinct transition clusters presented within the image can be labelled. Accordingly, pixel connectivity is analysed, which characterises the relationship between pixels. To consider that two neighbouring cells are connected, they must present the same pixel value. For this inquiry such a connectivity is formulated by applying the 4-neighbours adjacency criterion (see Fig. 4.7). Thus, the notation of neighbourhood for such a case is expressed hereunder.

$$N_4(p) = \{(x+1, y), (x-1, y), (x, y+1), (x, y-1)\} \quad (\text{Eq. 42})$$

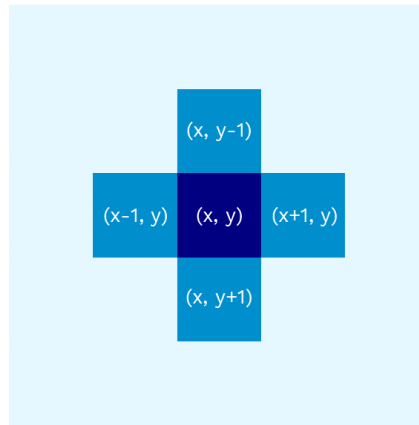


Fig. 4.7. Graphical representation of the 4-connected neighbourhood.

All possible neighbouring pixel connectivity is evaluated to determine the distinct sets of connected pixels, a.k.a. connected components. Therefore, the last step of this phase, named connected components labelling, is achieved, in which the different connected components are clustered to identify the different states, and in turn determine those that only refers to steady states. A graphical representation of such a phase is expressed in Fig. 4.8.

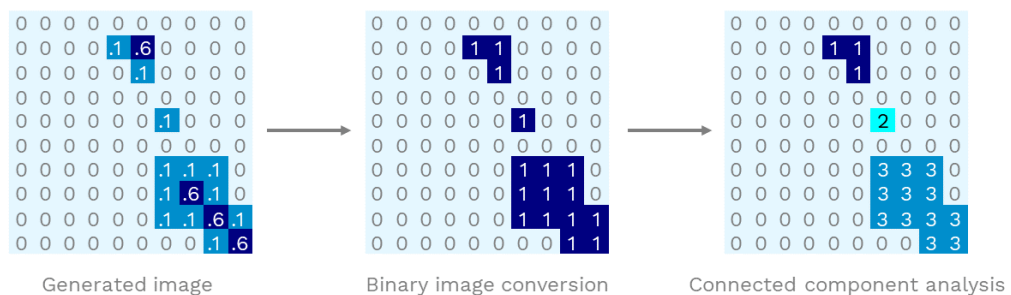


Fig. 4.8. Connected component analysis phase representation.

4.5.4. Data Post-processing

As the results outlined in the preceding section are structured in the form of images, the pixel values of which pertaining to the labels obtained from the connected component analysis, inverse transformation needs to be applied to convert such images into distinct time series sequences. Thus, each sequence instance is associated with a temporal label, as different timestamps can be contained in more than one resulting sequence. Thus, such sequences need to be pooled to obtain a unique label per instance of the input time series. To that end, the following approach is applied: if all the temporal labels for a particular instance present the same value, that instance is part of a steady state. Otherwise, if the temporal labels associated with a specific instance differ in regard to their respective values, it is assumed that the instance could not be related to a particular state, and thus such an instance cannot be considered for further analysis. A graphical representation of such a process is described in Fig. 4.9.

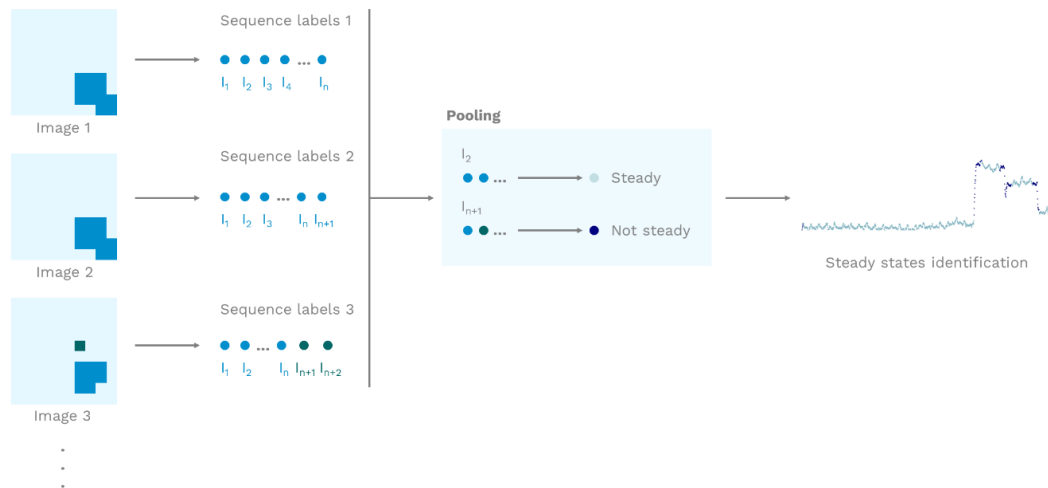


Fig. 4.9. Graphical representation of the post-processing phase.

4.6. Chapter Summary

This chapter presented the novel methodological components of the data pre-processing module. Two main phases have been analysed due to their lack of analysis and formalisation, and its criticality to ensure data quality. These are the data imputation and the operational steady states' identification phases. To address the challenge of data imputation, a total of three distinct methodologies have been proposed. The first one, refers to a comparative methodology of both univariate and multivariate and imputation techniques. The second method introduced is a novel hybrid imputation framework constituted by a total of three main stages: 1) operational states' identification through the implementation of the k -means algorithm, 2) univariate imputation by applying the first-order Markov chain, and 3) multivariate imputation through the integration of the multivariate comparative analysis performed in the preceding method. To finalise the data imputation study, a DL method was also analysed, which was identified as LSTM-based VAE for regression. With regards to the operational steady states' identification task, a time series imaging approach by implementing the first-order Markov chain in tandem with connected component analysis was proposed.

Chapter 5

Methodology: Part III. The Diagnostic and Predictive Analytics Modules

5.1. Chapter Overview

The novel methodologies introduced in both diagnostic and predictive analytics module of the maintenance analytics framework are presented in this chapter.

5.2. Diagnostic and Predictive Analytics Module Overview

The diagnostic analytics module aims to determine the current health state of the marine systems. Accordingly, two main stages are introduced: fault detection, and fault identification. In addition, the predictive analytics module is introduced. This module aims to determine the future health state of the marine systems. Accordingly, the Remaining Useful Life (RUL) prediction framework is introduced. The overview of these modules is graphically represented in Fig. 5.1. Each of the stages of the modules are further expanded in the subsequent sections of this chapter.

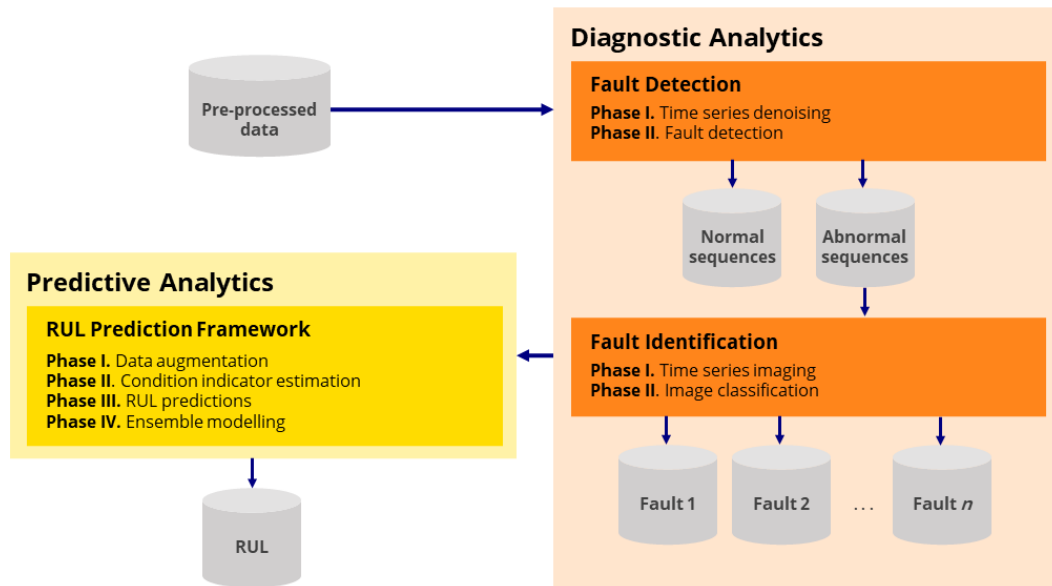


Fig. 5.1. Overview of the diagnostic and predictive analytics modules.

5.3. Diagnostic Analytics

5.3.1. Fault Detection⁵

The proposed methodology is graphically represented in Fig. 5.2. The first step refers to time series denoising, which is applied through the implementation of LSTM-based VAE NN. Accordingly, the NRMSE between the input time series and the generated time series is determined. By estimating such a coefficient for each generated time series, the NRMSE matrix is obtained. Therefore, if image thresholding is applied by considering multi-level Otsu's thresholding method, the anomalousness of each instance of all the analysed sequences can be determined, thus labelling the behaviour

⁵ The fault detection framework presented in this section has been already converted to a journal paper, and has been published in the Expert Systems with Applications journal (Velasco-Gallego and Lazakis, 2022d).

identified at each time step. The training process of such an approach are performed offline, whereas the remaining steps are performed online. The two processes are described next in more detail. The main contributions of this section are summarised hereunder.

- The introduction of a LSTM-based VAE as part of the fault detection methodology.
- The utilisation of image thresholding techniques for the detection of anomalies through the estimation of the NRMSE matrix and the application of multi-level Otsu's thresholding.

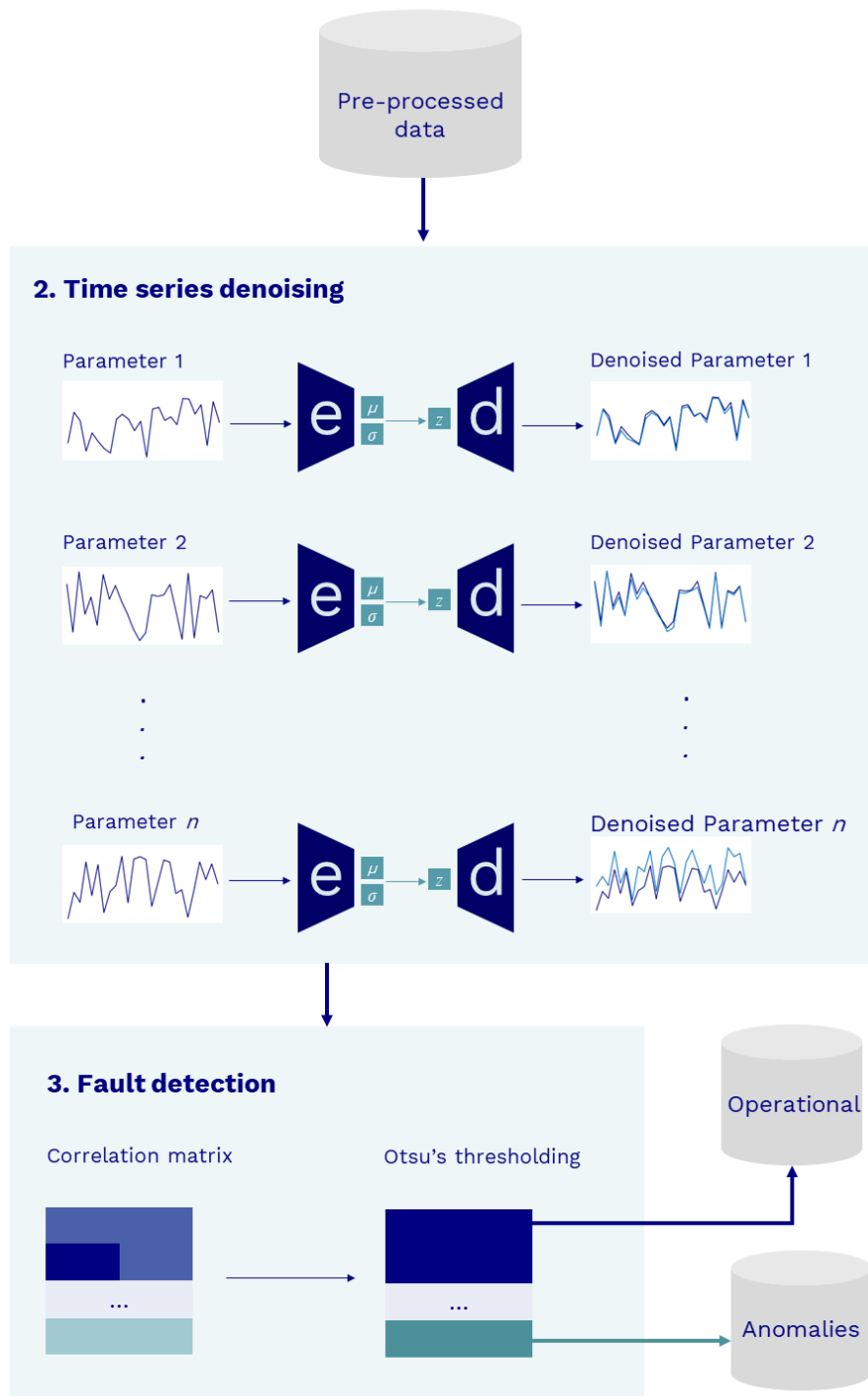


Fig. 5.2. Graphical representation of the fault detection framework.

5.3.1.1. Time Series Denoising

VAE, developed by Kingma and Welling (2013), is a generative algorithm capable of modelling the distribution of the data. This is a modification of an autoencoder that learns the parameters of a probability distribution. The model is constituted by a probabilistic encoder, which aims to learn both how to reduce the input dimensions and compress the inputs into an encoded representation. This compressed state, a.k.a. latent space representation, presents the lowest possible dimensions of the inputs. Subsequently, the decoder is utilised to learn how to reconstruct the data contained in the latent space representation to reproduce the inputs as analogously as possible. The architecture of such a model is described in Fig. 5.3.

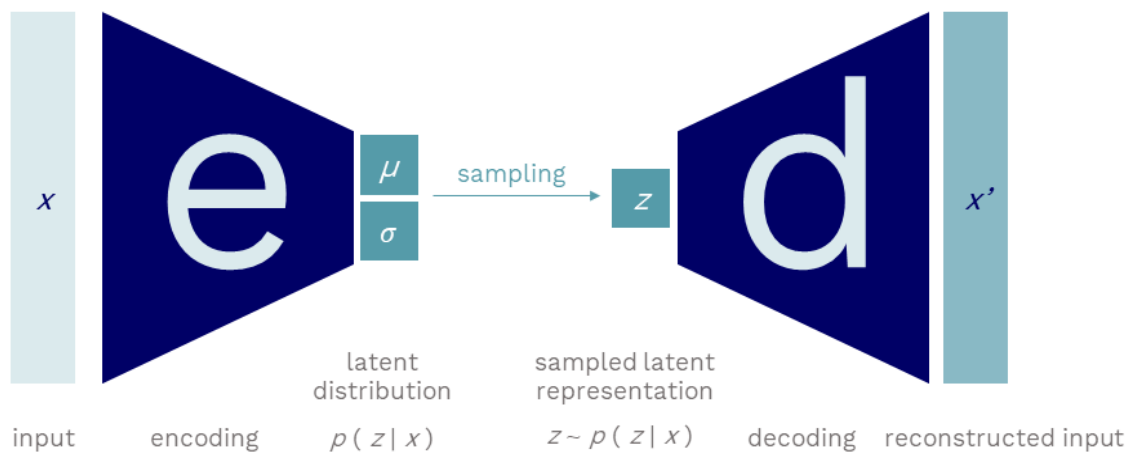


Fig. 5.3. VAE architecture.

The loss function being minimised is constituted by the reconstruction loss, which aims to ensure the efficient performance of the encoder-decoder arrangement, and the regularisation loss. The latter is determined by estimating the Kullback-Leiber divergence between the approximate posterior and prior latent variable z .

To consider the temporal dependences of sensor data of marine systems, the VAE approach is combined with LSTM in both the encoder and the decoder. LSTM is a type of Recurrent Neural Network (RNN) introduced by Hochreiter and Schmidhuber (1997) that learns long-term dependencies. As described in Fig. 5.4, the core component of such a network is the memory cell, which consists of a cell state vector and gating units, the latter regulating the information flow into and out of the memory, to maintain its state over time. Specifically, a total of three non-linear gating units are introduced, which regulate and protect the cell states. By introducing these gating units, information can easily flow throughout the entire chain, thus eliminating the gradient vanish problem whilst learning the long-term dependencies.

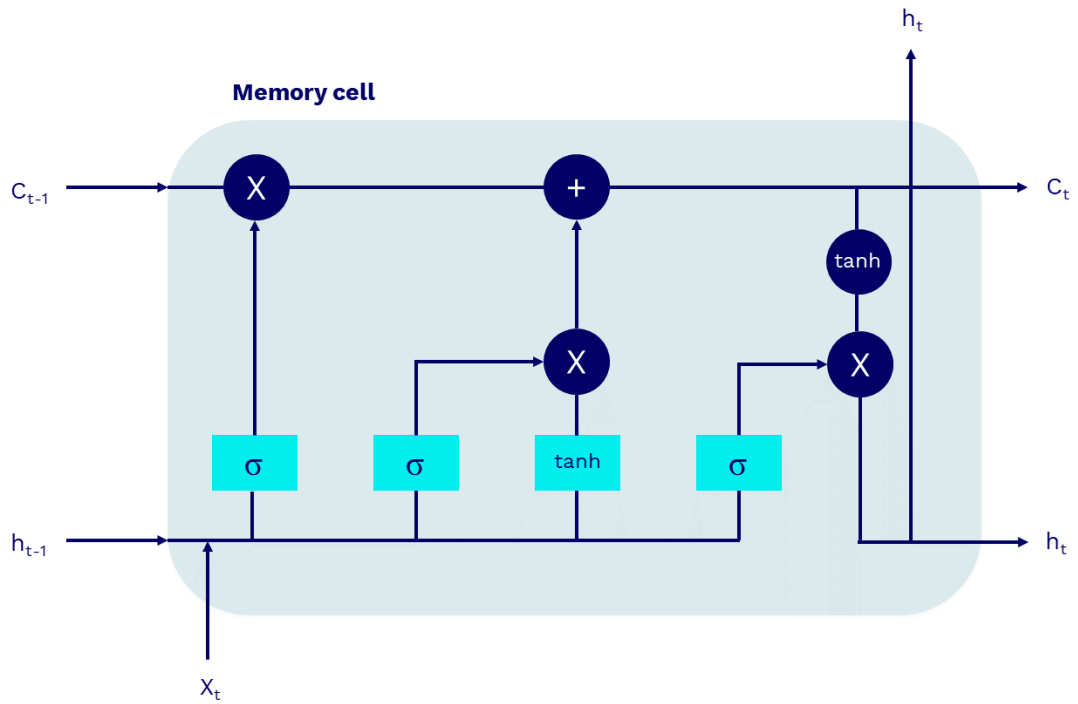


Fig. 5.4. LSTM cell architecture.

This step is performed through the implementation of the Python libraries Tensorflow (Abadi et al., 2016) and Keras (Chollet et al., 2015).

5.3.1.2. Fault Detection

By performing the preceding step, that is time series denoising, it is assumed that anomalous sequences will not be able to be properly reconstructed, and thus the resulting sequence will not be analogous to the observed one. Accordingly, the Normalised Root Mean Square Error (NRMSE) is estimated as presented in Eq. 43.

$$\text{NRMSE} = \frac{\text{RMSE}}{\bar{y}}, \quad (\text{Eq. 43})$$

where RMSE is the Root Mean Square Error \bar{y} and is the mean of the observed values presented in the subsequence. The NRMSE has been utilised instead of the RMSE to facilitate the comparison of other parameters that present distinct scales. The RMSE metric has been estimated as defined in *4.2.6. Evaluation*.

Therefore, if such a value is estimated for every subsequence, a NRMSE matrix is achieved. Thus, as anomalous pixels will present distinct intensities over operational values, they can be adequately detected by applying image thresholding. This is a phase that segments the image into significantly distinct and non-overlapping homogenous regions. For this inquiry, such a segmentation is performed by analysing the pixels' intensity and by considering a threshold-based technique. Specifically, the multi-level Otsu's method is applied. Such a method performs image thresholding by proposing a criterion for maximising the between-class variance of pixel intensity. If the NRMSE matrix is considered as a 2D greyscale intensity function, it can be established that such an image contains a total of N pixels with grey levels from 1 to L . The pixels' number with grey level i is denoted as f_i . Thus, the probability of grey level i in an image can be defined as expressed in Eq. 44.

$$p_i = \frac{f_i}{N} \quad (\text{Eq. 44})$$

By considering a total of $M-1$ thresholds, $\{t_1, t_2, \dots, t_{M-1}\}$, which segments the initial image into M classes: C_1 for $[1, \dots, t_1]$, C_2 for $[t_1+1, \dots, t_2]$, ..., C_i for $[t_{i-1}+1, \dots, t_i]$, ..., and C_M for $[t_{M-1}+1, \dots, L]$, the optimal thresholds $\{t_1^*, t_2^*, \dots, t_{M-1}^*\}$ are selected by maximising the between-class variance, σ_B^2 , as described hereunder.

$$\{t_1^*, t_2^*, \dots, t_{M-1}^*\} = \arg \max \{\sigma_B^2(t_1, t_2, \dots, t_{M-1})\}, \quad 1 \leq t_1 < \dots < t_{M-1} < L$$

where

$$\sigma_B^2 = \sum_{k=1}^M \omega_k (\mu_k - \mu_T)^2 \quad (\text{Eq. 45})$$

with

$$\omega_k = \sum_{i \in C_k} p_i, \quad (\text{Eq. 46})$$

$$\mu_k = \sum_{i \in C_k} \frac{ip_i}{\omega(k)}. \quad (\text{Eq. 47})$$

The ω_k in Eq. 49 relates to the zeroth-order cumulative moment of the k th class C_k , and the numerator in Eq. (50) refers to the first-order cumulative moment of the k th class C_k ; that is,

$$\mu(k) = \sum_{i \in C_k} ip_i. \quad (\text{Eq. 48})$$

To adequately select the optimal number of classes, GMM with an EM algorithm is implemented.

5.3.2. Fault Identification⁶

The proposed methodology is graphically represented in Fig. 5.5. The first phase refers to the encoding of time series sequences into images through the application of the first-order Markov chain model. Subsequently, to perform the fault classification task, the second phase is implemented, in which image classification is applied by applying the deep learning architectures ResNet50V2 and CNN. The contribution of this phase can be summarised as follows:

⁶ The fault identification framework presented in this section has been already converted to a journal paper, and has been published in the Ocean Engineering journal (Velasco-Gallego and Lazakis, 2022b).

- The introduction of an image classification approach for performing fault classification.
- The analysis of the first-order Markov chain model as a time series image technique.
- The analysis of CNN and ResNet50V2 as image classifiers.

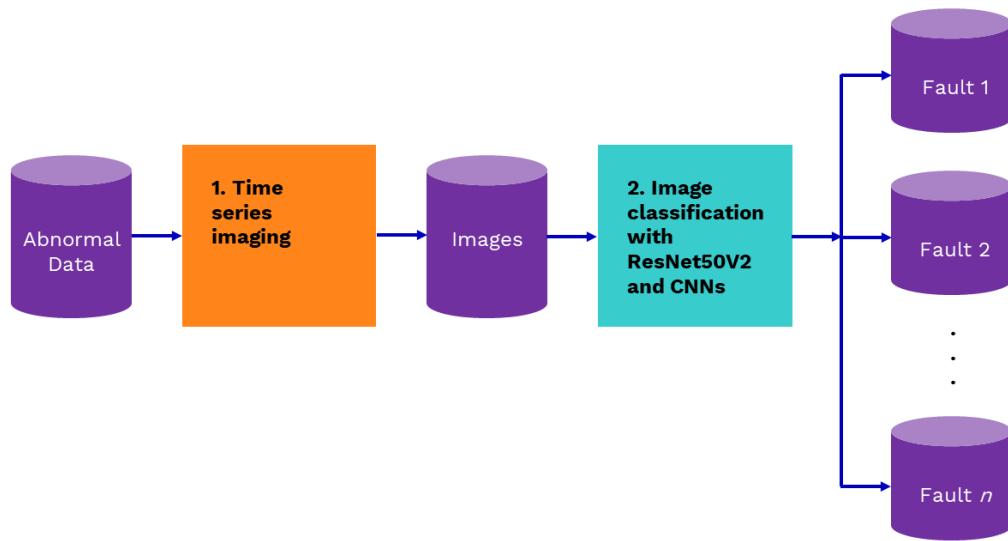


Fig. 5.5. Graphical representation of the proposed methodology.

5.3.2.1. Time Series Imaging

The time series are encoded into images through the estimation of the transition matrix of the first-order Markov chain as defined in *4.5.2. Image Generation*.

5.3.2.2. Image Classification with ResNet50V2 and CNNs

The first method considered as image classifier is the ResNet50V2. ResNet50V2 is a type of deep residual network proposed by He et al. (2016b). Deep residual networks, a.k.a. ResNets (He et al., 2016), consists of many stacked “Residual Units”, which can be expressed in a general form as presented in Eq. 54:

$$\begin{aligned} y_l &= h(x_l) + \mathcal{F}(x_l, W_l), \\ x_{l+1} &= f(y_l), \end{aligned} \tag{Eq. 49}$$

where x_l and x_{l+1} are input and output of the l -th unit, and \mathcal{F} is a residual function. $h(x_l) = x_l$ is an identity mapping and f is a ReLU (Nair and Hinton, 2010) function. The essence of ResNets is to learn the additive residual function \mathcal{F} respecting $h(x_l)$ by attaching an identity skip connection or “shortcut”. ResNet50V2 is an enhancement of ResNet50 in which a new residual unit has been introduced to both facilitate an easier training and enhance generalisation.

Due to the lack of fault data within the sipping industry, the ResNet50V2 network has been pretrained by utilising the popular dataset ImageNet, which presents more than 1000-class single labels (Russakovsky et al., 2014).

The second analysed method is the CNN, which is a type of feedforward Artificial Neural Networks (ANNs) that is constituted by a feature extraction step and either a classification or a regression task. As the main objective of this study is to develop an approach for fault classification, only the classification task is considered in this context.

The first stage, feature extraction, is comprised of both convolutional layers and pooling layers. The convolutional layer is usually also referred to as the main block of CNN models. This consists of a set of filters, which are learnt throughout the training process, that convolve with the image and generate a feature map. Specifically, the filter slides over the entire image so that the dot product between each element of both the filter and the input can be estimated at every spatial position. To reduce the dimension of the resulting feature map, a pooling layer is usually introduced after the application of a convolutional layer. Although a loss of information can be perceived by applying such layers, they assist in averting overfitting and reducing the computational cost. The pooling task is performed by sectioning the input into non-overlapping rectangular subregions so that information from each subregion can be extracted. For this inquiry the max pooling layer is implemented.

The second stage refers to the classification task, implemented through the utilisation of fully connected layers. Such layers apply high-level logical operations by considering

features from preceding layers. The output of the final layer is a n dimensional vector, n being the total number of classes being considered.

This step is performed by implementing of Python libraries Tensorflow (Abadi et al., 2016) and Keras (Chollet et al., 2015).

5.4. Predictive Analytics

5.4.1. Remaining Useful Life (RUL) Prediction Framework⁷

A graphical representation of the proposed RUL methodology is presented in Fig 5.9. Due to the lack of degradation data, the first stage refers to the data augmentation phase. This module is comprised of a total of two phases. The first phase aims to generate synthetic operational data from the observed time series. Thus, the number of sequences to be used for the training of the different DL models is increased. The second step aims to transform the simulated operational data into degradation sequences through the implementation of an exponential model with Brownian motion. The second step consists of estimating the condition indicator of each of the simulated sequences through the implementation of the piecewise linear algorithm. Step three performs the training of the three proposed models: Markov-CNN, 1D-CNN, and LSTM. As each of the models will provide different predictions, these need to be

⁷ The RUL prediction framework presented in this section has been already converted to a journal paper, and has been submitted in the Engineering Applications of Artificial Intelligence journal (Velasco-Gallego and Lazakis, 2022a).

integrated somehow. Accordingly, the fourth and final step is implemented, which considers the weighted average ensemble approach to obtain the final RUL prediction. The above steps will be described in more detail in the following paragraphs. A summary of the main contributions of this section is presented hereunder.

- The addition of a degradation simulation module prior to the RUL prediction approach to perform data augmentation, and thus deal with the lack of fault data of marine machinery systems. This comprises a simulation module with both first-order Markov Chain model and an exponential model with Brownian motion approach.
- The application of an ensemble method as a RUL prediction approach. A total of three distinct DL architectures are considered: Markov-CNN, 1D-CNN, and LSTM.
- The analysis of time series imaging for the prediction of RUL. A novel method is considered to transform the analysed time series into an image through the implementation of the first-order Markov chain. Once all the sequences of the time series have been transformed into images by applying such a method, a CNN is introduced to perform the RUL prediction task.

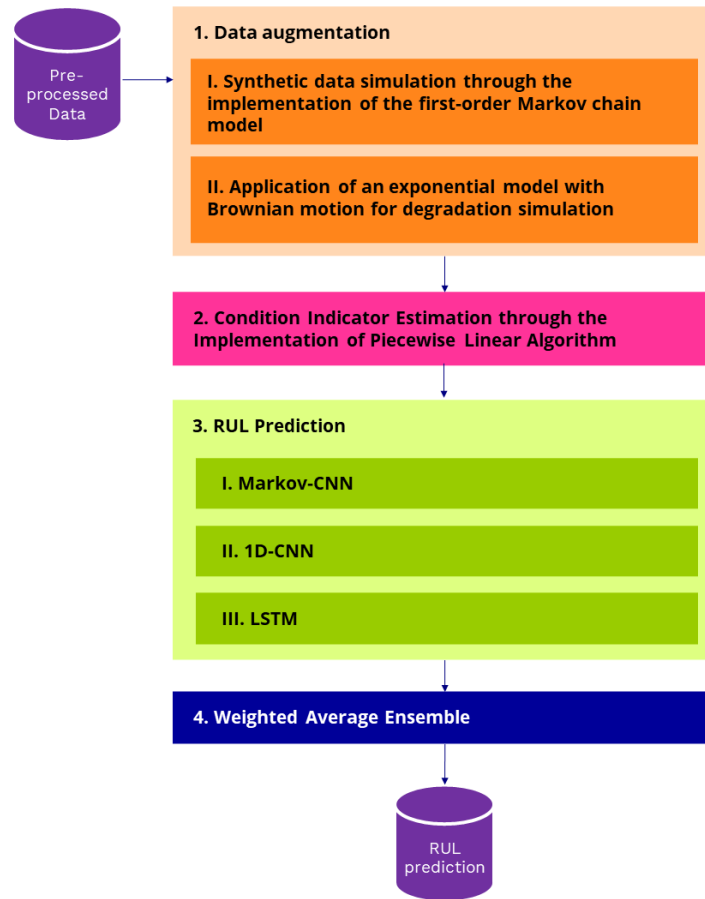


Fig. 5.6. Graphical representation of the predictive analytics module.

5.4.1.1. Data Augmentation

This is comprised mainly of two parts. The first part aims to generate synthetic time series by considering sensor data collected from marine machinery. The model implemented in this inquiry is the first-order Markov chain, as it has been presented in 4.3.2. *Univariate imputation. First-order Markov Chain*. However, the methodology has been slightly modified to generate synthetic sequences instead of imputing missing values.

The second part of this approach refers to the simulation of the degradation patterns. Accordingly, an exponential model with Brownian motion is considered due to their effective universality in machinery to reflect the characteristics of accelerated fault degradation in engineering (Li et al., 2021).

5.4.1.2. Condition Indicator Estimation through the Implementation of the Piecewise Linear Algorithm

The piecewise linear labelling approach is implemented to obtain the condition indicator based on the simulated degradation pattern, as this approach has been accepted as the most effective method for such a task (Gribbestad et al. (2021)). Therefore, the condition indicator is initially constant, as no symptoms of degradation are experienced until either a certain level is achieved, or a fault has arisen. On that occasion, the condition indicator decreases until the machinery is no longer operating due to a failure occurrence, thus achieving the minimum value possible of such an indicator, which is 0. In this study it is considered that the condition indicator is the normalised RUL.

5.4.1.3. Time Series Imaging through the Implementation of the First-order Markov Chain Model

As expressed, precedingly, this study also analysis the first-order Markov chain model by estimating the transition matrix for image representation. This part of the methodology is performed as defined in section 4.5.2. *Image Generation*.

5.4.1.4. Remaining Useful Life (RUL) Prediction

Markov-Convolutional Neural Network (CNN)

The architecture of the Markov-CNN was previously introduced in section 5.3.2.2. *Image classification with ResNet50V2 and CNNs*. However, such an architecture has been modified to perform a regression task instead of a classification one. Accordingly, the last Fully Connected Layer (FCL) has been modified to present as output a unique numeric value, which refers to the current RUL prediction. A graphical representation of the Markov-CNN architecture for regression is presented in Fig. 5.10.

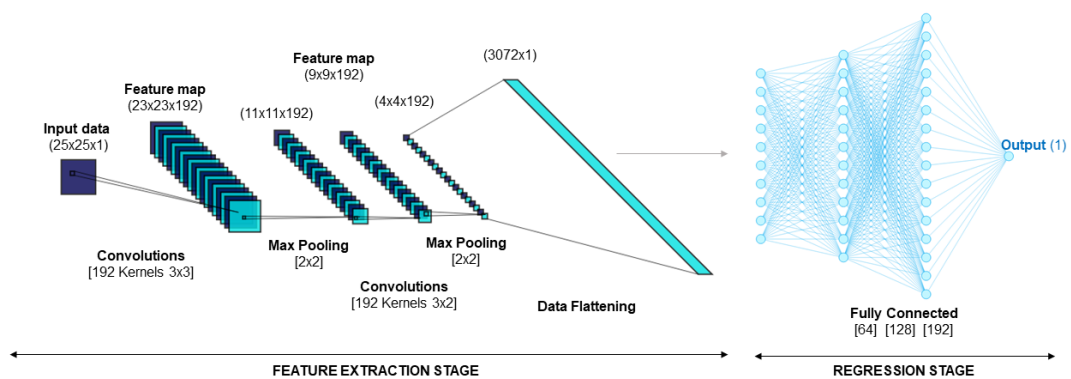


Fig. 5.7. Graphical representation of the Markov-CNN architecture for regression.

1D-Convolutional Neural Network (CNN)

Analogous to the preceding architecture, a 1D-CNN model is presented in this subsection. However, unlike Markov-CNN, the input of this model is a sequence of the original time series of sequence length n ($n \times 1$ (a 1D architecture is considered for this context) $\times 1$ (a univariate approach is considered for this inquiry)). A graphical representation of this type of architecture is described in Fig. 5.11. Although some indications of the hyperparameters utilised can be perceived in this figure, a more detailed explanation is provided in the subsequent section.

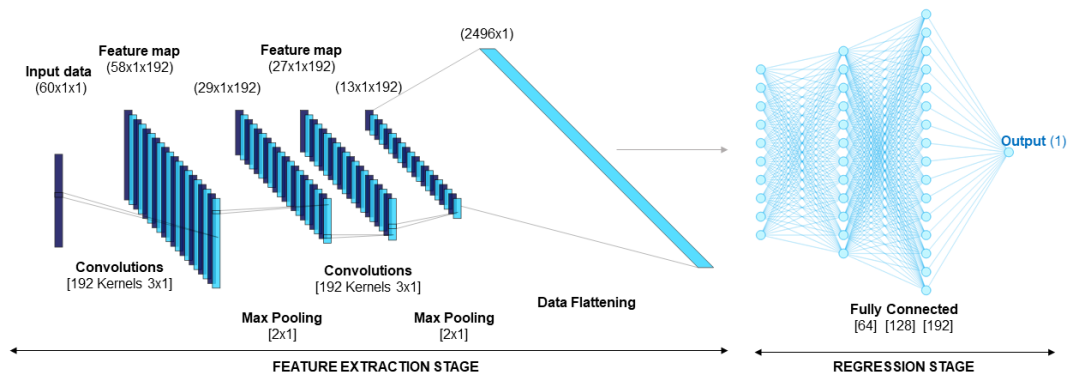


Fig. 5.8. Graphical representation of the 1D-CNN architecture.

Long Short-Term Memory (LSTM)

LSTM has provided promising results for predicting RUL of marine machinery, as stated by Han et al. (2021). Therefore, a continuous analysis of such a deep learning model is worth considering. As precedingly stated, LSTM is a type of RNN introduced by Hochreiter and Schmidhuber (1997) that learns long-term dependencies. The memory cell is its core component, which consists of a cell state vector and gating units, the latter regulating the information flow into and out of the memory so that its state can be maintained over time.

5.4.1.5. Weighted Average Ensemble

To integrate the resulting predictions obtained through the implementation of each model established in this inquiry, a simple pooling approach is considered by applying a weighted arithmetic mean. The estimation of such weights is performed through the application of the grid search optimisation algorithm, which considers a set of combinations that satisfies Eq. 50. The pooling operation has been performed according to Eq. 51.

$$w_1 + w_2 + \dots + w_n = 1 \quad (\text{Eq. 50})$$

$$P_{\text{final}} = \sum_{i=1}^n w_i P_i, \quad (\text{Eq. 51})$$

where w_i is the selected weight for the approach i , and p_i is the RUL prediction of that same approach. For this inquiry, n , which refers to the number of considered approaches, is three, thus only considering the ones presented in the preceding sections (Markov-CNN, 1D-CNN, LSTM).

5.5. Chapter Summary

This chapter presented the novel methodological components of the diagnostic analytics module. This module is comprised of fault detection and fault identification. For the fault detection task, a LSTM-based VAE in tandem with multi-Otsu's thresholding was introduced. With regards to the fault identification task, both time series imaging and image classification models were analysed. Specifically, ResNet50V2 and CNN were considered due to its capability of extracting deep features.

With regards to the predictive analytics module, a RUL prediction framework was presented, which was comprised of an ensemble model constituted by a total of three models: Markov-CNN, 1D-CNN, and LSTM.

Chapter 6

Case Studies and Results: Part I. The Data Pre-processing Module

6.1. Chapter Overview

Having formalised the novel maintenance analytics framework for marine systems based on the challenges and gaps identified within the maritime sector, a series of case studies are introduced to validate its performance. In total, 4 case studies are introduced in this chapter, each of one referring to one of the novelties presented in the data pre-processing module. It is worth highlighting that to assess the generalisation capabilities of the introduced modules and methodologies, each case study is unique and considers different time series sequences, even though some case studies may refer to the same system.

6.2. Case Study 1. Comparative Methodology for Data Imputation⁸

This case study aims to validate the methodology introduced in section 4.2.

Comparative Methodology of Univariate and Multivariate Imputation Techniques. For

⁸ The data imputation framework presented in this section has been already converted to a journal paper, and has been published in the Ocean Engineering journal (Velasco-Gallego and Lazakis, 2020).

this case study, a DMD-MAN B&W 6S50MC-C main propulsion engine of a cargo vessel is considered. This is a camshaft controlled two-stroke engine, which utilises super long stroke to bore ratio, constituted by a total of 6 cylinders with 50 centimetres diameter pistons. In total, seven parameters are analysed: 1) rotational speed, 2) power, 3) fuel flow rate of the main engine, 4) inlet pressure of the lubrication oil system, 5) inlet pressure of the jacket cooling water system, 6) turbine lubricating oil inlet pressure of the turbocharger, and 7) scavenging air pressure of the scavenging air receiver (please see Table 6.1 for a more comprehensive description).

Table 6.1. Main engine system monitored parameters.

	Parameter	Units
Main Engine	Rotational Speed	r/min
	Power	kW
	Fuel Flow Rate	t/hr
Lubrication Oil System	Inlet Pressure	bar
Jacket Cooling Water System	Inlet Pressure	bar
Turbocharger	Turbine Lubricating Oil Inlet Pressure	bar
Scavenge Air Receiver	Scavenging Air Pressure	bar

A total of 2,000 instances that refer to steady operational states of machinery are analysed. These instances, which have been recorded in a 1-minute frequency basis, are graphically represented in Figs. 6.1 – 6.7. As perceived in Fig. 6.1, a total of four major operational steady states are identified when the main engine rotational speed is considered. The first, and largest, steady operational state initiates at the first instant and persists over half of the recorded time. The values of this state are

stabilised around 105.0 r/min. Then, a slight adjustment is perceived, thus the rotational speed decreasing to approximately 102.5 r/min. After roughly 300 minutes, a sudden abrupt change facilitates the increment of the revolutions, and the maximum value perceived throughout the time series is achieved. This value is greater than 110.0 r/min. This state remains for various minutes. Subsequently, a decrease of the rotational speed is presented in three slight phases. This is when the minimum state is recorded, the values of which are lesser than 100.0 r/min. Such a state is initiated at 1,500 minutes and remains constant until the end of the series. The abrupt changes between steady operational states refer to small adjustments that are applied due to the contractual agreements between the charterer and the shipowner in relation to the vessel speed and the fuel oil consumption per day.

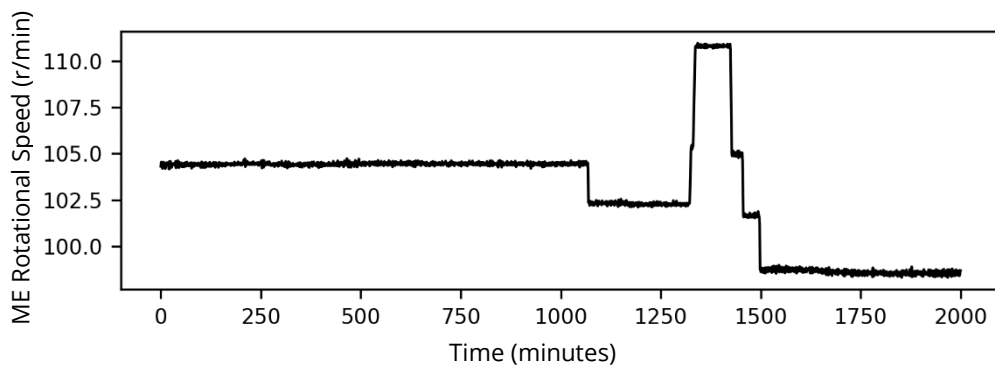


Fig. 6.1. Time series plot of the main engine rotational speed.

A similar evolution as the main engine rotational speed is perceived when the main engine power, the main engine fuel flow rate, and the scavenging air pressure of the scavenge air receiver system are considered (Figs. 6.2 – 6.4).

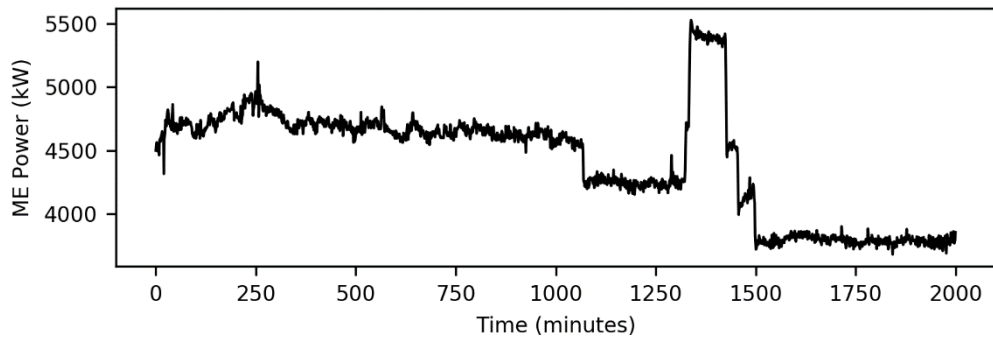


Fig. 6.2. Time series plot of the main engine power.

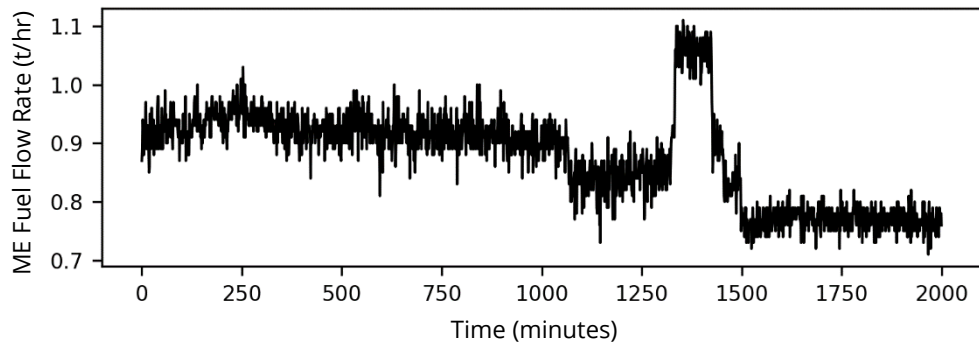


Fig. 6.3. Time series plot of the main engine fuel flow rate.

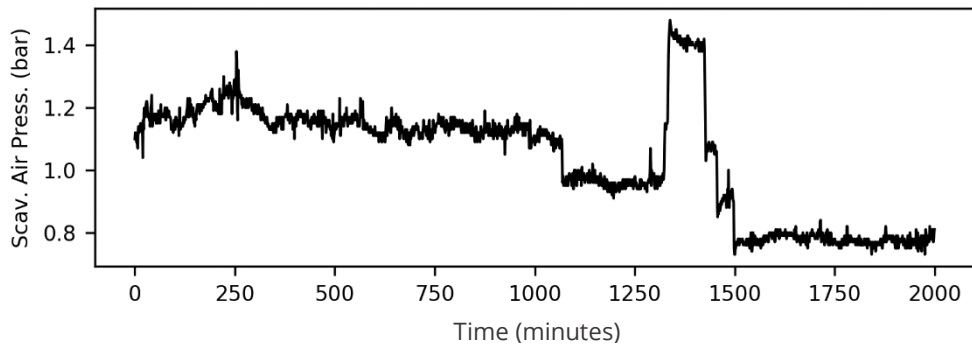


Fig. 6.4. Time series plot of the scavenging air pressure of the scavenge air receiver system.

With regards to the lubrication oil inlet pressure, only two main steady states are perceived. The first one initiates at the first instant and persists for more than 1,250 minutes. In this first state the values are stabilised between 2.39 and 2.40 bar. Then, a slight adjustment facilitates a decrease of the inlet pressure to approximately 2.35 bar (Fig. 6.5).

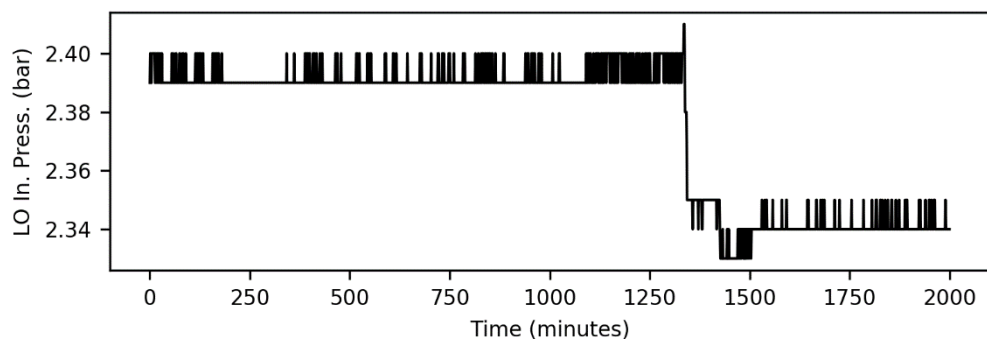


Fig. 6.5. Time series plot of lubrication oil inlet pressure.

By contrast, both the inlet pressures of both the jacket water cooling and the turbocharger lubrication oil systems present only a unique state (Fig. 6.6 and Fig. 6.7). In the case of the jacket water cooling system inlet pressure the values fluctuate between 3.50 and 3.65 bar. However, the values of the turbocharger lubrication oil inlet pressure remain between 1.8 and 2.6 bar (Fig. 6.7).

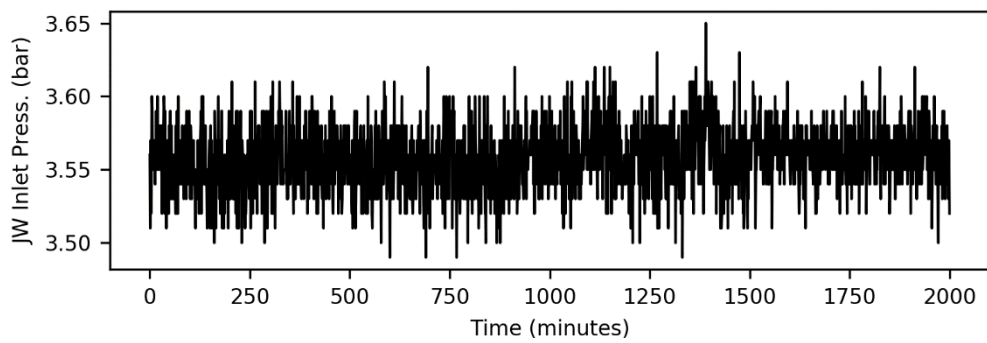


Fig. 6.6. Time series plot of the jacket water cooling system inlet pressure.

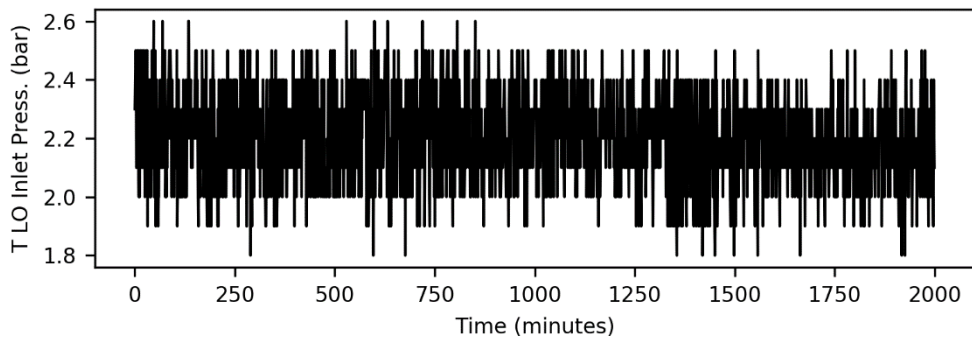


Fig. 6.7. Time series plot of the turbine lubrication oil inlet pressure of the turbocharger.

Table 6.2. Descriptive statistics of the monitored features.

	Main Engine			Lubrication Oil	Jacket Cooling Water	Turbocharger	Scav. Air Receiver.
	Speed (rev./min)	Power (kW)	Fuel Flow Rate (tn/hr)	Inlet Press. (bar)	Inlet Press. (bar)	T LO Inlet Press. (bar)	Scav. Air Press. (bar)
Mean	102.95	4421.09	0.88	2.38	3.56	2.20	1.04
Std.	2.98	438.71	0.08	0.02	0.02	0.16	0.18
Min.	98.32	3676.14	0.71	2.33	3.49	1.80	0.73
25%	98.94	3878.76	0.80	2.34	3.54	2.10	0.83
50%	104.37	4601.76	0.90	2.39	3.56	2.20	1.12
75%	104.47	4697.83	0.93	2.39	3.57	2.30	1.16
Max.	110.96	5528.74	1.11	2.41	3.65	2.60	1.48

The descriptive statistics are also presented in Table 6.2. As these statistics indicate, all the features excluding the inlet pressure of the jacket cooling water system and the turbine lubrication oil inlet pressure of the turbocharger, which present nearly a symmetric distribution, are slightly or highly skewed. Accordingly, the Box-Cox transformation needs to be applied to all the identified skewed features so that their skewness can be removed. Standardisation is also implemented to avoid unequal contribution of the features when considering the multivariate imputation techniques.

To analyse the relationship between features, the Pearson's and Spearman's rank correlation coefficients are estimated. The absolute values of the obtained results are contained in two different matrices (Pearson's correlation coefficient matrix and Spearman's rank correlation coefficient matrix) and represented by displaying heatmap plots (Fig. 6.8 and Fig. 6.9).

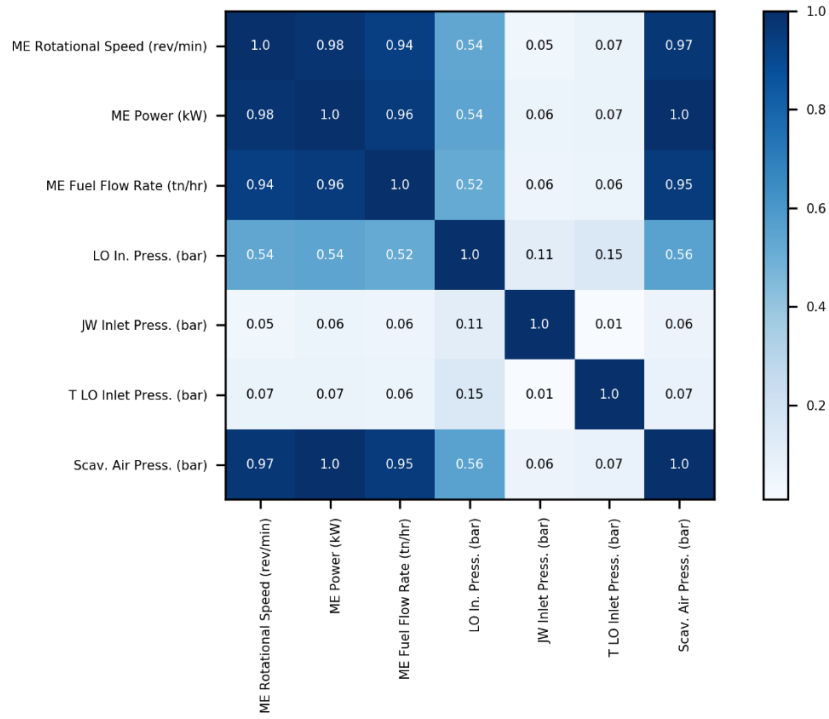


Fig. 6.8. Heatmap plot of Pearson's correlation coefficient matrix.



Fig. 6.9. Heatmap plot of Spearman's rank correlation coefficient matrix.

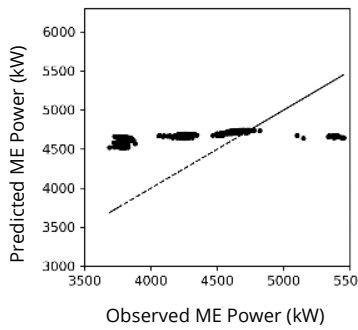
The rotational speed of the marine engine is highly correlated with the engine power. Additionally, it is also correlated with the fuel flow rate and the scavenge air pressure, as these parameters influence the engine combustion. Thus, the power of the marine engine presents a strong relationship with the engine rotational speed, the fuel flow rate, and the scavenge air pressure. The fuel flow rate and the scavenge air pressure are not only correlated with the rotational speed and the engine power but also between themselves, as they present a relationship derived from the conditions needed along the combustion process. The inlet pressure of the jacket cooling water system is not correlated with any presented feature, as it does not have any contact with any other analysed system. Similarly, the turbine lubrication oil inlet pressure of the turbocharger and the inlet pressure of the lubrication oil system neither influence nor are influenced by any analysed feature.

After performing the cross-reference between data-driven correlation analysis and engineering knowledge, the resulting correlation matrix is presented in Table 6.3. Only those features that have at least one relationship with another feature are presented in this table.

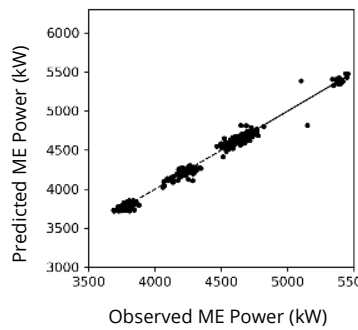
The main imputation results when considering the main engine power are graphically represented in Fig. 6.10. This graphical representation is obtained from presenting a scatterplot between the observed and the imputed instances for each analysed method.

Table 6.3. Correlation matrix of the monitored features.

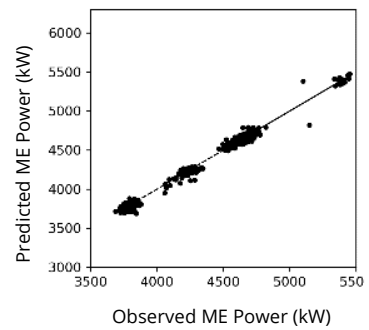
	Main Engine			Scav. Air Receiver
	Rotational Speed (rev/min)	Power (kW)	Fuel Flow Rate (tn/hr)	Scav. Air Press. (bar)
Rotational Speed (rev/min)		•	•	•
Power (kW)	•		•	•
Fuel Flow Rate (tn/hr)	•	•		•
Scav. Air Press. (bar)	•	•	•	



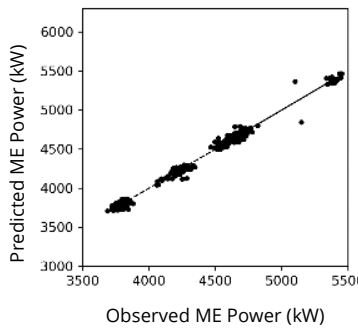
(a) Mean Imputation



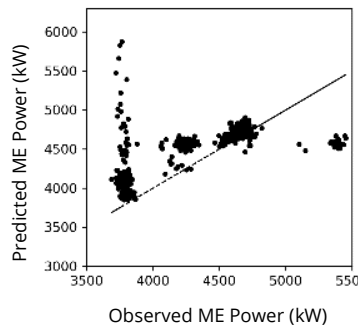
(b) STL Decomposition



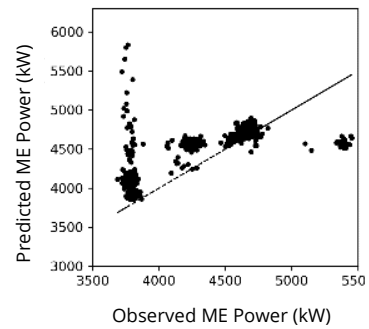
(c) Holt Winters' Seasonal



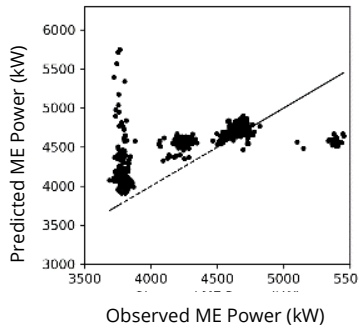
(d) ARIMA



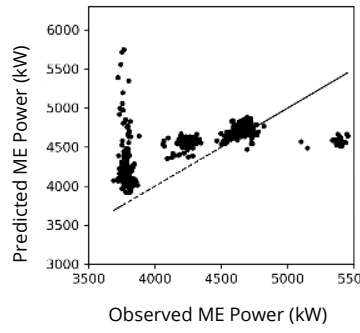
(e) PLS Regression



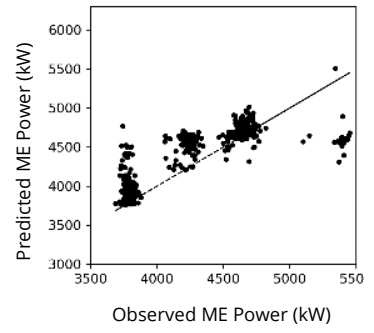
(f) Ridge Regression



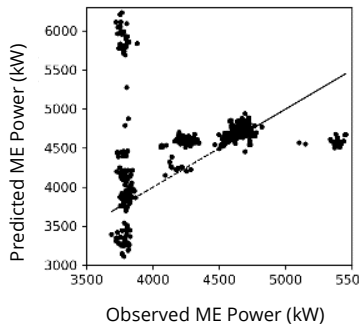
(g) LASSO Regression



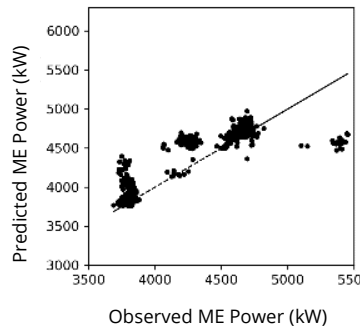
(h) ElasticNet Regression



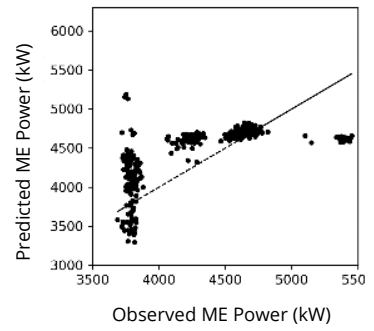
(i) k -NN



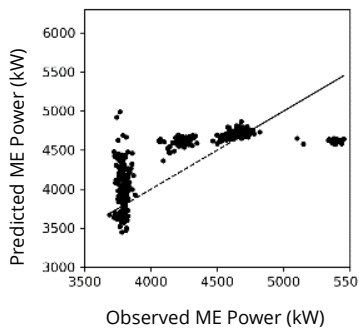
(j) SVR with Linear Kernel



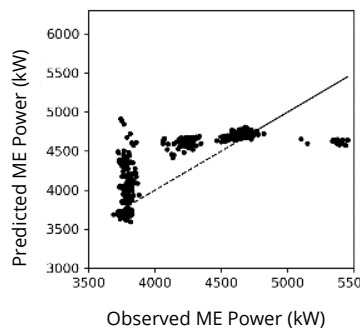
(k) SVR with RBF Kernel



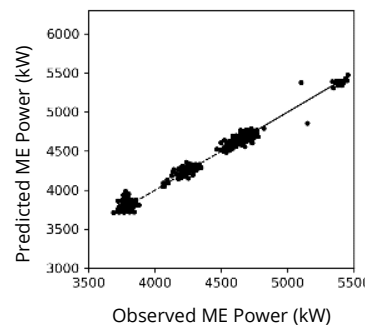
(l) NNs with 1 hidden layer



(m) NNs with 2 hidden layers



(n) NNs with 3 hidden layers



(o) VAR

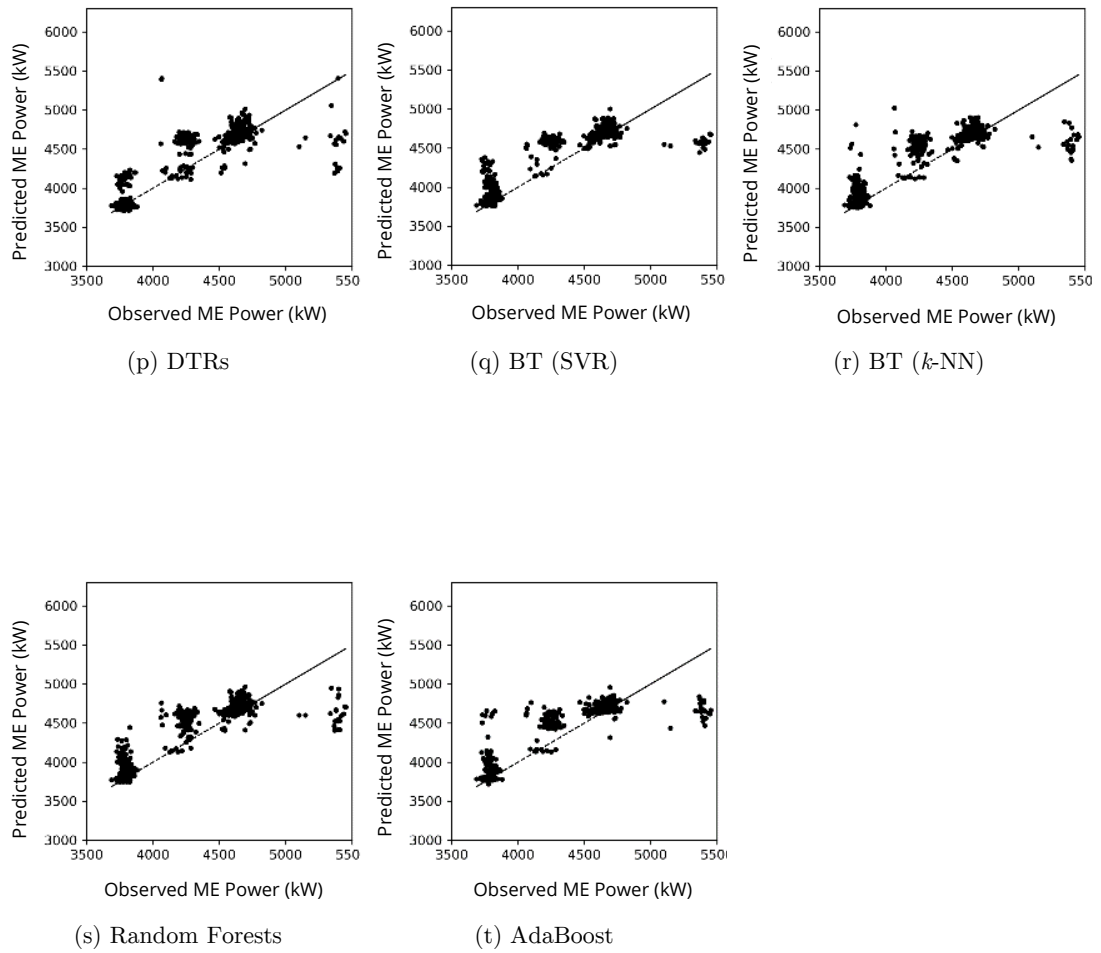


Fig. 6.10. Comparison between observed values of the main engine power parameter and imputed values through the implementation of the machine learning and time series forecasting models.

As perceived in Fig. 6.10 and Tables 6.4 - 6.10, VAR is the most accurate multivariate imputation technique and that ARIMA outperforms the other univariate imputation

methods. A comprehensive description of all analysed data imputation techniques is presented in the subsequent paragraphs.

The first univariate imputation method analysed is the mean imputation, considered a naïve method that imputes incomplete values with the mean of the current analysed sample. Mean imputation yielded one of the worst results of the entire modelled techniques, leading to the most biased estimates when imputing missing values identified in the main engine fuel flow rate (with a MAPE of 9.17% and MedAE of 0.066 tn/hr, see Table 6.6), and the scavenging air pressure of the scavenge air receiver (with a MAPE of 19.97% and MedAE of 0.18 bar, see Table 6.10). This lack of accuracy is due to the distortion of the parameter distribution, as expressed in Fig. 6.10 (a), where the incomplete values are imputed with nearly the same value, which corresponds with the mean of the current sample being analysed, and thus obtaining bias estimates. Therefore, by disrupting the distribution of the variable the relationship between variables is also affected, and thus reducing correlation estimates towards zero. Furthermore, if the nature of the incomplete values is determined by either missing at random (MAR) or not missing at random (NMAR) and the mean imputation is utilised, the mean estimate may be biased. Despite the disadvantages presented, this univariate imputation technique is usually applied due to its easiness in both interpretation and implementation.

The STL technique is the second univariate imputation technique analysed. This technique outperformed the remaining models when imputing the missing values for the inlet pressure parameter of the lubrication oil system. The technique achieved an execution time of 0.02 seconds and a RMSE of 0.002 bar (please see Table 6.7). STL is robust to outliers, the trend smoothness can be adjusted, and any type of seasonality can be considered and regulated over time. In addition, the execution time is low. However, certain disadvantages cannot be diminished. For instance, it can only be implemented if the time series presents trend and seasonality. In addition, the seasonal period needs to be estimated. This fact may lead to bias estimates if this period is not optimally selected.

The third univariate imputation technique refers to the Holt-Winters' seasonal method. This method leads to the one of the most accurate results in the main engine rotational speed parameter (with a MAPE of 0.13% and a MedAE of 0.08 r/min, see Table 6.4). Accordingly, as perceived in Fig. 6.10 (c), nearly all predicted estimates are similar to the observed occurrences. Exponential smoothing methods are easy to interpret and apply. Moreover, recent observations are considered more significant than earlier observations. Also, different exponential smoothing methods can be applied based on the characteristics of the data (e.g., simple exponential smoothing can be implemented when the time series does not present trend and seasonality, whereas Holt's linear trend method is more appropriate if a trend can be identified). Despite these advantages, these methods present two main drawbacks. The first one

refers to the adequate selection of the exponential smoothing method based on the characteristics of the data. The second one relates to the optimal selection of the smoothing parameter.

The fourth and last univariate imputation method refers to ARIMA models. ARIMA models outperformed the remaining techniques in all the imputation studies (see Table 6.4 – Table 6.10). For instance, a MedAE of 0.076 r/min and a Max Error of 2.4 r/min were obtained when imputing the missing values of the main engine rotational speed parameter. Accordingly, it can be stated that one of its main advantages is its high accuracy for imputing missing values in short-term time series data. Also, these models can be applied to any type of time series. As limitation, it should be highlighted that ARIMA models only capture linear relationships. Furthermore, they require a larger number of instances in comparison to the other univariate imputation techniques. Also, the orders of the differencing, the autoregression, and the moving average model need to be optimally selected. With regards to the execution time, ARIMA models present more computational cost than any other univariate imputation method analysed.

The next group of analysis refer to the multivariate imputation techniques. Accordingly, the inlet pressure parameter of the lubrication oil system, the inlet pressure parameter of the jacket cooling water system, and the turbine lubricating oil

inlet pressure parameter of the turbocharger are excluded from the analysis, as such parameters do not present any predictor. With regards to the analysed imputation techniques, the first group considered relates to the linear regression methods, which consist of PLS and penalised methods (Ridge regression, LASSO regression, and ElasticNet regression), are considered. In all cases, the results are analogous, being nearly equal in the case of the main engine fuel flow rate (with a MSE of 0.005 (tn/hr)², see Table 6.6). In the case of the main engine rotational speed, the Ridge regression is the linear regression technique that yields better results (with a RMSE of 1.92 r/min and a MedAE of 1.72 r/min, see Table 6.4), whereas in the case of the main engine power the PLS is the linear regression method with the best performance (with a RMSE of 272.63 kW and a MedAE of 182.38 r/min, see Table 6.5). As presented in Fig. 6.10 (e) – Fig. 6.10 (h), the predicted values are more dispersed in comparison to the ones estimated by the univariate imputation techniques. This indicates that the univariate imputation techniques lead to the best results. The lack of accuracy presented by these linear methods can be perceived more significantly in the outermost values, those farthest from the line, whereby the lesser values to be predicted in the fourth steady state (see Fig. 6.1) present the most extreme values; greater than 6,000 kW. Whereas the observed values present a value lower than 4,000 kW. Such an accuracy can be presented due to either the presence or distinct states or the lack of data for a specific operational state. Furthermore, linear regression methods only capture linear relationships between the response and the predictors, and they are sensitive to outliers. Nevertheless, these methods are easy to interpret,

and both their complexity and execution time are low with regards to other multivariate imputation methods.

Even though all the analysed relationships between the responses and the predictors are linear, non-linear regression methods are also analysed. The first refers to the k -NN. The main challenge perceived when analysing this method was the optimal selection of the number of neighbours. Furthermore, its performance may be degraded when considering large samples, the dataset present high dimensionality, and/or when either noisy instances or outliers are presented. However, analogous to the linear regression methods, its execution time is low. As presented in Tables 6.4 – 6.10 and Fig. 6.10 (i), the imputation of this method yields better results than linear regression models. Although the Euclidean distance is considered in this study, various distance criteria can be utilised when implementing k -NN.

SVR is another of the multivariate imputation methods analysed. Specifically, two distinct kernel functions are employed: 1) linear and 2) RBF. The SVR with linear kernel function model results in tandem with the mean imputation results lead to the worst imputation results. Nevertheless, the predicted values present less dispersion and are more analogous to the observed values when the SVR with RBF kernel is considered (see Fig 6.10 (k)). Thus, the possibility to employ distinct kernel functions makes SVR adaptable. Furthermore, both linear and non-linear relationships can be

considered. SVR is also robust to outliers, although a lack of accuracy may be presented if the dataset presents noisy data. Also, both the kernel function and the tuning parameters need to be optimally selected. Additionally, SVR is not convenient when the analysed sample is large, it is difficult to interpret, and its computational cost is high.

NNs with one, two, and three hidden layers are also applied in this study. The three models yield similar results. However, the execution time increases exponentially when additional hidden layers are introduced. Also, from observations of Figs. 6.10 (l) – 6.10 (n), the predicted values do not correspond with the observed instances. The main reason of such an observation may be the amount of data required to train DL models. In this study short-term data is considered, whereas DL models required large amounts of data to achieve accurate imputations. Also, the architectures of the NNs are complex to define, they are susceptible to over-fitting, and their performances are unexplained. However, they capture both linear and non-linear relationships.

Analogous to the results obtained from the analysis of univariate imputation techniques, the most accurate results of the multivariate imputation group are obtained when the VAR models are considered. Their results are analogous to ARIMA models (see Tables 6.4 – 6.10), as VAR models are a generalisation of the autoregressive models. Thus, as presented in Fig. 6.10 (o), the predicted values are

remarkably analogous to the observed instances. VAR models are easy to implement and presents high accuracy when short-term data is considered. However, stationarity is required. Also, the computational cost of VAR models is higher than other analysed models.

The last group of multivariate imputation methods analysed relates to decision tree regressors and ensemble methods. All these methods present analogous results (please see in Fig. 6.10 (p) – Fig. 6.10 (t)). Decision tree regressors present better imputation results than the ensemble methods, although this difference is not significant. Additionally, when Tables 6.4 – 6.10 are considered, it can be observed that both decision tree regressors and ensemble methods present results similar to other non-linear regression methods analysed (SVR with RBF kernel and k -NN). Decision tree regressors are easy to interpret, present low execution time, and are unlikely to overfit. However, they are likely to present sub-optimal imputation performance when dealing with continuous data. Also, they are also likely to be unstable, as only one tree is modelled and. Thus, small adjustments in the training data may alter the partitions completely. To solve such a drawback, ensemble methods are considered.

A summary of both advantages and disadvantages identified for each imputation method are summarised in Table 6.11. As stated, precedingly, univariate imputation methods yield better results than multivariate imputation methods. A major reason

for such a fact may be that large amounts of data are required to train the model every time a missing value is imputed. Moreover, multivariate imputation methods cannot be performed if the analysed parameter does not present predictors or the instances to be utilised include predictors with missing values. Also, their computational cost is high, and thus they are not appropriate when the imputation is implemented in real-time. Also, multicollinearity among independent variables has not been analysed and treated accordingly, which may have led to a decrease in the imputation accuracy.

Table 6.4. Imputation results of the main engine rotational speed parameter.

	Execution Time (s)	MSE $((\text{r/min})^2)$	MSLE $(\log((\text{r/min})^2))$	RMSE (r/min)	MAPE (%)	MedAE (r/min)	Max. Error (r/min)
Mean imputation	0.0002	11.49	0.00011	2.39	2.355	1.920	6.652
STL decomposition	0.0234	0.066	0.00001	0.118	0.115	0.077	2.643
Holt Winters	0.1384	0.071	0.00001	0.129	0.126	0.081	2.581
ARIMA	2.8614	0.063	0.00001	0.118	0.116	0.076	2.404
PLS regression	0.0059	9.162	0.00084	1.989	1.952	1.731	9.746
Ridge regression	0.0037	8.342	0.00076	1.920	1.882	1.718	9.455
LASSO regression	0.0964	8.875	0.00081	1.976	1.939	1.854	9.604
ElasticNet regression	0.0879	8.935	0.00082	1.981	1.944	1.855	9.570
k-Nearest Neighbors	0.0025	4.666	0.00041	1.198	1.151	0.236	8.520
SVR (linear kernel)	0.0401	10.215	0.00094	2.164	2.129	1.852	8.515
SVR (RBF Kernel)	0.0268	4.799	0.00043	1.226	1.178	0.185	6.960
NN (1 hidden layer)	53.3214	6.274	0.00057	1.636	1.593	1.193	7.107
NN (2 hidden layers)	55.5707	5.913	0.00054	1.553	1.509	0.934	6.887
NN (3 hidden layers)	57.6319	6.004	0.00054	1.562	1.518	0.737	6.826
Vector autoregression	2.9989	0.085	0.00001	0.144	0.140	0.084	2.557
Decision tree regressor	0.0040	5.292	0.00047	1.143	1.095	0.140	9.170
Bagged tree (SVR)	0.1115	4.770	0.00042	1.221	1.173	0.201	7.033
Bagged tree (k-NN)	0.0208	4.494	0.00040	1.147	1.099	0.162	8.367
Random forest	0.2710	4.621	0.00041	1.129	1.081	0.179	8.425
AdaBoost	0.0259	4.020	0.00036	1.125	1.084	0.330	9.203

Table 6.5. Imputation results of the main engine power parameter.

	Execution Time (s)	MSE (kW ²)	MSLE (log(kW ²))	RMSE (kW)	MAPE (%)	MedAE (kW)	Max. Error (kW)
Mean imputation	0.0002	282912.4	0.0155	423.433	10.541	419.265	938.840
STL decomposition	0.0237	1808.440	0.0001	29.691	0.688	23.877	330.164
Holt Winters	0.1862	1867.622	0.0001	30.660	0.716	24.158	329.228
ARIMA	1.5618	1520.140	0.0001	27.123	0.631	21.868	303.016
PLS regression	0.0066	165252.430	0.0084	272.632	6.609	182.381	2109.975
Ridge regression	0.0040	165309.421	0.0085	273.729	6.637	183.178	2069.081
LASSO regression	0.0849	176189.410	0.0092	297.426	7.256	233.327	1986.912
ElasticNet regression	0.0693	178432.169	0.0093	299.607	7.311	242.007	1986.023
k-Nearest Neighbors	0.0029	78476.090	0.0039	189.779	4.394	111.675	1061.520
SVR (linear kernel)	0.0439	418935.348	0.0195	370.683	9.193	183.248	2469.608
SVR (RBF Kernel)	0.0446	70917.784	0.0035	176.527	4.052	89.021	919.772
NN (1 hidden layer)	63.6674	128537.8	0.0070	271.869	6.556	226.968	1436.062
NN (2 hidden layers)	63.8124	112474.7	0.0059	250.320	5.988	161.291	1224.988
NN (3 hidden layers)	64.0917	115563.2	0.0061	250.641	5.993	141.166	1172.232
Vector autoregression	3.0802	2562.996	0.0001	36.648	0.858	27.429	290.651
Decision tree regressor	0.0033	77802.296	0.0037	171.422	3.885	72.340	1339.880
Bagged tree (SVR)	0.1985	71858.359	0.0035	178.552	4.107	89.405	925.058
Bagged tree (k-NN)	0.0210	68086.261	0.0033	167.098	3.819	96.821	1041.015
Random forest	0.2512	61853.362	0.0030	161.962	3.688	88.909	994.745
AdaBoost	0.0217	60843.155	0.0030	160.461	3.692	97.181	940.996

Table 6.6. Imputation results of the main engine fuel flow rate parameter.

	Execution Time (s)	MSE ((tn/hr) ²)	MSLE (log((tn/hr) ²))	RMSE (tn/hr)	MAPE (%)	MedAE (tn/hr)	Max. Error (tn/hr)
Mean imputation	0.0002	0.009	0.0025	0.075	9.169	0.066	0.191
STL decomposition	0.0234	0.001	0.0002	0.021	2.487	0.017	0.118
Holt Winters	0.1988	0.001	0.0002	0.020	2.358	0.015	0.132
ARIMA	3.8901	0.001	0.0002	0.020	2.355	0.015	0.132
PLS regression	0.0067	0.005	0.0014	0.052	6.239	0.037	0.330
Ridge regression	0.0040	0.005	0.0014	0.052	6.188	0.037	0.330
LASSO regression	0.0877	0.005	0.0015	0.055	6.561	0.042	0.329
ElasticNet regression	0.0753	0.005	0.0015	0.055	6.589	0.042	0.328
k-Nearest Neighbors	0.0028	0.003	0.0009	0.041	4.690	0.030	0.208
SVR (linear kernel)	0.0432	0.010	0.0026	0.060	7.175	0.034	0.463
SVR (RBF Kernel)	0.0501	0.003	0.0008	0.038	4.379	0.027	0.194
NN (1 hidden layer)	41.4862	0.004	0.0011	0.047	5.610	0.036	0.213
NN (2 hidden layers)	42.4137	0.004	0.0011	0.047	5.534	0.034	0.197
NN (3 hidden layers)	43.2885	0.004	0.0011	0.047	5.561	0.034	0.191
Vector autoregression	3.4904	0.001	0.0004	0.028	3.300	0.022	0.142
Decision tree regressor	0.0035	0.004	0.0011	0.043	4.971	0.030	0.280
Bagged tree (SVR)	0.2094	0.003	0.0008	0.038	4.458	0.028	0.194
Bagged tree (k-NN)	0.0208	0.003	0.0008	0.037	4.266	0.027	0.219
Random forest	0.2638	0.003	0.0008	0.038	4.401	0.028	0.227
AdaBoost	0.0299	0.003	0.0008	0.039	4.579	0.028	0.192

Table 6.7. Imputation results of the inlet pressure parameter of the lubrication oil system.

	Execution Time (s)	MSE (bar ²)	MSLE (log(bar ²))	RMSE (bar)	MAPE (%)	MedAE (bar)	Max. Error (bar)
Mean imputation	0.0002	0.00095	0.000084	0.021	0.911	0.008	0.061
STL decomposition	0.0236	0.00002	0.000001	0.002	0.103	0.001	0.013
Holt Winters	0.2137	0.00001	0.000001	0.003	0.110	0.002	0.014
ARIMA	5.3972	0.00002	0.000001	0.002	0.104	0.001	0.013

Table 6.8. Imputation results of the inlet pressure parameter of the jacket cooling water system.

	Execution Time (s)	MSE (bar ²)	MSLE (log(bar ²))	RMSE (bar)	MAPE (%)	MedAE (bar)	Max. Error (bar)
Mean imputation	0.0002	0.0005	0.00002	0.017	0.481	0.017	0.066
STL decomposition	0.0239	0.0006	0.00003	0.019	0.537	0.016	0.080
Holt Winters	0.1656	0.0005	0.00002	0.017	0.491	0.014	0.070
ARIMA	4.3112	0.0004	0.00002	0.017	0.468	0.014	0.066

Table 6.9. Imputation results of the turbine lubricating oil inlet pressure parameter of the turbocharger.

	Execution Time (s)	MSE (bar ²)	MSLE (log(bar ²))	RMSE (bar)	MAPE (%)	MedAE (bar)	Max. Error (bar)
Mean imputation	0.0002	0.026	0.0026	0.135	6.231	0.111	0.412
STL decomposition	0.0231	0.030	0.0029	0.142	6.506	0.126	0.498
Holt Winters	0.1847	0.028	0.0027	0.139	6.360	0.131	0.438
ARIMA	3.5027	0.026	0.0025	0.134	6.168	0.114	0.412

Table 6.10. Imputation results of the scavenging air pressure of the scavenge air receiver.

	Execution Time (s)	MSE (bar ²)	MSLE (log(bar ²))	RMSE (bar)	MAPE (%)	MedAE (bar)	Max. Error (bar)
Mean imputation	0.0002	0.047	0.0121	0.17	19.97	0.18	0.41
STL decomposition	0.0232	0.001	0.0001	0.01	1.53	0.01	0.18
Holt Winters	0.2338	0.001	0.0002	0.02	1.55	0.01	0.19
ARIMA	1.6972	0.001	0.0002	0.01	1.50	0.01	0.18
PLS regression	0.0068	0.037	0.0085	0.13	13.88	0.08	0.88
Ridge regression	0.0047	0.036	0.0085	0.13	13.88	0.08	0.86
LASSO regression	0.0775	0.037	0.0086	0.13	14.19	0.08	0.87
ElasticNet regression	0.0803	0.037	0.0088	0.13	14.44	0.09	0.86
k-Nearest Neighbors	0.0028	0.018	0.0042	0.09	9.62	0.06	0.47
SVR (linear kernel)	0.0418	0.068	0.0149	0.15	17.10	0.08	1.02
SVR (RBF Kernel)	0.0501	0.015	0.0035	0.09	8.66	0.05	0.35
NN (1 hidden layer)	72.371	0.027	0.0067	0.12	13.47	0.09	0.65
NN (2 hidden layers)	74.0086	0.023	0.0057	0.11	12.29	0.08	0.54
NN (3 hidden layers)	75.7735	0.022	0.0054	0.11	11.91	0.07	0.50
Vector autoregression	3.3160	0.001	0.0002	0.02	1.91	0.01	0.19
Decision tree regressor	0.0035	0.016	0.0036	0.08	7.92	0.04	0.54
Bagged tree (SVR)	0.2176	0.015	0.0036	0.09	8.78	0.05	0.35
Bagged tree (k-NN)	0.0205	0.015	0.0035	0.08	8.36	0.05	0.43
Random forest	0.2711	0.014	0.0031	0.08	7.88	0.05	0.42
AdaBoost	0.0217	0.014	0.0034	0.08	8.64	0.06	0.41

Table 6.11. Advantages and disadvantages of the implemented imputation techniques.

Technique	Advantages	Disadvantages
Mean imputation	<ul style="list-style-type: none"> • Easy to interpret and implement. • The execution time is low. 	<ul style="list-style-type: none"> • Distortion of the parameter distribution. • Disruption of the relationship between features. • Bias of the mean estimates when the nature of the incomplete values are either MAR or NMAR.
STL decomposition	<ul style="list-style-type: none"> • Robust to outliers. • Trend smoothness can be regulated. • Execution time is low. • Easy to interpret. • Any seasonality type can be considered, and the seasonal component can be adjusted over time. 	<ul style="list-style-type: none"> • Possibility to be applied only when the time series presents trend and seasonality. • The definition of the seasonal period is required.
Exponential smoothing methods	<ul style="list-style-type: none"> • Easy to interpret and implement. • Recent observations are considered more significant than earlier observations. • Various exponential smoothing methods can be applied based on the characteristics of the time series. 	<ul style="list-style-type: none"> • The type of exponential smoothing method to be utilised needs to be identified. • Different parameters need to be optimally selected.
Autoregressive integrated moving average (ARIMA) models	<ul style="list-style-type: none"> • Present higher accuracy when imputing incomplete values in short-term data. • Applicable to nearly all types of time series. 	<ul style="list-style-type: none"> • Only captures linear relationships. • Need more data than other univariate imputation methods analysed. • Differencing is required if the data is not stationary. Then, the order of differencing needs to be specified. • The orders of the autoregression and the moving average model need to be optimally selected. • The computational cost is high in comparison with the other univariate imputation methods analysed, and thus its execution time is also large.

Linear regression methods (PLS regression and penalised models (LASSO, Ridge, and ElasticNet regression))	<ul style="list-style-type: none"> • Easy to interpret. • Low execution time. • Regularisation models avoid over-fitting. • Great performance when dealing with a linear relationship. 	<ul style="list-style-type: none"> • Sensitive to outliers. • Only captures linear relationships between the response and the predictors. • Risk when extrapolating. • Large amount of data required every time an incomplete value needs to be imputed. • Penalised parameters (penalised models) and the number of components to be utilised (PLS) need to be optimally selected.
k -Nearest Neighbors	<ul style="list-style-type: none"> • Different distance criteria can be implemented. • Easy to interpret. • Weights can be added to the estimated distances hinging on the closeness of the records. 	<ul style="list-style-type: none"> • Performance degradation when the sample considered is large or dimensions are high. • Feature scaling is needed. • Sensitive to outliers and noisy data. • The number of neighbours required to impute the incomplete value needs to be optimally selected.
SVR (with linear and RBF kernels)	<ul style="list-style-type: none"> • Captures both linear and non-linear relationships. • Easily adaptable. • Robust to outliers. • Various kernel functions can be utilised. 	<ul style="list-style-type: none"> • Poor performance when the sample contains noise. • Inconvenient when the sample is large. • Both the kernel function and the tuning parameters need to be optimally selected. • Feature scaling is required. • Difficult to interpret. • High computational cost.
Neural Networks (NN) (with 1, 2, and 3 hidden layers)	<ul style="list-style-type: none"> • Capture both linear and non-linear relationships. 	<ul style="list-style-type: none"> • Requires large amounts of data to train the model. • Complexity in the network structure definition. • Susceptible to over-fitting. • High computational cost. • Unexplained performance.
Vector autoregressive (VAR) models	<ul style="list-style-type: none"> • Easy to interpret. • High accuracy when dealing with short-term data. • All parameters included in the model are considered endogenous. • All incomplete values of an instance can be imputed by implementing the model once. 	<ul style="list-style-type: none"> • Can only be utilised if data is stationary. • High computational cost. • The order of the model needs to be optimally selected.

Decision tree regressors	<ul style="list-style-type: none"> • Apply feature selection intrinsically. • Require less data pre-processing. • Easy to interpret. • Handle incomplete values. • Low execution time. • Unlikely to over-fit. 	<ul style="list-style-type: none"> • Likely to present sub-optimal imputation performance when continuous data is considered. • Imputations are not accurate if instances with incomplete values are not similar to instances utilised for training the model. • Instability.
Ensemble methods (Bagged trees (with k -NN and SVR regressors), Random forests, and AdaBoost)	<ul style="list-style-type: none"> • Apply feature selection intrinsically. • Require less data pre-processing. • Handle incomplete values. • Unlikely to over-fit. 	<ul style="list-style-type: none"> • Complex to interpret. • Higher computational cost than decision tree regressors. • Likely to present sub-optimal imputation performance when continuous data is considered. • Imputations are not accurate if instances with incomplete values are not similar to instances utilised for training the model.

6.3. Case Study 2. Hybrid Imputation Framework⁹

Case study 2 aims to validate the methodology introduced in section 4.3. *Hybrid Data Imputation Framework*. Accordingly, a DMD-MAN B&W 6S50MC-C main propulsion engine of a cargo vessel is analysed. A total of four parameters are considered for this case study: 1) main engine rotational speed, 2) main engine power, 3) main engine fuel flow rate, and 4) scavenging air pressure of the scavenge air receiver.

Prior to the performance of data imputation, non-operational states are identified and discarded. Standardisation is also applied. After data pre-processing, a total of roughly

⁹ The data imputation framework presented in this section has been already converted to a journal paper, and has been published in the *Ship and Offshore Structures* journal (Velasco-Gallego and Lazakis, 2021a).

2,160 instances are considered for analysis. Such instances have been collected in a one-minute frequency basis.

The evolution of the main engine rotational speed time series is visualised in Fig. 6.11. In total, four steady states are identified. The first one initiates at the first instant and persist around 103 r/min over 1,250 minutes. Suddenly, an abrupt change occurs and the rotational speed increases to over 112 r/min. Such a state remains approximately 130 minutes. Then, a slight adjustment occurs where the rotational speed decreases to roughly 108 r/min. This state lasts for around 215 minutes. Finally, a last change is observed. The rotational speed decreases until it accomplishes the values perceived in the first state. These adjustments refer to arrangements applied due to the contractual agreements between the charterer and the shipowner that define the fuel oil consumption and the vessel speed per day.

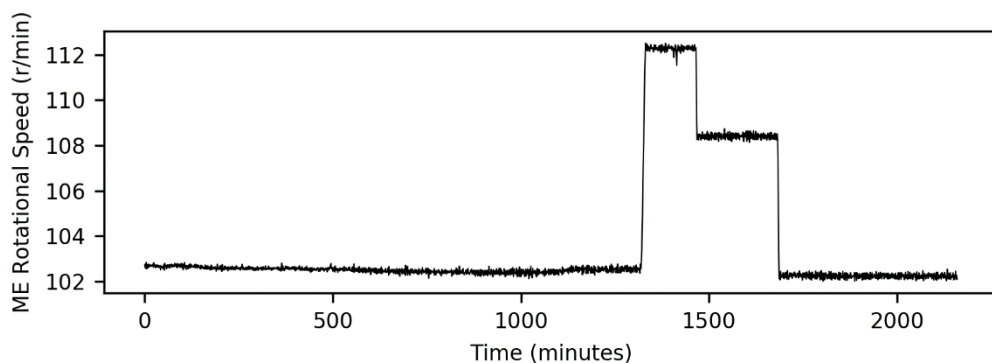


Fig. 6.11. Time series plot of the main engine rotational speed.

The remaining parameters (main engine power, the main engine fuel flow rate and the scavenging air pressure of the scavenge air receiver system) present an analogous evolution (please see Figs. 6.12-6.14).

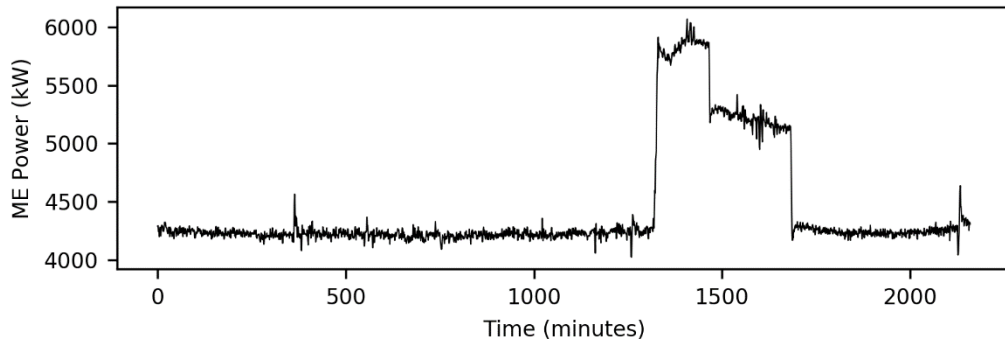


Fig. 6.12. Time series plot of the main engine power.

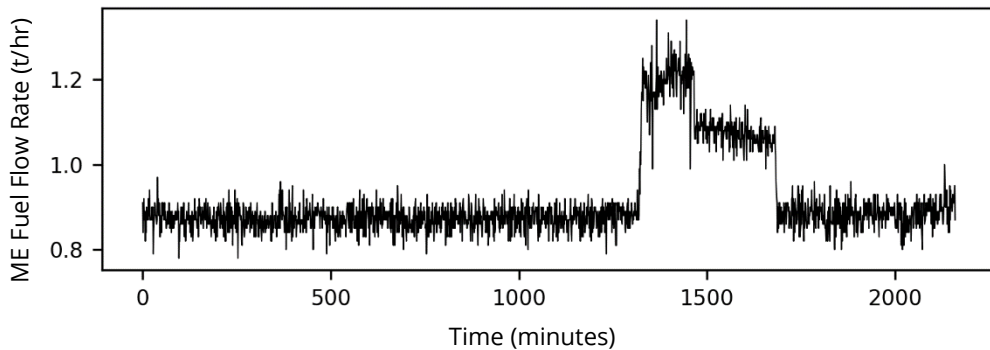


Fig. 6.13. Time series plot of the main engine fuel flow rate.

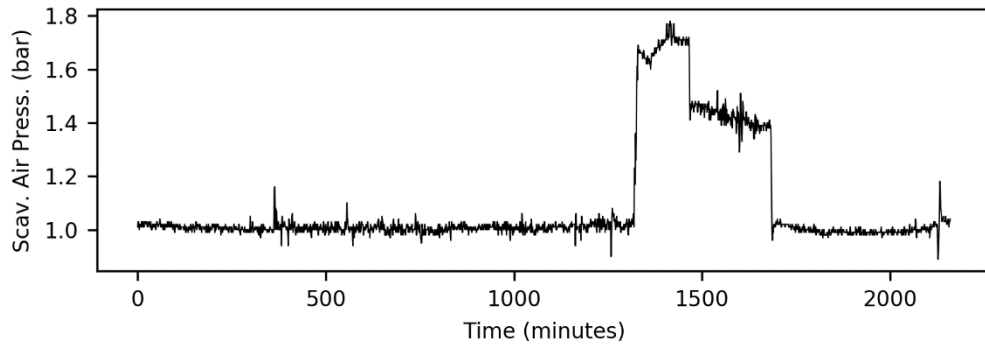


Fig. 6.14. Time series plot of the scavenging air pressure.

As observed in Table 6.12, all four parameters present a strong linear relationship. Accordingly, the Pearson's correlation coefficient matrix values lie between 0.95 and 1.00.

Table 6.12. Pearson's correlation coefficient matrix.

	ME rotational speed (rev/min)	ME power (kW)	ME fuel flow rate (tn/hr)	Scav. air pressure (bar)
ME rotational speed (rev/min)		0.99	0.95	0.99
ME power (kW)	0.99		0.96	1.00
ME fuel flow rate (tn/hr)	0.95	0.96		0.96
Scav. air pressure (bar)	0.99	1.00	0.96	

As the selected time series do not present missing values, these are generated to evaluate the imputation performance of the introduced framework. Specifically, as large gaps of missing values are analysed, a total of 5 large gaps of missing values are

generated at random in each time series. The dimensions of such gaps are selected at random by selecting these dimensions between two thresholds. Based on the number of instances perceived in the time series, the minimum and maximum number of missing values possible in each gap are set to 50 and 100.

Additionally, a comparative study is performed for validation purposes. Accordingly, the Multivariate Imputation by Chained Equations (MICE), which is one of the most widely implemented imputation techniques, is also implemented. Such an imputation technique has been precedingly implemented in the shipping domain (Cheliotis et al., 2019).

The first imputations are performed in the main engine rotational speed parameter (please see Fig. 6.15). All the large gaps are in the first state, except for last large gap, which is situated in the last state. As observed, the introduced methodology outperforms the MICE approach in terms of imputation performance. In this case, the MLP regressor is identified as the most adequate imputation model. It can be observed that the imputation performance decreases when the predictors present missing values in the same instances that need to be imputed. Such an aspect expresses the importance of preventing possible sensor failures to guarantee the quality of the data to avoid biased estimates.

For this specific example, a percentage of improvement of 77% is obtained. Such a fact demonstrates the efficient imputation performance of the first-order Markov chain model as a univariate imputation method against the mean imputation, which is employed in the MICE method. It is also observed that mean imputation disrupts the relationships between variables.

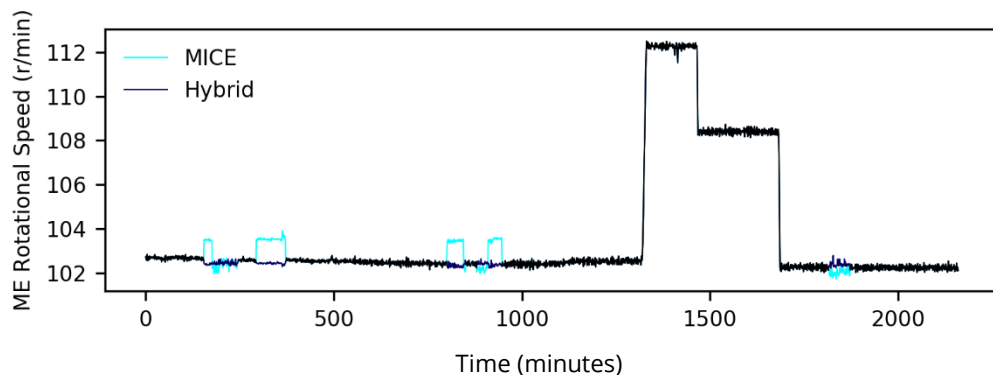


Fig. 6.15. Large gaps imputation of the main engine rotational speed.

Likewise, analogous patterns are observed in the remaining parameters. In the case of the main engine power parameter, it can be observed that two consecutive gaps are located between the end of the second and the start of the third sate (please see Fig. 6.16). No significant differences between the performance of the approaches are perceived in these instances. Conversely, their performance differs when the first and third gaps are considered. A major reason for such a fact is the unavailability of some predictors. Consequently, the introduced framework leads to an improvement percentage of 56%.

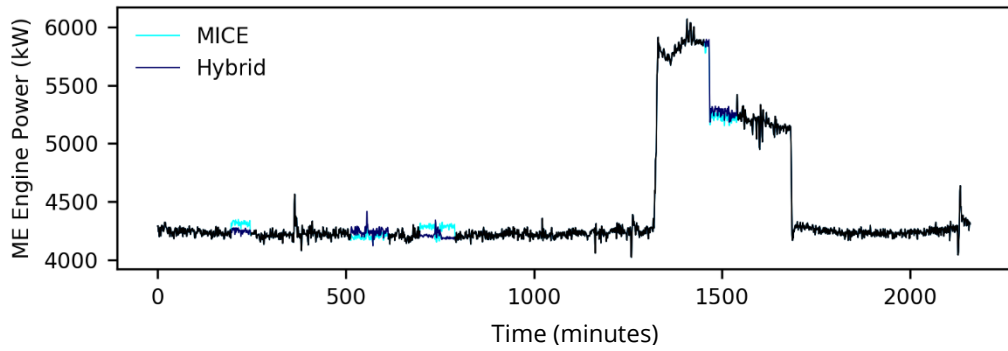


Fig. 6.16. Large gaps imputation of the main engine power.

An analogous pattern is again observed when the main engine fuel flow rate parameter is considered (please see Fig. 6.17 and Fig. 6.18). As presented in Table 4.13, the percentage of improvement for these two cases are 21% and 73% respectively.

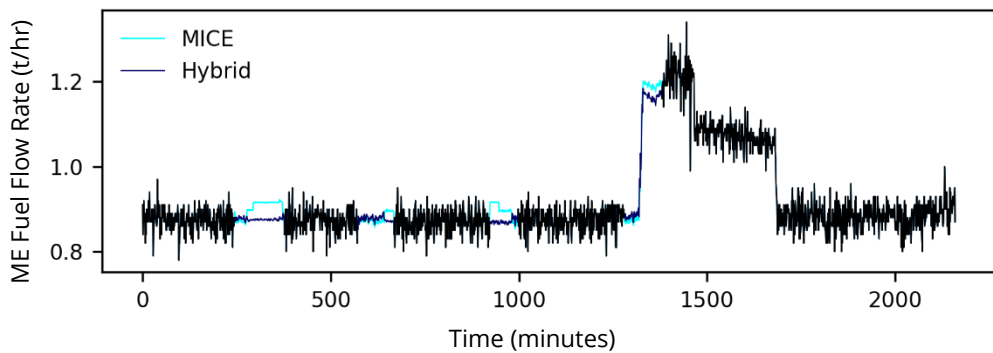


Fig. 6.17. Large gaps imputation of the main engine fuel flow rate.

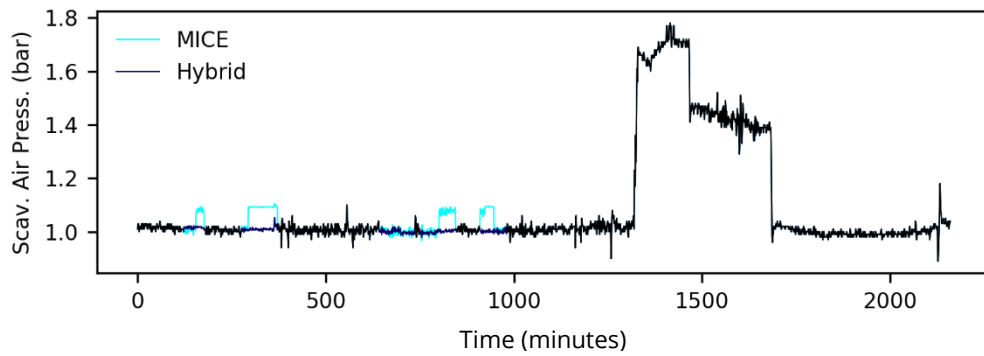


Fig. 6.18. Large gaps imputation of the scavenging air pressure of the scavenge air receiver.

Table 6.13. Large gaps of missing values imputation results.

	RMSE		
	Proposed framework	MICE	Percentage of improvement
ME rotational speed (rev/min)	0.17	0.75	77%
ME power (kW)	29.48	67.41	56%
ME fuel flow rate (tn/hr)	0.03	0.04	21%
Scav. air pressure (bar)	0.02	0.06	73%

6.4. Case Study 3. Analysis of LSTM-based VAE Regressor

for Data Imputation¹⁰

Having explored the methodology being analysed as a data imputation technique in section 4.4. *Analysis of LSTM-based Variational Autoencoders for Regression for Data Imputation*, a case study is introduced to assess its imputation performance.

Specifically, a total of 14 parameters (see Table 6.14) collected from a diesel generator

¹⁰ The data imputation framework presented in this section has been already converted to a journal paper, and has been published in the Journal of Ship Research (Velasco-Gallego and Lazakis, 2022c).

of a tanker ship are considered. P1 is considered the target variable, whereas the remaining parameters are considered predictors.

Table 6.14. Parameters of the diesel generator considered for the case study.

Id	Parameter
P1	Power
P2	Exhaust gas outlet temperature of cylinder 6
P3	Exhaust gas outlet temperature of cylinder 5
P4	Exhaust gas outlet temperature of cylinder 4
P5	Exhaust gas outlet temperature of cylinder 3
P6	Exhaust gas outlet temperature of cylinder 2
P7	Exhaust gas outlet temperature of cylinder 1
P8	Winding temperature T phase
P9	Winding temperature S phase
P10	Winding temperature R phase
P11	Turbocharger exhaust gas outlet temperature
P12	Cooling air temperature
P13	Lube oil inlet temperature
P14	Cylinder exhaust gas outlet temperature (average)

A total of 66,207 instances are considered for analysis. Such instances have been collected in a 1-minute frequency (Fig. 6.19). The introduced time series presents several non-operational states that need to be excluded from the analysis. Moreover, the adjustments introduced due to either contractual agreements between the charterer and the shipowner or weather conditions need to be also identified and treated accordingly. To perform this identification process, GMMs with EM is applied. The minimum and maximum of mixture components being analysed are 1 and 10,

respectively. Also, four distinct types of covariance are also evaluated (full, tied, diagonal, and spherical). As presented in Fig. 6.20, the spherical covariance type, and a total of 2 components have been selected as the best hyperparameters to train the model. In total, more than 49% of the dataset has been identified as non-operational instances, and thus these are discarded. Accordingly, only 33,745 are analysed in the subsequent steps.

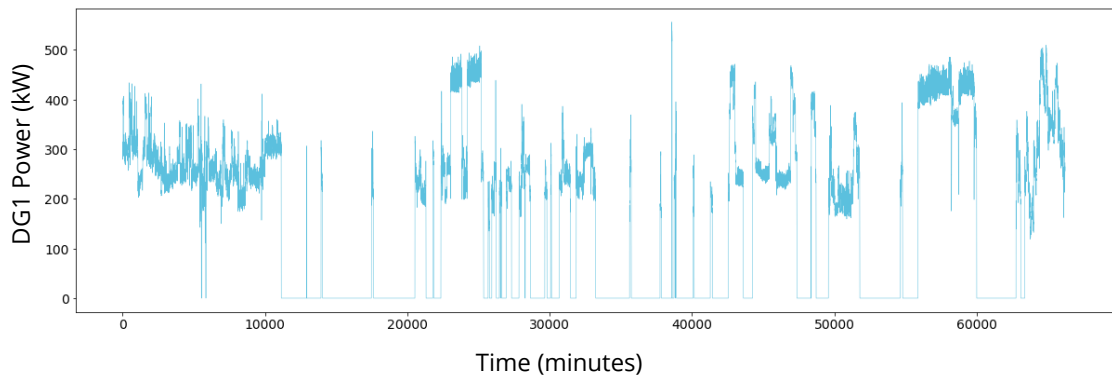


Fig. 6.19. Time series plot of the diesel generator power.

The descriptive statistics of the analysed instances are presented in Table 6.15. Additionally, the histograms and the Pearson's correlation coefficients are introduced in Fig. 6.21 and Table 6.16, respectively. All parameters present a strong correlation with the P1, except for the parameter P13.

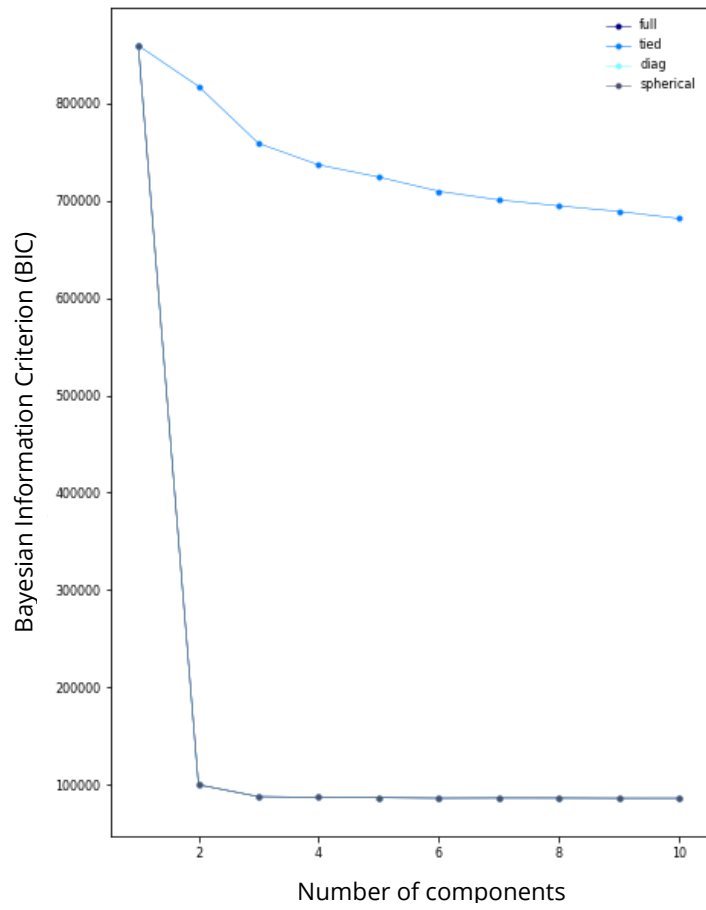
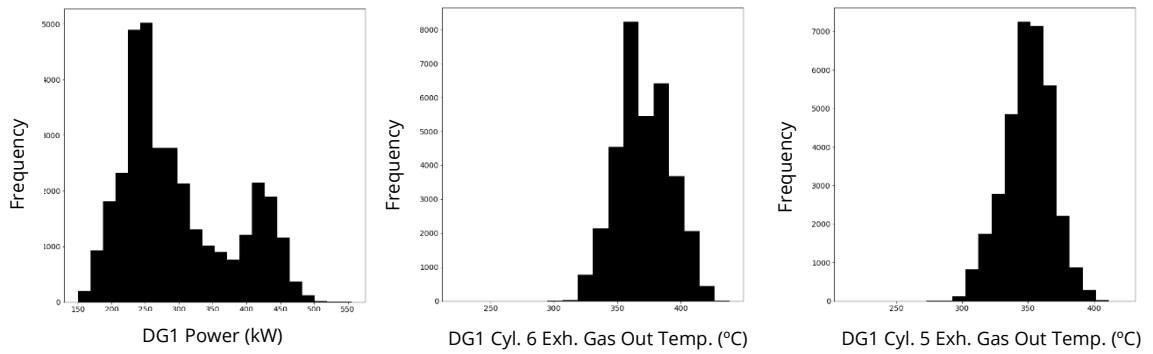


Fig. 6.20. Parameters' selection of the GMMs.



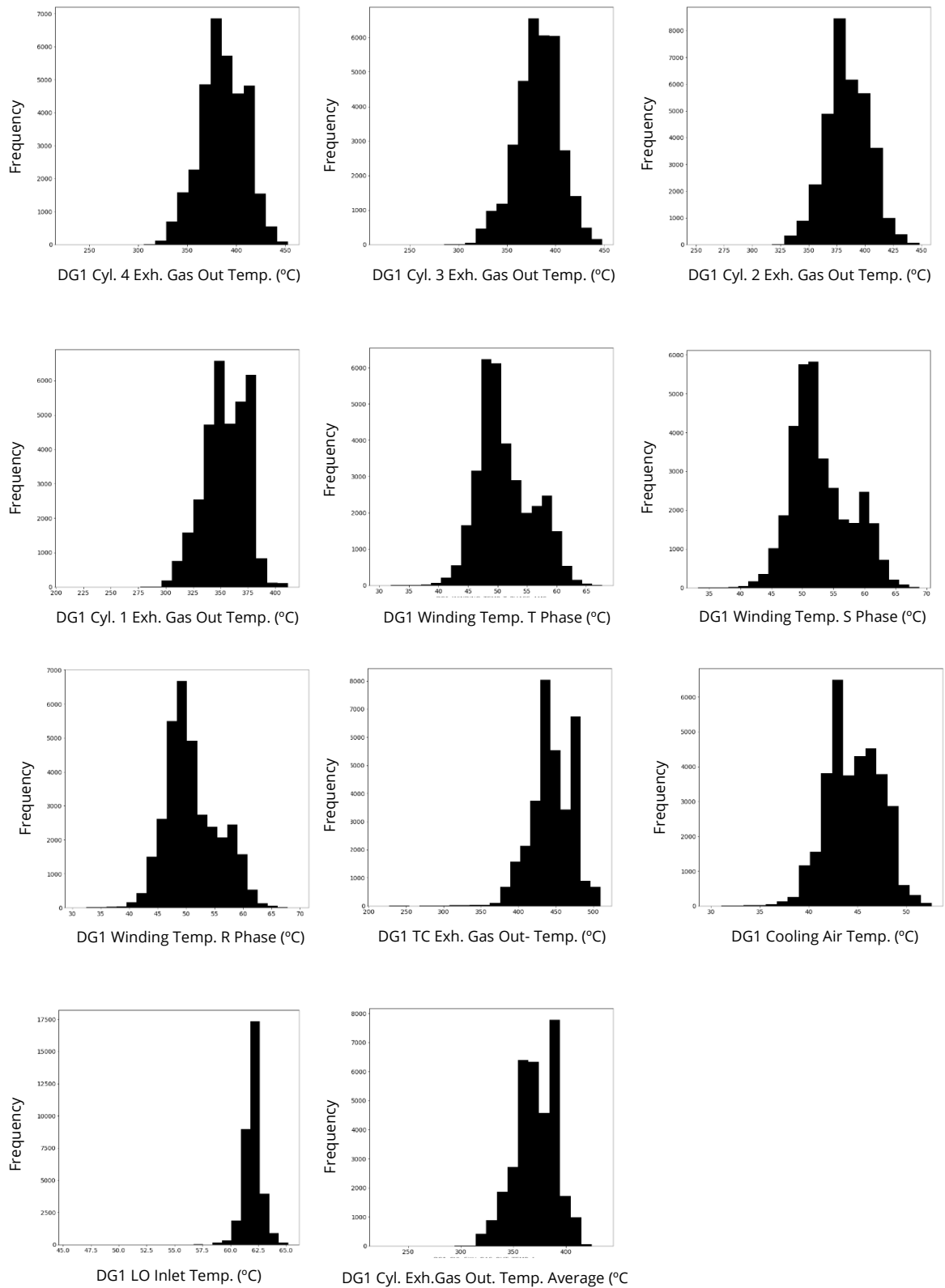


Fig. 6.21. Histograms of the monitored features (P1-P14).

The validation of the imputation performance is performed by considering five scenarios. Each of these scenarios refer to one of the five ratios of missing values analysed (0.05, 0.15, 0.25, 0.5, 0.8). Accordingly, three distinct contexts of missing values are assessed (small, medium, and large). The results of the comparative study are presented in Tables 6.16-6.19. As presented in the methodology section, a total of four models are analysed: 1) LSTM VAE-based regressor, 2) mean imputation, 3) application of Forward Fill and Backward Fill, and 4) k -NN. To avoid over-fitting, the number of neighbours selected for the k -NN imputer is set to the square root of the number of instances.

LSTM VAE-based regressor outperforms the remaining imputers. Nevertheless, analogous imputation performances are observed between the LSTM VAE-based regressors and the remaining parameters when the dataset contain small number of missing values. For instance, the RMSE of the LSTM VAE-based regressor and Forward Fill and Backward Fill (FF-BF) algorithms are 8.91 kW and 9.1 kW, respectively, when the missing ratio of 0.05 is considered. Conversely, the RMSE of the LSTM VAE-based regressor and FF-BF are 12.20 kW and 22.00 kW, respectively, when a large rate is considered (missing ratio of 0.8). Such a fact demonstrates the robustness of LSTM VAE-based regressor, whereas FF-BF only performs well when small ratios of missing values are considered. With regards to the k -NN imputer, it can be observed that this imputer also performs well when medium ratios of missing values are considered, thus being more robust than the FF-BF approach. Nevertheless,

several limitations of such an approach are worth highlighting. An example of this is the degradation performance when the sample contains large gaps of missing values and when both dimensions and number of instances are high. Also, the number of neighbours need to be optimally selected.

The worst results are achieved when the mean imputation is applied. This is when the maximum RMSE is perceived (when the missing ratio is 0.8 the RMSE is estimated to be 108.07 kW). The variability of the data and the different operational states identified within the dataset may be the cause of such results. As results did not vary significantly when applying this methodology, this has not been included in the analysis described in Fig. 6.22. Nevertheless, an example to observe the distortion of the parameter is expressed in Fig. 6.23. The study of performing mean imputation in each operational state can be implemented instead of estimating the mean for the entire dataset, thus determining if the bias and the limitations of such an approach still occur.

Table 6.15. Descriptive statistics of the monitored features.

	P 1	P 2	P 3	P 4	P 5	P 6	P 7
Count	33745	33745	33745	33745	33745	33745	33745
Mean	296.93	370.61	349.95	386.52	382.16	384.43	354.64
Std.	81.09	20.95	18.31	22.79	22.43	18.91	19.12
Min.	150.19	222.7	213.5	226.5	219.9	251.9	209.4
25%	236.97	355.6	338.9	371.6	368.2	372.1	341.6
50%	270.94	369.3	350.8	386	383.1	383.1	354.9
75%	356.14	385.8	362.2	404.1	397.5	399.5	371.6
Max.	555.93	438.7	421.1	453.1	448.1	448.5	411.7

	P 8	P 9	P 10	P 11	P 12	P 13	P 14
Count	33745	33745	33745	33745	33745	33745	33745
Mean	51.30	52.89	51.06	444.82	44.59	62.00	371.39
Std.	4.612	4.79	4.67	28.64	2.72	0.84	19.36
Min.	30.2	33.2	30.7	213.7	29.9	45.6	223.9
25%	48.1	49.5	47.8	429.1	42.6	61.7	359.66
50%	50.3	51.9	50	443.5	44.5	62.1	371.06
75%	54.3	55.9	54.2	469.1	46.7	62.5	388.16
Max.	67.7	68.8	69.6	510.1	52.6	65.2	434.55

Table 6.16. Pearson's correlation coefficient (absolute values).

Parameter	Coefficient
P2	0.88
P3	0.82
P4	0.87
P5	0.80
P6	0.84
P7	0.81
P8	0.86
P9	0.87
P10	0.86
P11	0.88
P12	0.77
P13	0.51
P14	0.89

When the imputation performance is assessed in a general manner, it can be perceived that the performance decreases when the missing ratio increases. A major cause of such a fact may be the reduction of the observed instances that are utilised for training purposes. Accordingly, this indicates the importance of preventing errors that may lead to corrupted or incomplete data. Additionally, should an imputation method need to be applied due to the impossibility of preventing these errors, a comprehensive analysis needs to be performed to avoid adding bias estimates that may lead to an inaccurate model that could ultimately be utilised for decision-making strategies.

Despite the undeniable performance of the LSTM VAE-based regressor, there are certain limitations that cannot be diminished. For example, the imputation performance of this type of methods relies on the amount of data available. In addition, the computation cost is higher than other analysed methodologies that presented similar imputation results in specific contexts. Also, these models present a lack of transparency and flexibility. The risk of obtaining over-fitting models is also a concern. All these preceding limitations may develop a lack of trust towards these models within the sector if these are not adequately addressed.

Table 6.17. Imputation results of the LSTM-VAE-based regressor.

Missing ratio	RMSE (kW)	MSE (kW ²)	Max. Error (kW)	MedAE (kW)	MAE (kW)	R ²
0.05	8.91	79.47	41.48	5.33	6.69	0.99
0.15	9.80	96.12	196.66	4.72	6.48	0.98
0.25	10.10	102.13	144.23	4.98	6.86	0.99
0.5	10.74	115.39	153.57	5.06	7.23	0.98
0.8	12.20	148.88	196.06	7.29	8.87	0.98

Table 6.18. Imputation results of the k -NN imputer.

Missing ratio	RMSE (kW)	MSE (kW ²)	Max. Error (kW)	MedAE (kW)	MAE (kW)	R ²
0.05	14.68	215.51	69.41	7.97	10.92	0.97
0.15	15.26	232.89	70.68	8.51	11.44	0.97
0.25	15.89	252.56	151.11	8.64	11.6	0.96
0.5	16.22	262.96	149.24	8.88	11.97	0.96
0.8	18.31	335.37	150.91	10.87	13.72	0.95

Table 6.19. Imputation results of the application of Forward Fill and, subsequently, Backward Fill algorithms.

Missing ratio	RMSE (kW)	MSE (kW ²)	Max. Error (kW)	MedAE (kW)	MAE (kW)	R ²
0.05	9.1	82.9	42.61	4.68	6.53	0.99
0.15	10.89	118.69	107.46	4.99	7.28	0.98
0.25	12.2	148.72	107.46	5.27	7.91	0.98
0.5	16.89	285.18	221.99	6.24	9.58	0.96
0.8	22.00	484.29	262.13	8.52	13.59	0.94

Table 6.20. Imputation results of the mean imputation technique.

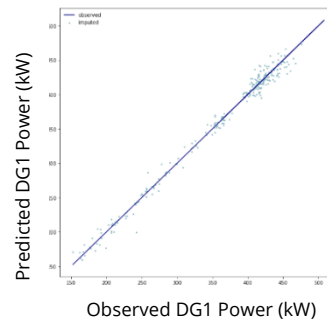
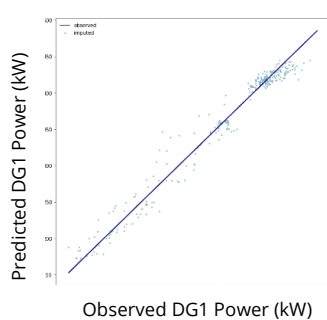
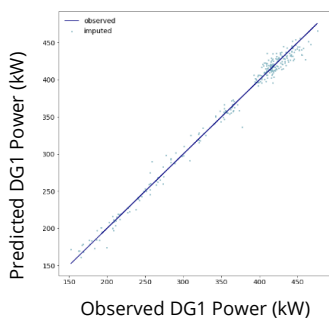
Missing ratio	RMSE (kW)	MSE (kW ²)	Max. Error (kW)	MedAE (kW)	MAE (kW)	R ²
0.05	107.87	11635.41	180.1	113.53	100.16	0
0.15	107.68	11595.06	181.68	113.63	99.95	0
0.25	107.89	11641.99	181.68	113.44	100.39	0
0.5	108.47	11764.97	189.86	113.75	101.14	0
0.8	108.07	11679.65	189.42	112.89	100.66	0

LSTM-VAE-based regressor

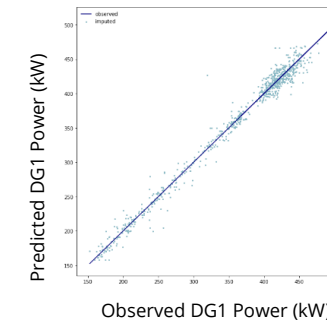
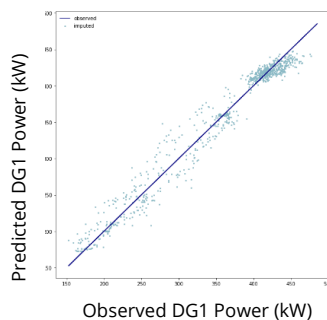
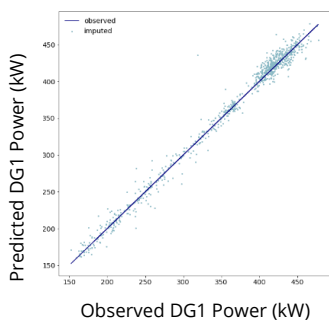
***k*-NN imputer**

Forward Fill and Backward Fill

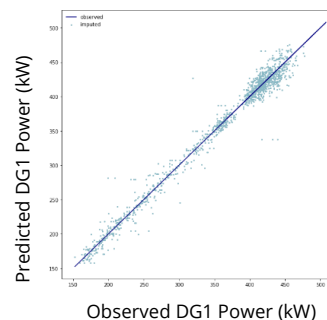
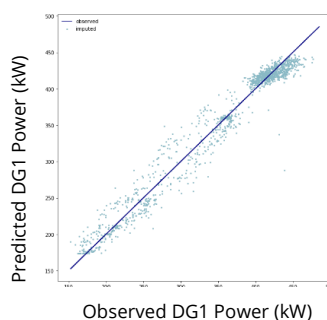
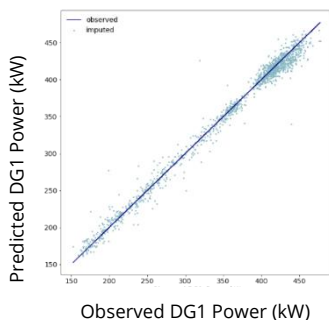
0.05



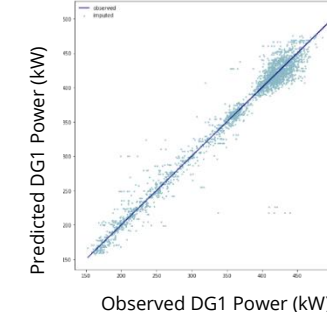
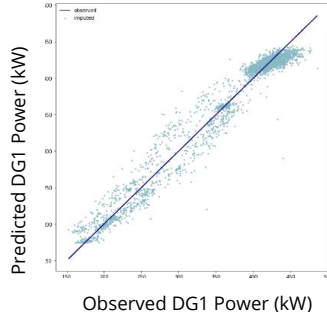
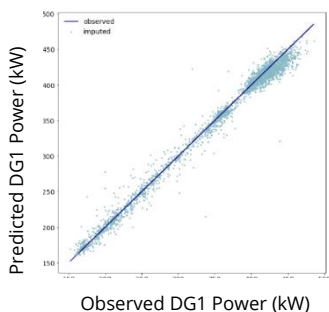
0.15



0.25



0.50



0.80

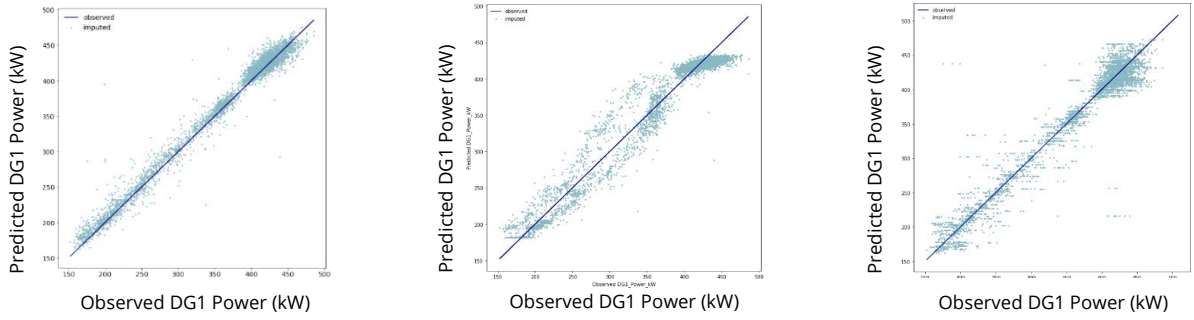


Fig. 6.22. Comparison between observed values and imputed values of the diesel generator power (the line refers to the observed values, whereas the scatter points refer to the imputed values).

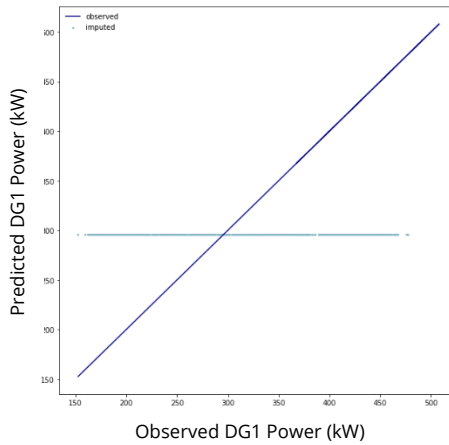


Fig. 6.23. Comparison between observed values and imputed values of the diesel generator power when the mean imputation is applied, and the missing ratio is 0.15 (the line refers to the observed values, whereas the scatter points refer to the imputed values).

6.5. Case Study 4. Novel Framework for the Identification of Steady States¹¹

Having explored the methodology being analysed to identify the different steady states widely observed when dealing with critical marine machinery in section 4.5. *A Novel Framework for the Identification of Steady States*, a case study is presented to assess its performance. As such, the power parameter collected from a total of three diesel generators (DG1, DG2, DG3) of a tanker ship is considered.

This parameter includes more than 65,000 instances and has been collected in a 1-minute frequency (please see Figs. 6.24-6.26). The descriptive statistics are presented in Table 6.21.

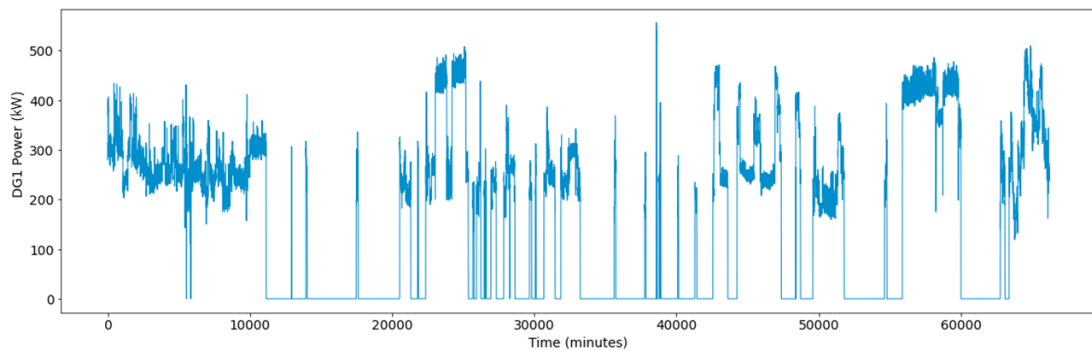


Fig. 6.24. DG1 power parameter time series plot.

¹¹ The operational states' identification framework presented in this section has been already converted to a journal paper, and has been published in the Applied Ocean Research journal (Velasco-Gallego and Lazakis, 2022e).

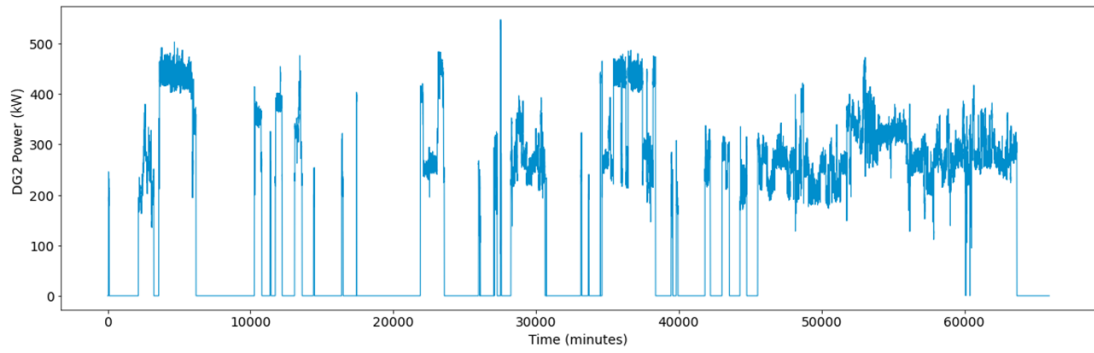


Fig. 6.25. DG2 power parameter time series plot.

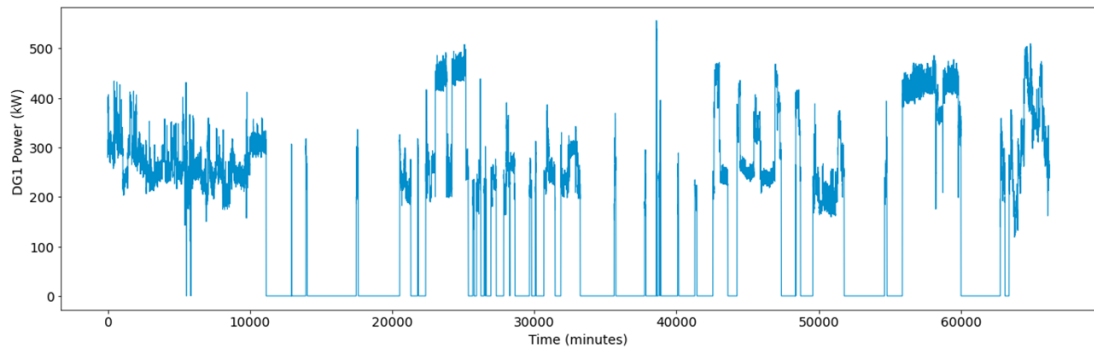


Fig. 6.26. DG3 power parameter time series plot.

Table 6.21. Descriptive statistics of the monitored parameters.

	DG1 Power (kW)	DG2 Power (kW)	DG3 Power (kW)
Count	66207	65947	65943
Mean	151.67	151.41	227.24
Std.	159.15	157.56	176.99
Min.	0.00	0.00	0.00
25%	0.00	0.00	0.00
50%	177.95	183.95	261.22
75%	273.30	277.85	373.93
Max.	555.93	546.76	597.86

The time series present distinct typologies of states that need to be identified and address accordingly. Examples of these are idle, transient, and operation states of machinery. Accordingly, this time series can be considered for validating the introduced methodology.

Prior to applying the steady states' identification stage, data pre-processing needs to be implemented. The sliding window algorithm and the EWMA are implemented in this phase. After heuristic evaluation, the configuration of the sliding window algorithm is set to present a time step of 1 and a sequence length of 60.

Once performed the data pre-processing phase, the steady state identification stages (image generation, connected component analysis, and data post-processing) are performed. The results obtained after the application of such stages are expressed in Figs. 6.27-6.29. Table 6.22 describes the different instances that are further described in this section.

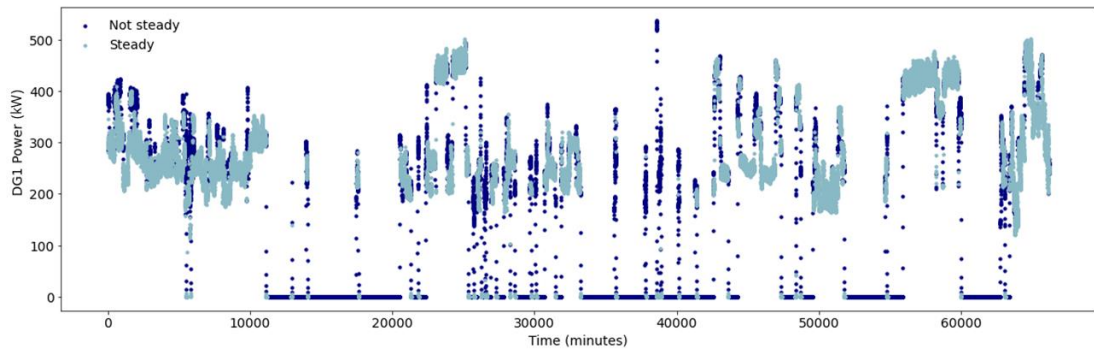


Fig. 6.27. Steady states' identification for the DG1 power parameter.

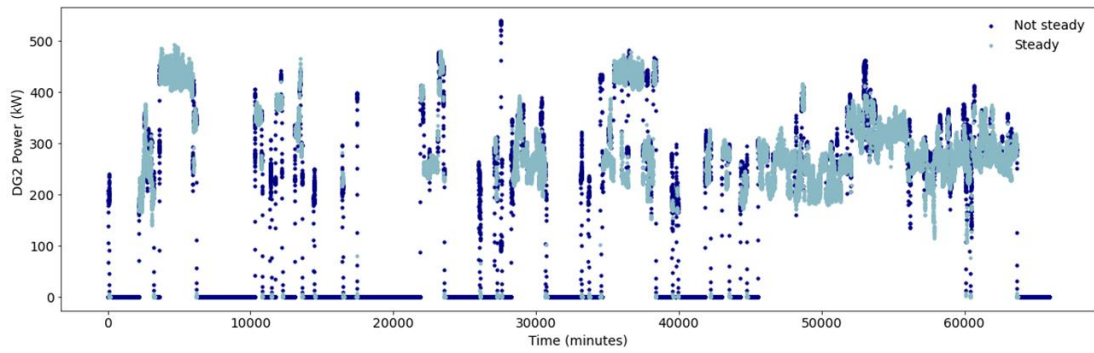


Fig. 6.28. Steady states' identification for the DG2 power parameter.

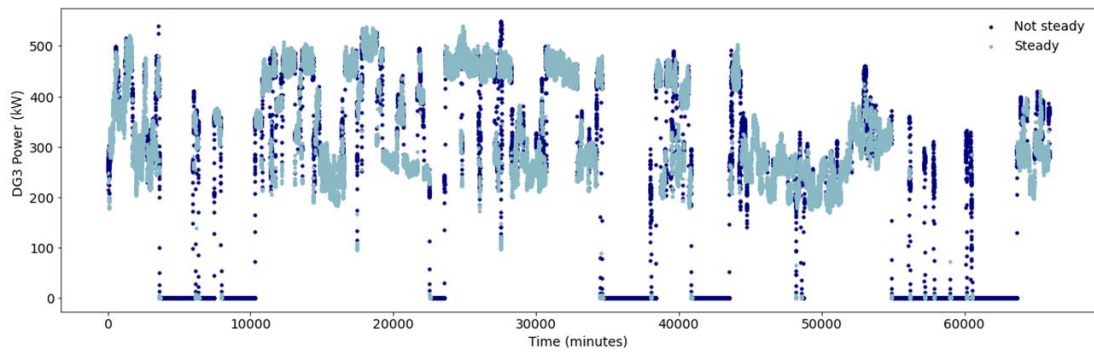


Fig. 6.29. Steady states' identification for the DG3 power parameter.

Table 6.22. Sequence selection for visual analysis.

	Starting sequence instance	Ending sequence instance	Total of instances in sequence
DG1 sequence	46,000	47,400	1,400
DG2 sequence	35,400	37,750	2,350
DG3 sequence	24,000	26,500	2,500

As presented in Fig. 6.30, the introduced methodology identified the states presented in the DG1 sequence effectively. The first state initiates and persists over half of the recorded time. This is when the values are stabilised between 200 and 300 kW. Then, an abrupt change is perceived to enable the transition between states. This transition is identified and labelled accordingly. The second state remains for approximately 200 minutes until another transition occurs. Such a transition is also identified accordingly. The values of the third state are stabilised around the 300 kW. The last state is achieved when an abrupt adjustment is again originated. This last adjustment is also identified accordingly.

Similarly, the same approach and results are employed and obtained in the remaining sequences (please see Fig. 6.31 and Fig. 6.32). Nevertheless, in these sequences there are several instances, the typology of which is unclear. These instances usually occur when the transition between states is originated. Thus, these instances can be easily discarded from further analysis through the application of filters.

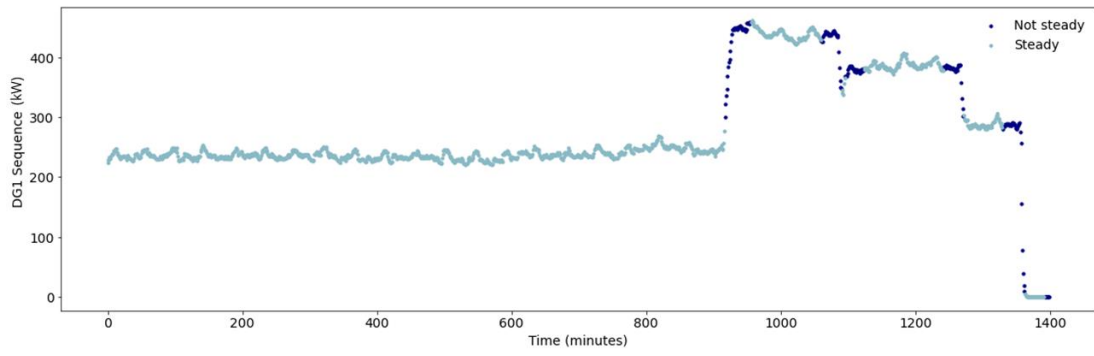


Fig. 6.30. Steady states' identification for DG1 power parameter sequence based on the results obtained from the proposed methodology.

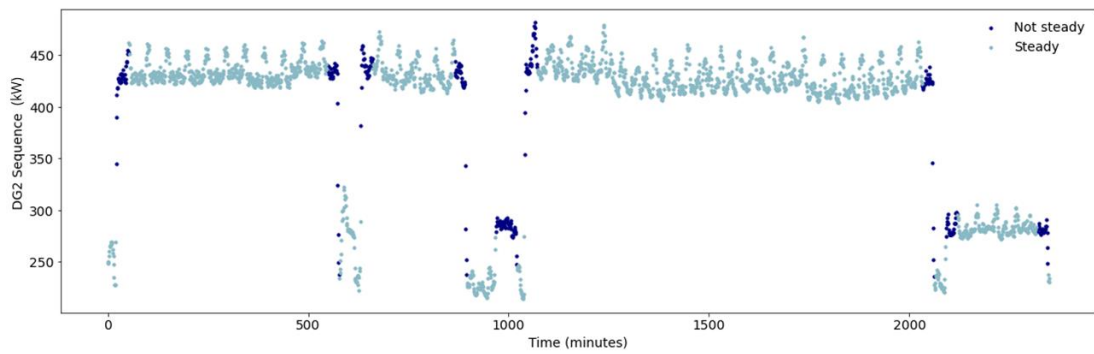


Fig. 6.31. Steady states' identification for DG2 power parameter sequence based on the results obtained from the proposed methodology.

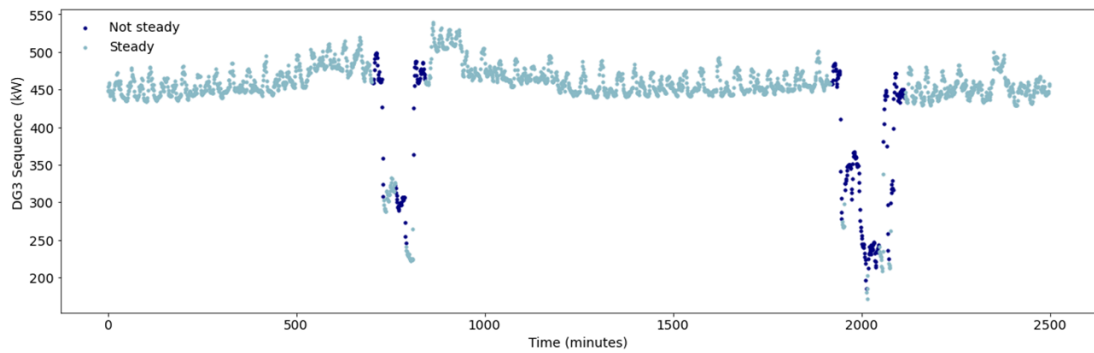


Fig. 6.32. Steady states' identification for DG3 power parameter sequence based on the results obtained from the proposed methodology.

To compare the performance of the proposed methodology a comparative study is applied. Accordingly, both k -means and GMMs with EMM are employed. These methods have been precedingly utilised in similar studies within the shipping domain. k -means has been successfully applied in section 6.3. *Case Study 2. Hybrid Imputation Framework* when dealing with short-term time series data collected from a main marine engine. To adequately select the optimal number of clusters, both the Silhouette and Davies-Boulding indices are estimated. The outlined results from the application of k -means are expressed in Figs. 6.34-6.36.

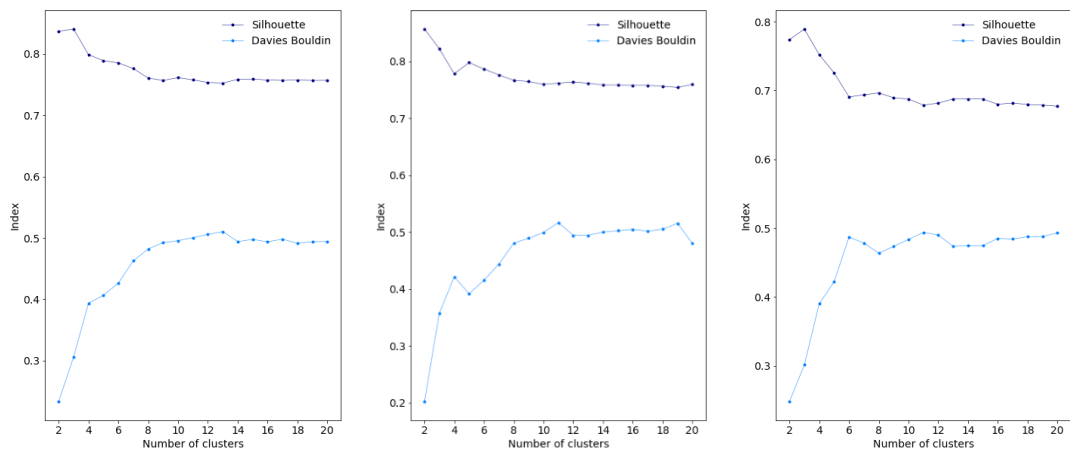


Fig. 6.33. Estimation of the Silhouette and Davies-Boulding indices for DG1, DG2, and DG3 power parameter, respectively.

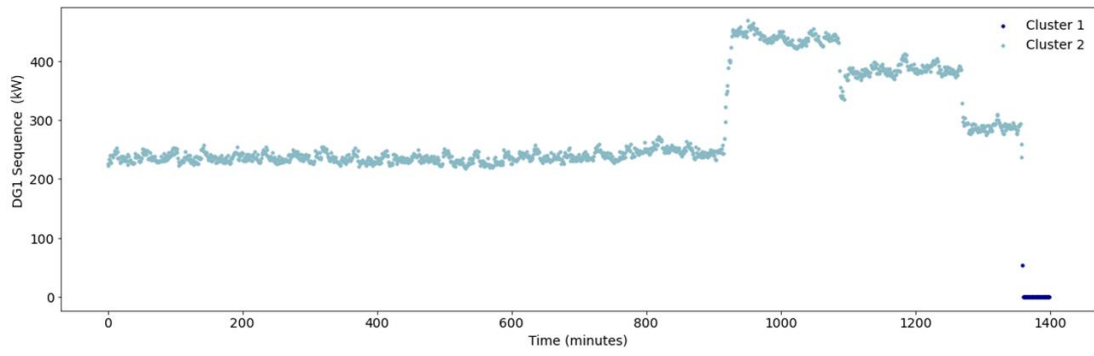


Fig. 6.34. Steady states' identification for DG1 power parameter sequence based on the results obtained from the k -means application.

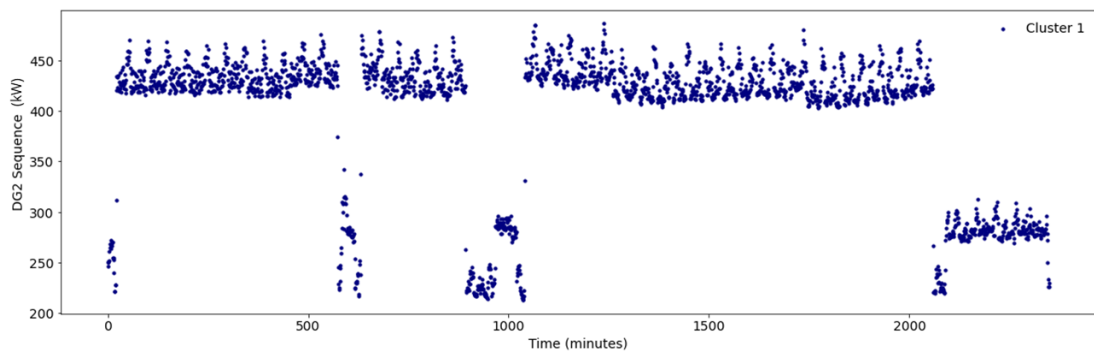


Fig. 6.35. Steady states' identification for DG2 power parameter sequence based on the results obtained from the k -means application.

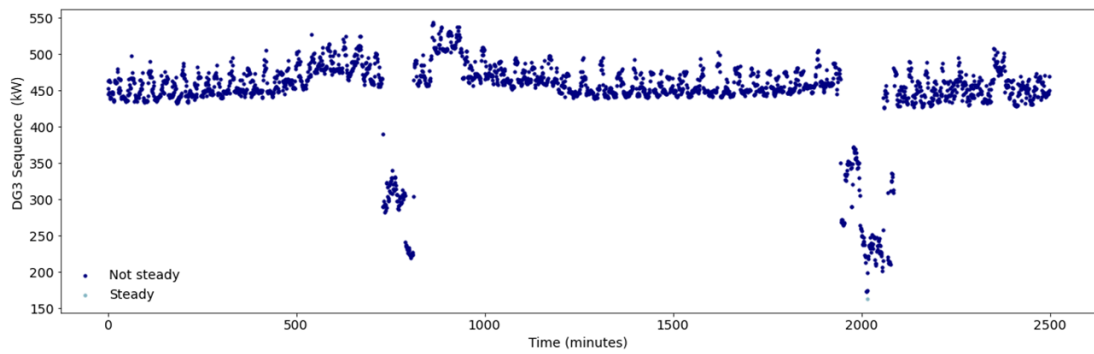


Fig. 6.36. Steady states' identification for DG3 power parameter sequence based on the results obtained from the k -means application.

The number of clusters selected in all scenarios is two, as presented in Fig. 6.33. One of the identified states relates to the idle states, whereas the other cluster refers to the remaining group. Thus, as indicated in Fig. 4.34, the distinct operational states are not adequately distinguished, whereas the idle states are clearly identified. Conversely, in the remaining sequences (please see Figs. 6.34-6.36) none of the states can be adequately differentiated. Also, a differentiation in the labelling can be also observed between the proposed methodology and the current approach, and thus the distinct steady states cannot be automatically detected. Consequently, expert knowledge is then required to label the identified clusters as defined originally (steady, and not steady). Thus, the application of k -means is not feasible when dealing with long-term time series data. Nevertheless, several adjustments in the approach can be made to enhance its performance. For example, the sliding window algorithm can be considered to only consider short-term time series data. The performance of multiple iterations in each of the groups of data identified may be another potential possibility to be worth investigating.

Analogous results are presented when considering the GMMs with EM approach (please see as Figs. 6.37-6.40). A total of two hyperparameters need to be selected in this respect: 1) type of covariance, and 2) number of components. With regards to the type of covariance, four types are analysed in accordance with Pedregosa et al. (2011): 1) full, 2) tied, 3) diagonal, and 4) spherical. A range between 1 and 10 (inclusive) of mixture models are considered for the selection of the number of components. As

stated in Fig. 6.37, the number of components are set to three in all cases. The type of covariance is set to spherical. Thus, the distinct operational states are more effectively selected than the k -means (please see Figs. 6.38-6.40). Nevertheless, analogous to k -means, the steady states are not adequately labelled. Furthermore, the transition between states is not properly defined either.

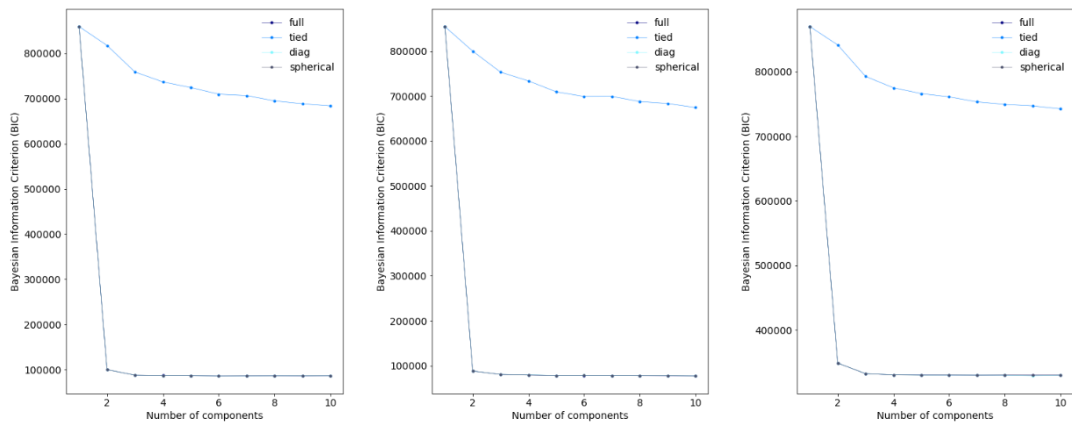


Fig. 6.37. Selection of the number of components for DG1, DG2, and DG3 power parameter, respectively.

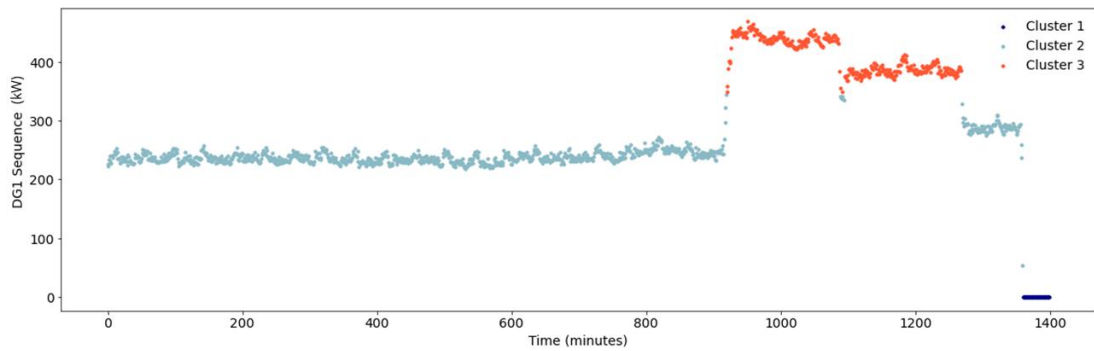


Fig. 6.38. Steady states' identification for DG1 power parameter sequence based on the results obtained from the GMMs with EM algorithm application.

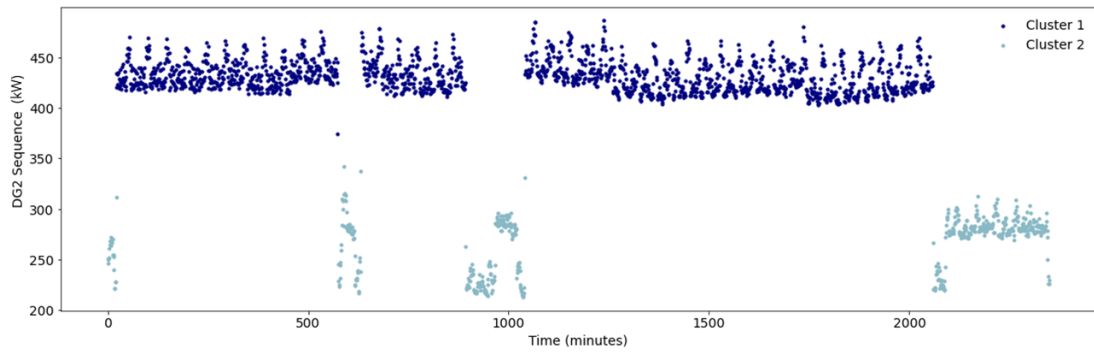


Fig. 6.39. Steady states' identification for DG2 power parameter sequence based on the results obtained from the GMMs with EM algorithm application.

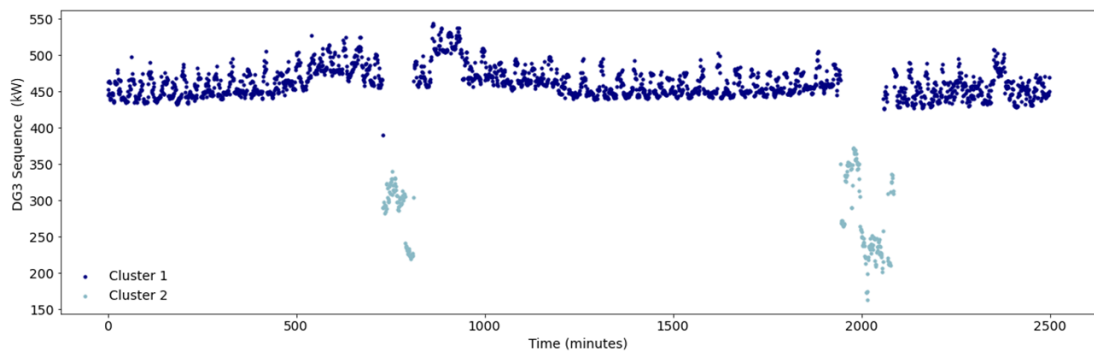


Fig. 6.40. Steady states' identification for DG3 power parameter sequence based on the results obtained from the GMMs with EM algorithm application.

Accordingly, the introduced methodology outperforms the other analysed approaches for automatically identifying and differentiating the steady states. Thus, by excluding from the analysis non-operational profiles, an increase in both the system performance and computational efficiency are expected, thus promoting an enhancement in the decision-making strategies.

6.6. Chapter Summary

This chapter has introduced a summary of four case studies performed to validate each of the methodologies presented in the data pre-processing module. Based on the results obtained, it can be perceived the effectiveness of applying novel data-driven methodologies to ensure data quality. Specifically, the developed data-driven methodologies for data imputation have demonstrated its capability of imputing missing values to avoid biased results and under-utilisation of data. With regards to the proposed method for the identification of operational states, it can be perceived that the model can differentiate in an adequate manner operational states, and both transient and idle states. Thus, the computational cost is decreased while the efficient and effective performance of the model is ensured through the consideration of only operational states in the analysis.

Chapter 7

Case Studies and Results: Part II. The Diagnostic and Predictive Analytics Modules

7.1. Chapter Overview

A total of 3 case studies are introduced in this chapter, each of one referring to one of the novelties presented in the diagnostic and predictive analytics module. An additional case study is introduced to validate the performance of the holistic MA framework. It is worth highlighting that to assess the generalisation capabilities of the introduced modules and methodologies, each case study is unique and considers to different time series sequences, even though some case studies may refer to the same system.

7.2. Case Study 5. Fault Detection as Part of the Diagnostic Analytics Module¹²

Case study 5 aims to validate the methodology introduced in section 5.3.1. *Fault Detection*. Specifically, a Diesel GenSet of a tanker ship used for auxiliary needs is considered. This is a four-stroke in-line engine, comprised of a total of 6 cylinders.

The following parameters are analysed in this study: the diesel generator power, the exhaust gas outlet temperature of each cylinder, the winding temperature (phases T, S, and R), the turbocharger exhaust gas outlet temperature, the cooling air temperature, the lubricating oil inlet temperature, and the cooling fresh water in pressure are analysed.

More than 66,000 instances are analysed for each parameter. Each of these instances has been collected in a 1-minute frequency basis (please see Fig. 7.1). The descriptive statistics are presented in Table 7.1.

¹² The fault detection framework presented in this section has been already converted to a journal paper, and has been published in the Expert Systems with Applications journal (Velasco-Gallego and Lazakis, 2022d).

Table 7.1. Descriptive statistics of the monitored parameters.

Id	Parameter Name	Mean	Std.	Min.	25%	50%	75%	Max.
0	Power (kW)	151.67	159.15	0	0	177.9	273.2	555.9
1	CYL6 EXH GAS OUT TEMP (°C)	222.68	153.00	48.1	64.3	329.7	370	438.7
2	CYL5 EXH GAS OUT TEMP (°C)	209.99	144.67	47.6	58.4	309.8	351.3	421.1
3	CYL4 EXH GAS OUT TEMP (°C)	231.52	160.16	48.5	65.7	336.8	386.5	453.1
4	CYL3 EXH GAS OUT TEMP (°C)	230.16	157.23	48.7	67	330.7	383.8	448.1
5	CYL2 EXH GAS OUT TEMP (°C)	231.68	158.10	48.5	67.3	344.9	383.3	448.5
6	CYL1 EXH GAS OUT TEMP (°C)	215.61	143.68	47.1	67.5	312.1	355.4	411.7
7	WINDING TEMP T PHASE (°C)	44.88	8.34	26.6	39.5	45.8	50.5	67.7
8	WINDING TEMP S PHASE (°C)	46.11	8.80	26.7	40.6	47.3	51.9	68.8
9	WINDING TEMP R PHASE (°C)	44.53	8.55	25.5	39.3	45.4	50.2	69.6
10	TC EXH GAS OUT TEMP (°C)	263.94	190.23	31.6	49.4	384.7	444.2	510.1
11	COOLING AIR TEMP (°C)	39.36	6.46	25.5	35.2	40.6	44.7	52.6
12	LO INLET TEMP (°C)	57.30	6.62	31.8	51.4	61.3	62.1	68.4
13	CFW IN PRESS (bar)	2.91	1.20	0	1.7	3.8	4.1	4.5

As observed in preceding case studies, time series sensor data of marine machinery presents various states that need to be identified and addressed accordingly. For this reason, the methodology presented in section 4.5. *A Novel Framework for the Identification of Steady States* is implemented. After heuristic evaluation, the transition matrix is estimated to present a total of 25 states. The outcome of this phases when analysing the diesel generator power parameter is presented in Fig. 7.2. In total, 81 operational sequences have been identified. Each of these sequences have been further analysed to either accept or reject them for the training, validation, and test stages. Also, as part of the pre-processing step, data normalisation is applied so that each value lies between -1 and 1.

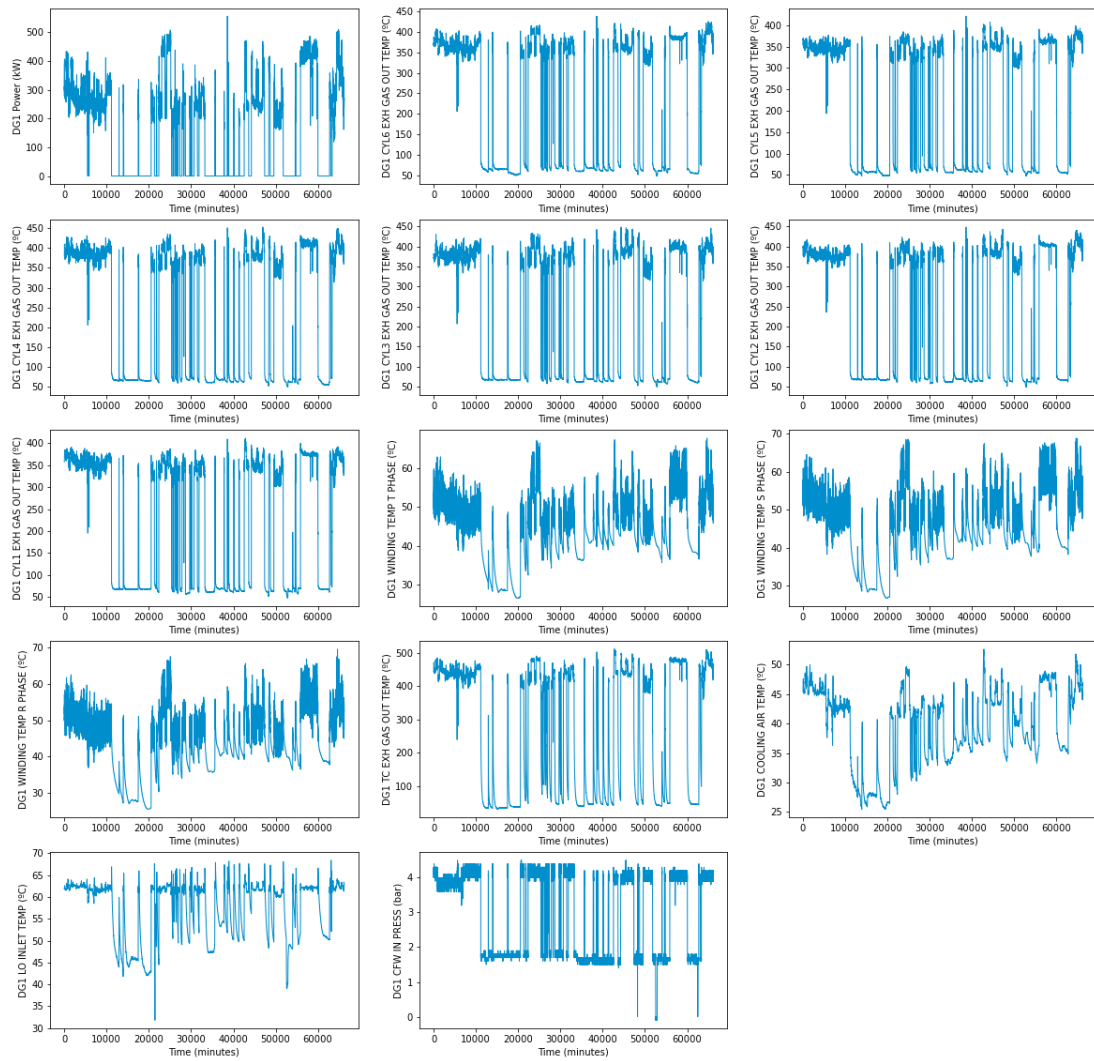


Fig. 7.1. Time series plot of the fourteen monitored parameters.

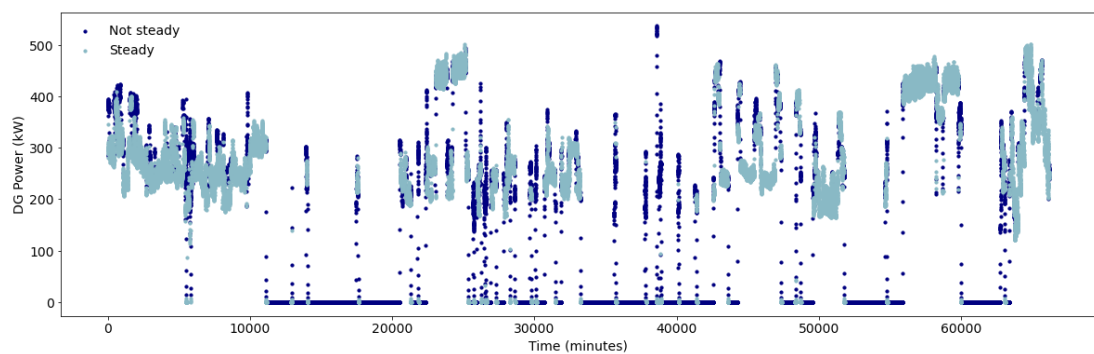


Fig. 7.2. Steady states' identification for the diesel generator power parameter.

Once the data is pre-processed, the LSTM-based VAE is implemented as part of the time series reconstruction stage. To select the distinct hyperparameters the grid search algorithm in conjunction with the k -fold cross-validation are applied. The hyperparameters to be selected relate to the following: 1) the number of layers in both encoder and decoder and their respective units, 2) the number of latent dimensions, and 3) the length of the sequences. The search space is defined in Table 7.2.

Based on the result obtained from the hyperparameters optimisation stage, the architecture of the LSTM-based VAE is defined. The LSTM of both the encoder and decoder of the VAE-LSTM is formed by 1 layer (129 units) and tanh activation function. The number of latent dimensions is set to 6 and the sequence length to 3. Furthermore, the ratio of the validation set has been set to 0.20. Adam optimizer has been applied to compile such a model. As part of the training process, the number of epochs is set to 100 and the batch size to 5.

Table 7.2. Range of values analysed for hyperparameter optimisation.

Hyperparameter	Range
Number of layers in both encoder and decoder	(1, 2)
Number of units per layer	(2, 256)
Number of latent dimensions	(1, 24)
Length of sequences	(3, 180)

As all the analysed sequences contain normal instances, anomalies need to be generated. Accordingly, noise is injected by considering distinct Gaussian distributions

of diverse mean levels, as analogous performed by Zhao et al. (2019). In this study, parameter 4 is considered to simulate the anomalous scenario.

The average execution time of RADIS is 0.68 seconds per instance (0.05 seconds per parameter). The hardware utilised consists of an Intel(R) Core (TM) i7-4790 CPU @ 3.60GHz 3.60 GHz and Windows 10. A total 75 simulations are applied for each test sequence to guarantee the generalisation capabilities. For the training process, a total of 65 sequences identified in the steady states' identification phase are considered, a total 20% of them being considered for validation purposes. The average NRMSE with a Confidence Interval (CI) of 95% obtained for each parameter is presented in Fig. 7.3. Furthermore, to complement such results and enhance visibility, the NRMSE obtained at each instance of the sequence for parameters 0 (power), 4 (exhaust gas outlet temperature of cylinder 3), which has been altered to simulate anomalies, and 9 (winding temperature at R phase), are also graphically represented in Fig. 7.4.

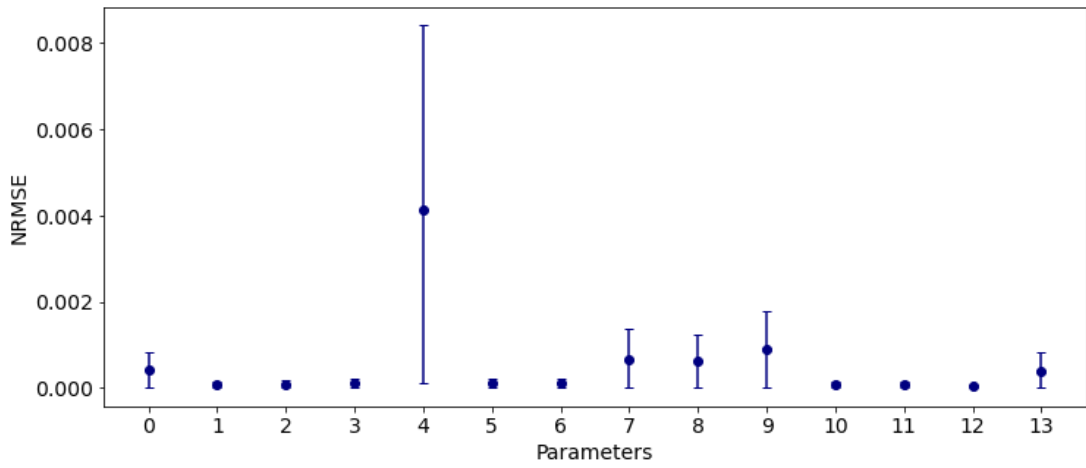


Fig. 7.3. Average NRMSE with CI 95% of the analysed parameters with injected anomalies in parameter 4.

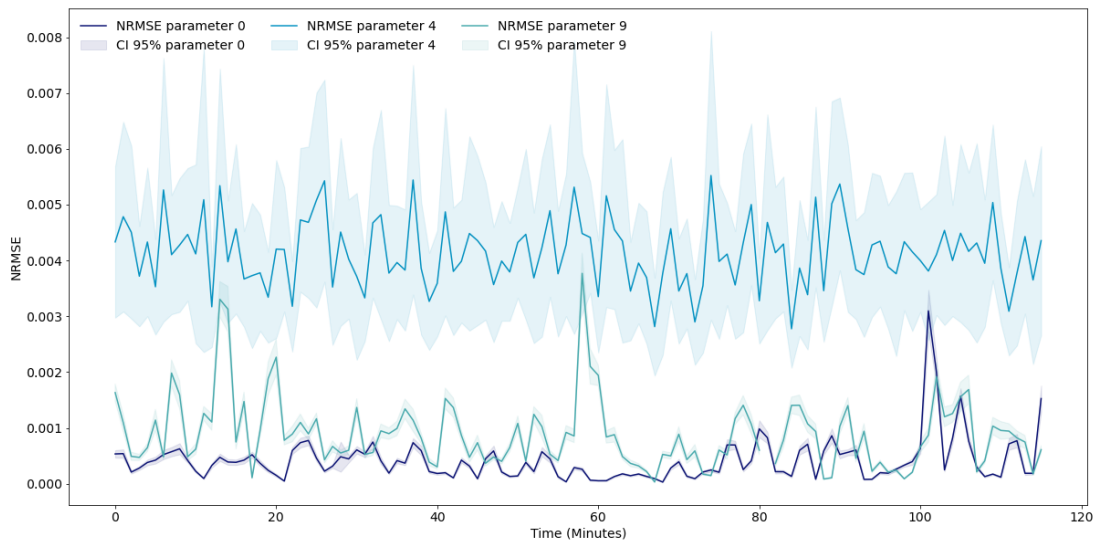


Fig. 7.4. NRMSE with CI 95% of parameters 0, 4, and 9 divided by instance with injected anomalies in parameter 4.

As stated in such figures, the injection of random noise to generate anomalies in parameter 4 caused an increment in the reconstruction error. Accordingly, the estimated NRMSE value for such a parameter is higher than the remaining analysed

features. Conversely, if the original parameter 4 is considered, it can be observed that the reconstruction error is analogous to the remaining parameters (please see Figs. 7.5-7.46).

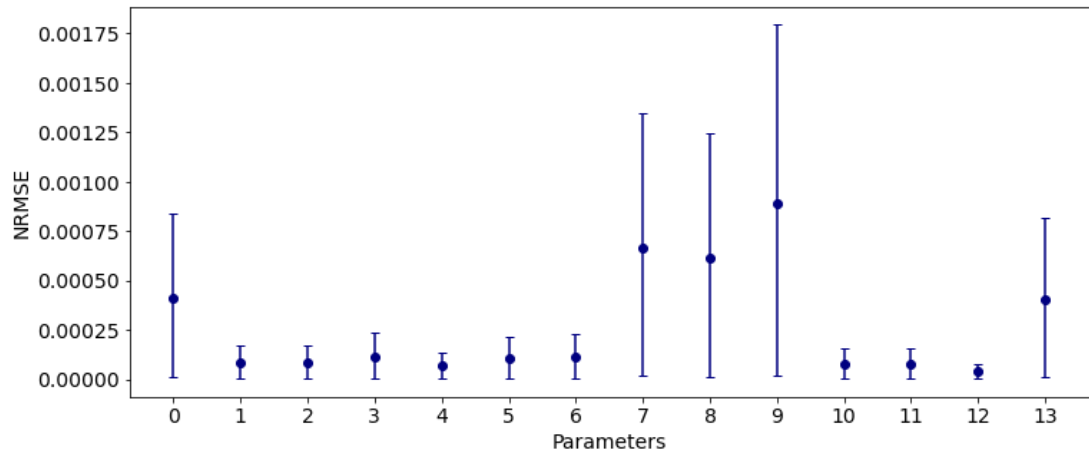


Fig. 7.5. Average NRMSE with CI 95% of the analysed parameters.

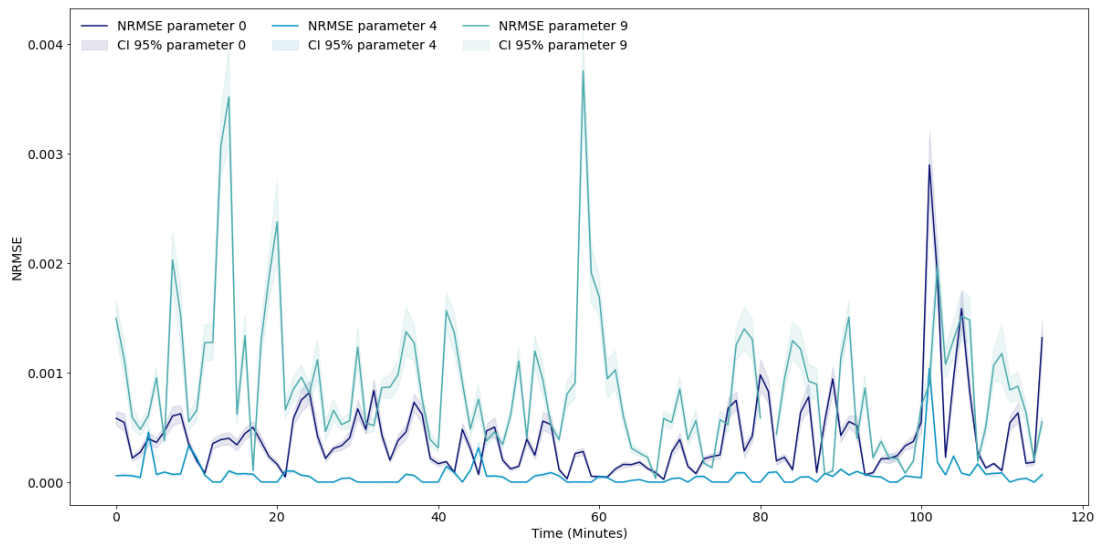


Fig. 7.6. NRMSE with CI 95% of parameters 0, 4, and 9 divided by instance.

After the time series reconstruction stage, the image generation and thresholding phase is implemented. The image generation is obtained by scaling the NRMSE values into the most common pixel format, the range of which lies between 0 and 255. Pixels with higher intensity are expected to refer to those parameters that present any anomalous behaviour, whereas those parameters with non-anomalous behaviours are expected to constitute the pixels with the lowest intensity of the image (see Fig. 7.7).

Consequently, image thresholding can be applied to not only automatically detect such anomalous values but also rank them depending on their level of anomalousness. Hence, possible relationships between parameters may be suggested for further analysis based on the resulting thresholding.

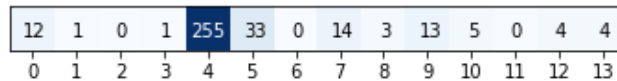


Fig. 7.7. Image generated for instance 0 and simulation 0.

Image thresholding is applied through the introduction of multi-level Otsu's method. The adequate number of classes is selected by employing the method GMMs with an EM algorithm. Accordingly, the Bayesian Information Criterion (BIC) is estimated. The initial study considers the range of components to lie between 1 and 10. Nevertheless, it has been perceived that this image thresholding method is unfeasible

for many classes. Accordingly, a bi-level thresholding task is performed instead. Therefore, C1 refers to anomalous instances, while C2 refers to normal instances. The selection of the number of classes for each sequence is applied by implementing the same approach. This process is applied to perform image thresholding in the resulting images at each specific instance of the sequence and for each simulation performed. After thresholding all the resulting images, the results are summarised. The average accuracy for this specific case study is 92.5%

7.3. Case Study 6. Fault Identification as Part of the Diagnostic Analytics Module¹³

This case study is presented to validate the methodology introduced in section 5.3.2. *Fault Identification.* For this study, a Diesel GenSet of a tanker ship is considered. This is a four-stroke in-line engine comprised of a total of 6 cylinders. The power parameter is the feature subject of study.

In total, more than 66,000 instances are considered, which have been collected in a 1-minute frequency basis (please see Fig. 7.8). The descriptive statistics are also presented in Table 7.3.

¹³ The fault identification framework presented in this section has been already converted to a journal paper, and has been published in the Ocean Engineering journal (Velasco-Gallego and Lazakis, 2022b).

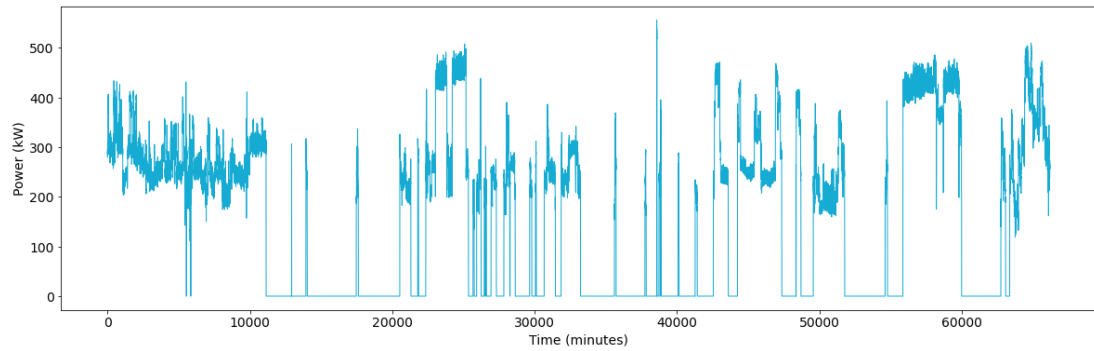


Fig. 7.8. Time series plot of the cooling air temperature monitored parameter.

Table 7.3. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
Power (kW)	151.67	159.15	0.0	0.0	177.95	273.30	555.93

As data pre-processing, steady states' identification and data normalisation are performed. Also, the sliding window algorithm is implemented. After heuristic evaluation, the window size parameter is set to 250.

Anomalies have been simulated due to the lack of fault data and complete degradation data. Therefore, to validate the proposed methodology, a total of six data patterns that do not refer to normal steady operational conditions have been simulated based on the detection of such patterns in analysed marine systems datasets. These are identified as point anomalies: 1) only one point anomaly is presented in the sequence,

2) two-point anomalies are identified in the sequence, 3) multiple point anomalies (>2) can be observed in the sequence, 4) collective anomalies, 5) degradation sequences, and 6) transitional occurrences between operational states.

Point anomalies refer to those instances that differ from others with regards to their attributes (Ramchandran and Sangaiah, 2018). An abrupt change in a steady operational state is an example of a point anomaly. This type of anomalies has been determined by obtaining the anomalous ratio and the instance in which the point anomaly is presented. The anomalous ratio establishes the intensity of the abnormality, and it is determined by selecting a random value from a pre-defined range.

By contrast, collective anomalies comprise a combination of anomalous instances. High variability in the exhaust gas outlet temperature parameter of one of the cylinders in a steady operational state context is just an example. Collective anomalies are simulated by injecting noise. This noise is generated by considering distinct Gaussian distributions of various mean levels, as analogous performed in Zhao et al. (2019c).

With regards to degradation patterns, an exponential model with Brownian motion is considered to simulate these due to its effective universality in machinery when

reflecting the characteristics of accelerated fault degradation in engineering (Li et al., 2021).

The last scenario refers to transitional occurrences between operational states that occur due to, for instance, environmental situations or variations in the operating condition (Theotokatos et al., 2020). This scenario is generated by dividing the entire sequence into distinct sub-sequences.

To complement the preceding explanation with regards to the distinct patterns simulated, a graphical representation of them can be perceived in Fig. 7.9. Fig. 7.10 presents the sequence of a diesel generator power parameter that has been altered to simulate such patterns.

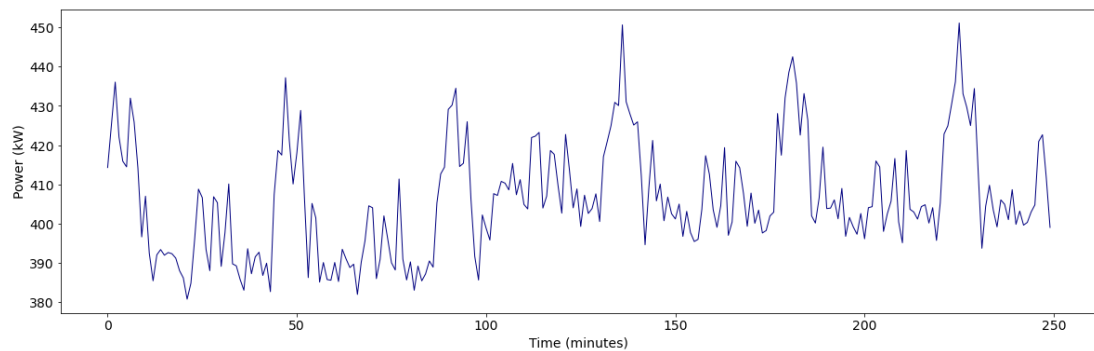


Fig. 7.9. Original sequence that has been altered to simulate the distinct abnormal operational sequences.

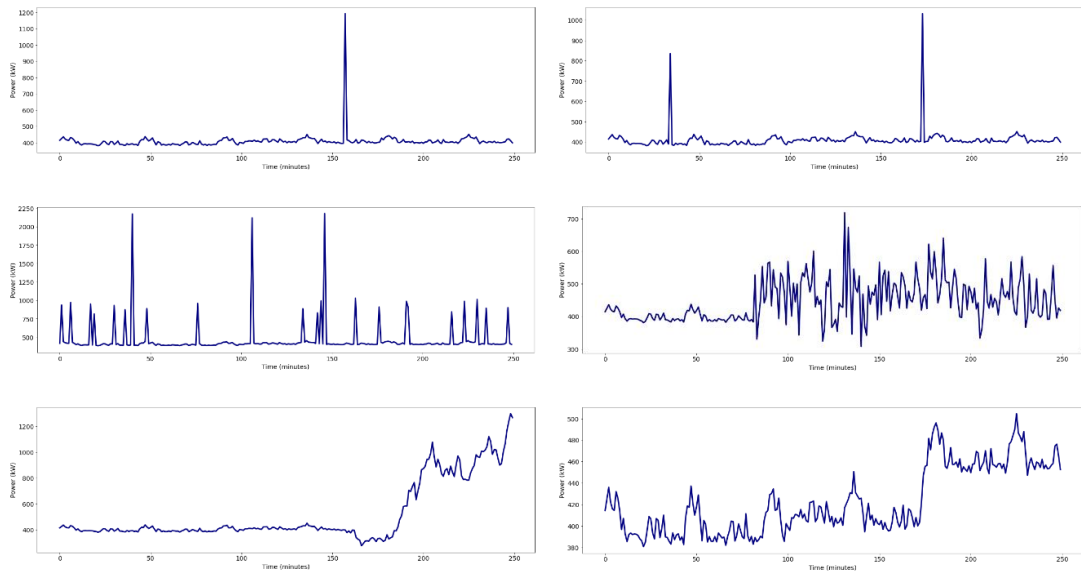


Fig. 7.10. Simulated sequence with (a) a point anomaly, (b) two-point anomalies, (c) multiple point anomalies, (d) collective anomalies, (e) degradation, and (f) transition occurrences between steady operational states.

After the data pre-processing and simulation stages, the time series are encoded into images so that the image classifier can be implemented. Accordingly, the transition matrix is estimated by considering the first-order Markov chain model. After heuristic evaluation, the number of states is set to 50, as values greater than 50 do not present an increase in accuracy, while the computational cost increases significantly. Also, the risk of over-fitting increases.

Figs. 7.11 and 7.12 graphically represents the images obtained after implementing the image generation stage. Even though the images represented in such figures contain a dimension of 10x10 for a better visual interpretation and description, the input images for this study present a resolution of 50x50, as it was stated in the preceding

paragraphs. An image generated from a normal operation sequence is represented in Fig. 7.11. A clear diagonal can be perceived in such an image. This suggests that when an operational state is considered their instances do not vary in a significant manner. Accordingly, the next state of an instance usually adopts the state of the preceding instance or one around it. Consequently, the diagonal is created. For example, if the current instance refers to state 2, it is highly probable that the next instance will relate to state 2 or a near state (e.g., 1 or 3). An analogous representation of this can be perceived in Fig. 7.12 (a). This image refers to a sequence that presents a point anomaly. Accordingly, the diagonal is shorter than when a normal operational image is considered, as the states are defined based on a different range of values. This does not apply when considering multiple point anomalies, as the relationship between states is distorted, thus disrupting the diagonal observed in normal images. Such a fact can be analogously perceived when collective anomalies and degradation images are being considered, as the large number of abnormalities alters the steady context. Nevertheless, if the transition occurrences between operational state context is considered it can be observed that the evolution is slightly different. For such an instance, two distinct diagonals can be perceived, each of them relating to each operational state. Isolated pixels are also represented, which relate to the transition between operational states. As indicated, each class presents unique characteristics. Thus, by applying image classifiers it is expected that the trained model can learn such characteristics, and thus identify the distinct defined categories.

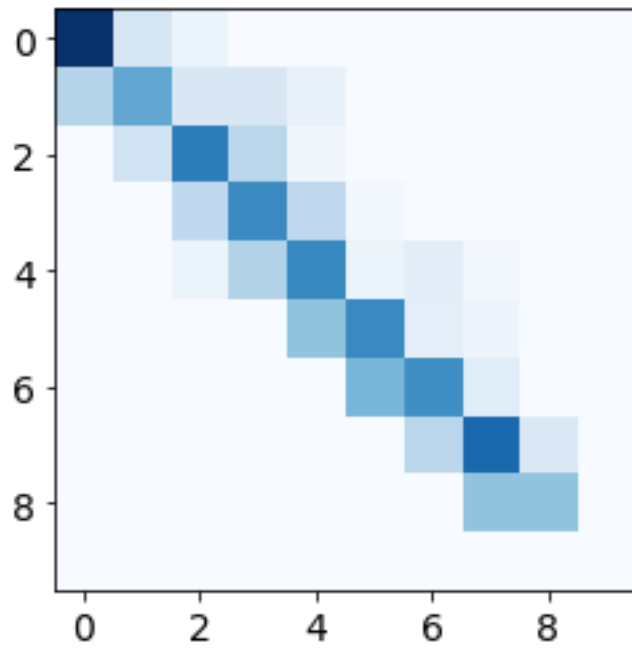


Fig. 7.11. Example of a normal sequence encoded into an image.

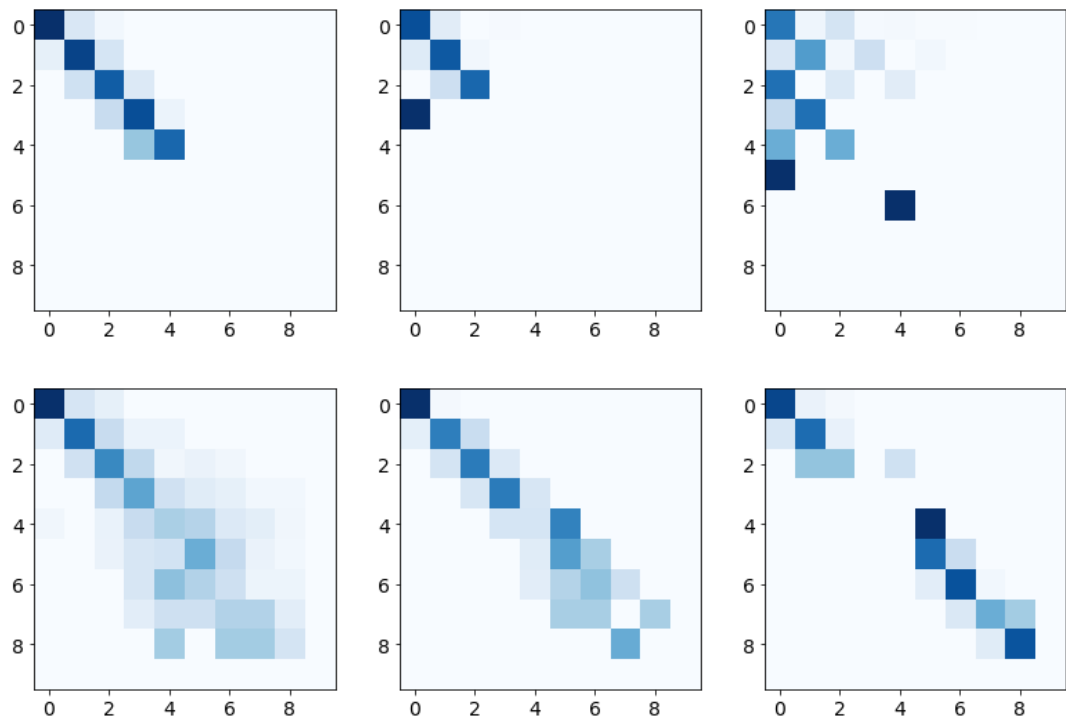


Fig. 7.12. Example of images generated that contain (a) a point anomaly, (b) two-point anomalies, (c) multiple point anomalies, (d) collective anomalies, (e) degradation, and (f) transition occurrences between steady operational states.

To validate the proposed methodology, a comparative study is applied. The first model considered is the 1D-CNN, as versions of such a model has presented promising results when dealing with time series data in analogous tasks, such as when predicting the RUL (Yao et al., 2021).

Additionally, to evaluate the performance of the first-order Markov chain model as time series imaging method, the proposed methodology (Markov-ResNet50V2) and the CNN model (Markov-CNN) are modified to present the GAF as the time series imaging method (GAF-ResNet50V2, and GAF-CNN). To encode time series into images by implementing GAF, the *pyts* package is utilised (Wang and Oates, 2015).

Based on comprehensive reviews of classification metrics, such as the one performed by Grandini et al. (2020), and to adequately assess the models included in the comparative study, a total of six metrics has been selected: 1) accuracy, 2) balanced accuracy, 3) Micro F1, 4) Macro F1, 5) Mattheus Correlation Coefficient (MCC), and 6) Cohen's Kappa.

Prior to the definition of such metrics, the confusion matrix needs to be defined, as some of the metrics are computed based on such a concept. This matrix can be defined

as a cross table that describes the number of occurrences between two rates (true/actual classification and predicted classification). A diagram representing a confusion matrix for multi-class classification is presented in Fig. 7.13.

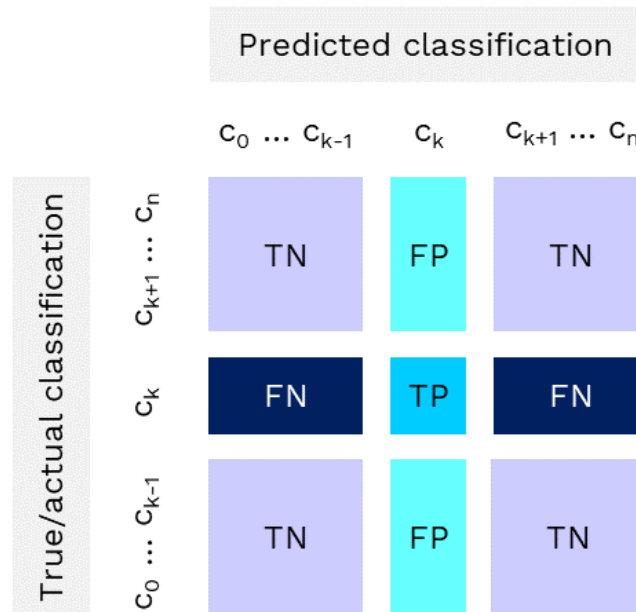


Fig. 7.13. Confusion matrix for multi-class classification with n classes. The estimation of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) is presented when considering a class k ($0 \leq k \leq n$).

Based on this concept, the first metric, accuracy, is defined. This is probably the most popular metric when addressing the multi-class classification task that considers all the elements of the confusion matrix (True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN)) as expressed in Eq. 52.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (\text{Eq. 52})$$

The balanced accuracy is another widely used metric, the estimation of which is also related with the confusion matrix. This metric can be defined as an average of Recalls, as, firstly, an evaluation of the Recall for each class is performed and, subsequently, the obtained values are averaged to determine the balanced accuracy score. The Recall is the fraction of True Positive elements divided by the total number of the actual positives (see Eq. 53).

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (\text{Eq. 53})$$

The Micro F1-Score is computed by estimating Micro-Precision (Eq. 54) and Micro-Recall (Eq. 55). The Micro-averaging is presented in this case to avert differences between classes. As the Micro-Average Precision and Recall refer to the same values, the Micro-Average F1-Score is equal to both Micro-Average Precision and Recall, as the harmonic mean of two equal values is just the value (see Eq. 56).

$$\text{Micro Average Precision} = \frac{\sum_{k=1}^K \text{TP}_k}{\text{Grand Total}} \quad (\text{Eq. 54})$$

$$\text{Micro Average Recall} = \frac{\sum_{k=1}^K \text{TP}_k}{\text{Grand Total}} \quad (\text{Eq. 55})$$

$$\text{Micro F1} = \frac{\sum_{k=1}^K \text{TP}_k}{\text{Grand Total}} \quad (\text{Eq. 56})$$

Analogously, the Macro F1-Score is determined by estimating the Macro-Precision (Eq. 57) and Macro-Recall (Eq. 58). The Macro F1-Score is then estimated by determining the harmonic mean of Macro-Precision and Macro-Recall (see Eq. 59).

$$\text{Macro Average Precision} = \frac{\sum_{k=1}^K \text{Precision}_k}{K} \quad (\text{Eq. 57})$$

$$\text{Macro Average Recall} = \frac{\sum_{k=1}^K \text{Recall}_k}{K} \quad (\text{Eq. 58})$$

$$\text{Macro F1} = 2 * \left(\frac{\text{Macro Average Precision} * \text{Macro Average Recall}}{\text{Macro Average Precision}^{-1} + \text{Macro Average Recall}^{-1}} \right) \quad (\text{Eq. 59})$$

The MCC is defined in terms of a confusion matrix C for K classes, as expressed hereunder.

$$\text{MCC} = \frac{c \times s - \sum_k p_k \times t_k}{\sqrt{(s^2 - \sum_k p_k^2)(s^2 - \sum_k t_k^2)}}, \quad (\text{Eq. 60})$$

where,

$c = \sum_k^K C_{kk}$ is the total number of elements correctly predicted,

$s = \sum_i^K \sum_j^K C_{ij}$ is the total number of elements,

$p_k = \sum_i^K C_{ki}$ is the number of times that class k was predicted,

$t_k = \sum_i^K C_{ik}$ is the number of times that class k truly occurred.

Finally, the last metric considered is the Cohen's Kappa, which is similar to MCC when a multi-class classification task is being considered. Cohen's Kappa metric (K) can be described as follows:

$$K = \frac{c \times s - \sum_k^K p_k \times t_k}{s^2 - \sum_k^K p_k \times t_k} \quad (\text{Eq. 61})$$

The first image classifiers analysed are the ResNet50v2 and CNN networks. The architecture of the CNN has been set by applying heuristic evaluation and the consideration of architectures implemented in analogous studies (Alumtairi et al., 2021; Yao et al., 2021). Accordingly, the CNN architecture is comprised of two convolutional layers with 192 filters and kernel size of 3x3. The pooling operation presents a 2x2 dimension. Once the feature extraction phase is finalised, a total of three fully connected layers with 64, 128, and 192 hidden units are defined. To ensure

the adequate performance of the comparative study, the same parameters have been considered for the 1D-CNN architecture. Also, the time series imaging approach GAF is also analysed. Accordingly, two additional models are evaluated: GAF-ResNet50v2, and GAF-CNN.

The Table 7.4 presents the results after implementing the classification task ordered based on the resulting accuracy score. Markov-CNN outperforms the remaining models. For instance, Markov-CNN presents a performance enhancement of a 2% when considering the second most accurate model, and a 23% when the least accurate model is considered. Although a 2% of performance enhancement with regards to the second most accurate model, 1D-CNN, may not be significant, Markov-CNN is a turning point in the consideration of time series imaging approaches for performing fault classification tasks. Markov-ResNet50V2, present nearly identical results as 1D-CNN. After analysing the third most accurate model, a significant drop can be observed in the accuracy performance, leading to a decrease of the accuracy score of more than 10%. This suggests that the proposed time series imaging approach outperforms GAF.

Table 7.4. Classification metrics results for performance evaluation of the multi-fault classification task.

Model	Accuracy	Balanced Accuracy	Micro F1	Macro F1	MC C	Cohen's Kappa
Markov-CNN	0.95	0.95	0.95	0.94	0.94	0.94
1D-CNN	0.93	0.94	0.93	0.93	0.92	0.92
Markov-ResNet50V2	0.93	0.93	0.93	0.93	0.91	0.91
GAF-CNN	0.83	0.84	0.83	0.83	0.81	0.81
GAF-ResNet50V2	0.72	0.72	0.72	0.71	0.67	0.67

7.4. Case Study 7. Predictive Analytics Module¹⁴

This case study aims to validate the methodology presented in section 5.4. *Remaining Useful Life (RUL) Prediction Framework*. The turbocharger of a diesel GenSet of a tanker ship used for auxiliary needs is analysed. More than 66,000 instances collected in a 1-minute frequency basis are considered (please see Fig. 7.14). The descriptive statistics are presented in Table 7.5.

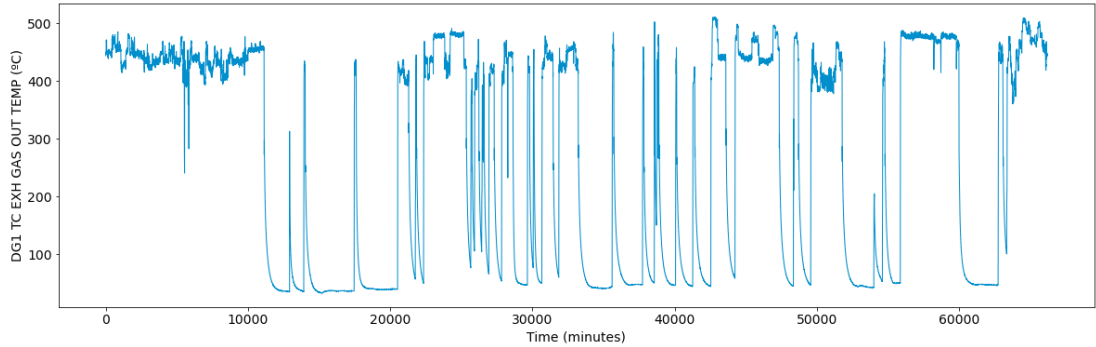


Fig. 7.14. Time series plot of the turbocharger exhaust gas outlet temperature monitored parameter.

¹⁴ The RUL prediction framework presented in this section has been already converted to a journal paper, and has been submitted in the *Engineering Applications of Artificial Intelligence* journal (Velasco-Gallego and Lazakis, 2022a).

Table 7.5. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
TC EXH GAS OUT TEMP (°C)	263.94	190.23	31.6	49.4	384.7	444.2	510.1

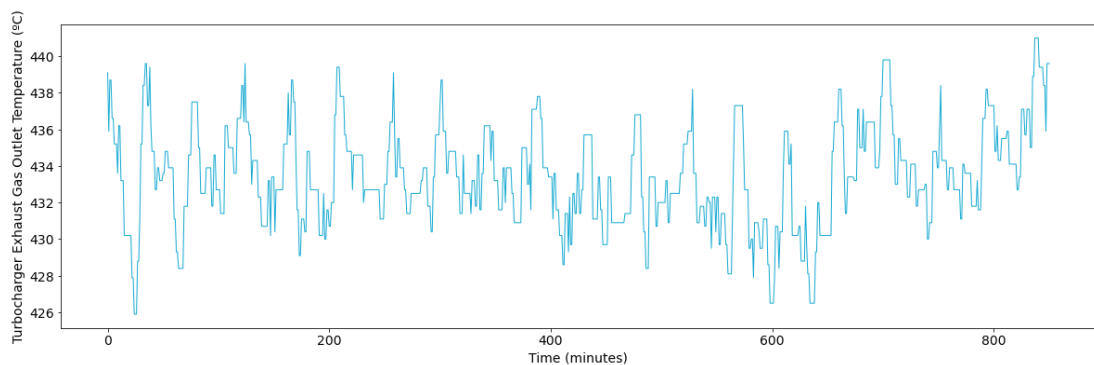
As data pre-processing, steady states' identification and data normalisation are performed. Also, the sliding window algorithm is implemented. After heuristic evaluation, the window size parameter is set to 60.

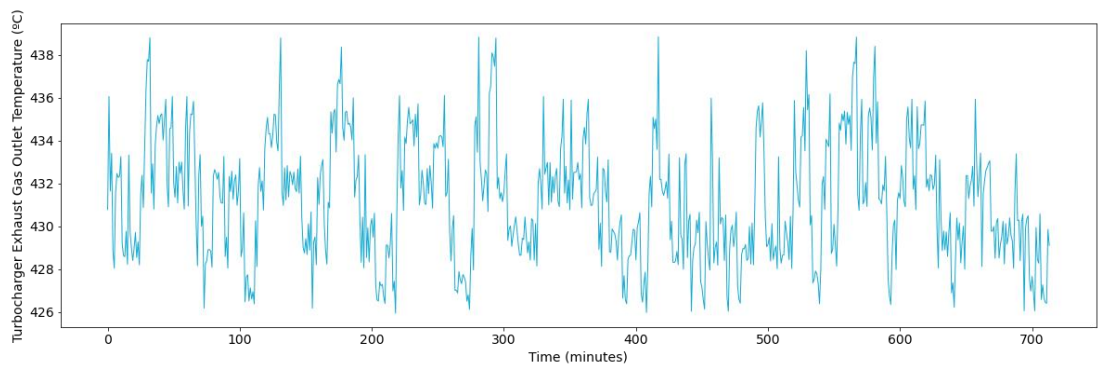
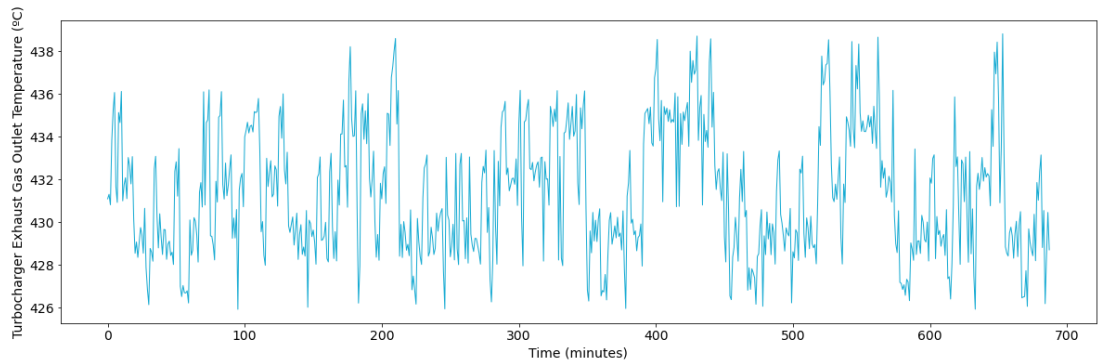
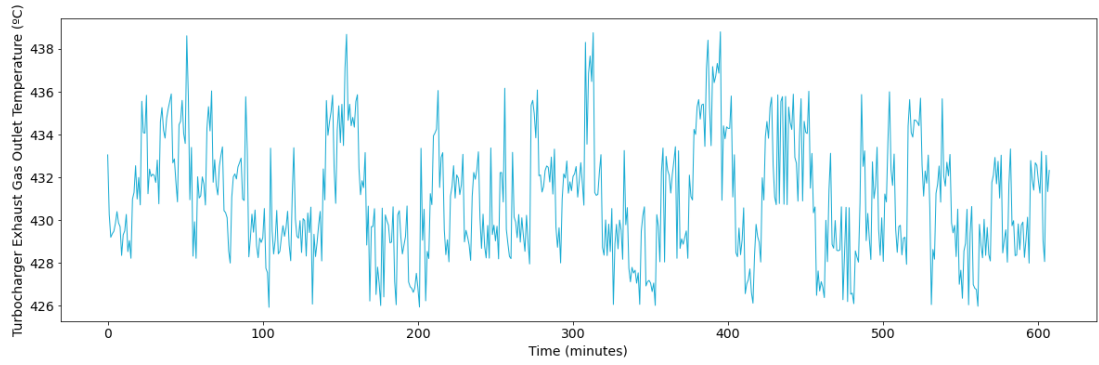
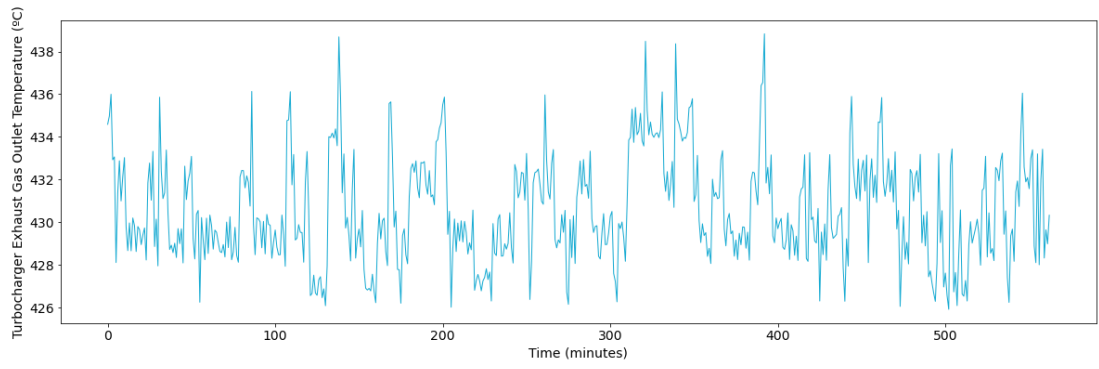
Subsequently, the degradation data simulation step is applied. Accordingly, synthetic data are generated through the implementation of the first-order Markov chain model. Examples of original and synthetic subsequences are presented in Fig. 7.15 and Table 7.6. It can be perceived that simulated data present similar characteristics to the original one. For instance, in all cases the mean value lies between 430.33 and 433.43 °C. Also, the standard deviation lies between 2.40 and 3.24. Therefore, this approach can be utilised to generate analogous operational scenarios to the original ones. Thus, more data can be utilised for training the models, and thus enhance their accuracy.

Due to the lack of degradation patterns, the phase that considers an exponential model with Brownian motion for degradation simulation is also applied. In total, 100 synthetic data are generated based on the observed operational sequences. For each

synthetic time series, a total of additional 100 degradations are simulated. Examples of these are graphically represented in Fig. 7.16. Fig. 7.16 (a) describes a simulated sequence without degradation pattern, as such a time series presents a steady trend. Fig. 7.16 (b-f) presents examples distinct generated degradation patterns.

Once the degradation patterns are obtained, both the condition indicators and the time series images can be estimated. After heuristic evaluation the resolution of the image is set to 25x25. With regards to the architectures, the Markov-CNN architecture is comprised of two convolutional layers, in which a total of 192 of size 3x3 has been considered. A max pooling layer has been introduced after the implementation of each convolutional layer. The pooling operation considered in this inquiry presents a 2x2 dimension. In relation to the regression stage, a total of three fully connected layers with 64, 128, 192 hidden units has been examined. The same configuration is set for the 1D-CNN.





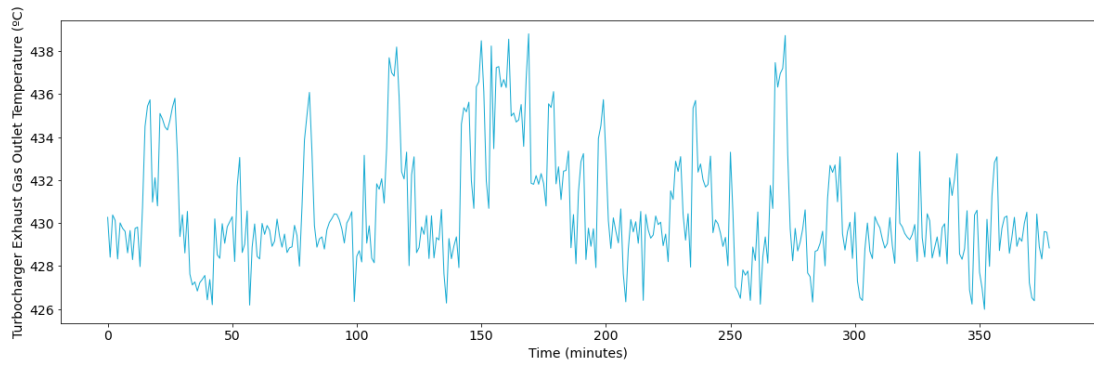
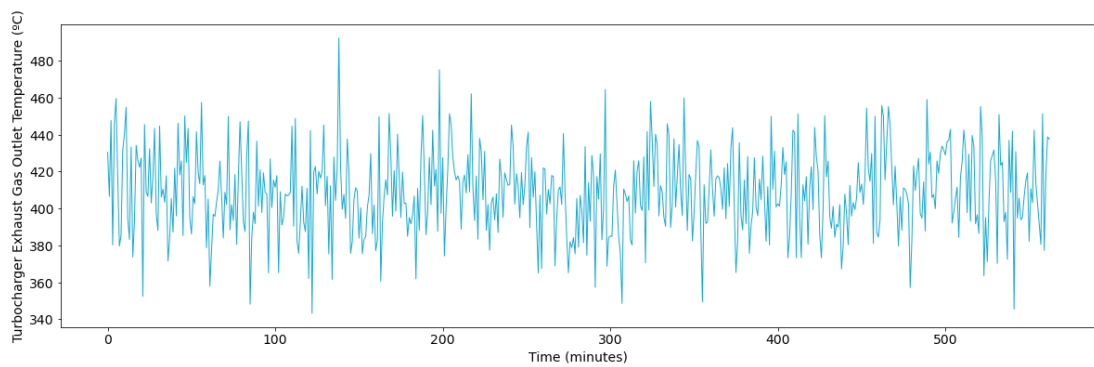
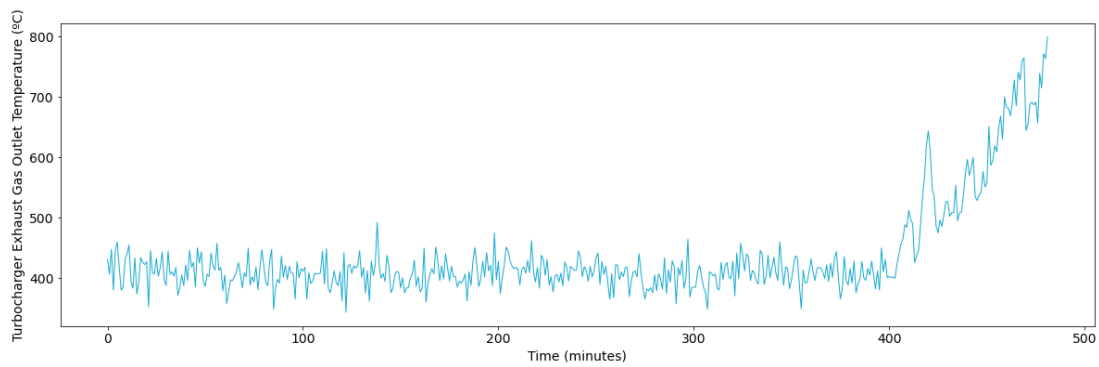
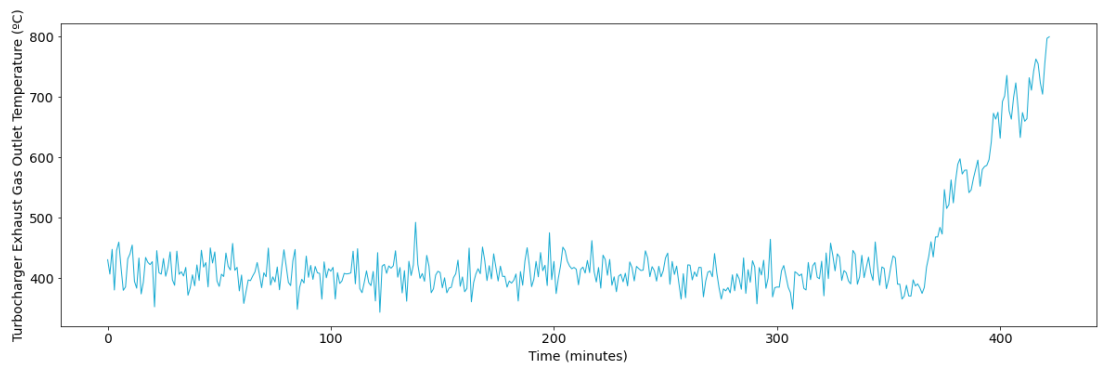
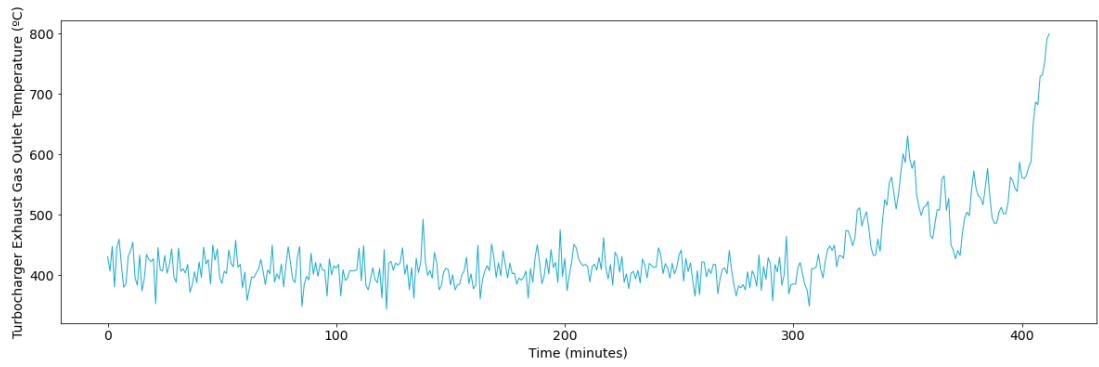
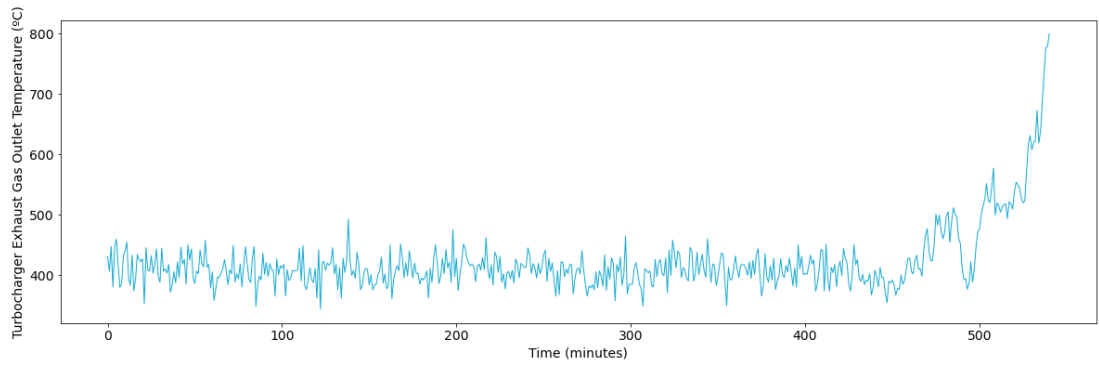


Fig. 7.15. (a) Example of an original subsequence plot (b-f) Example of a synthetic sequence plot.

Table 7.6. Descriptive statistics of an observed subsequence and its respective simulated subsequences.

	Count	Mean	Std.	Min.	0.25	0.5	0.75	Max.
Observed	851	433.43	2.76	425.90	431.40	433.40	435.20	441.00
Simulation 1	554	430.66	2.62	425.90	428.61	430.17	432.45	438.86
Simulation 2	709	430.59	2.59	425.94	428.79	430.06	432.24	438.83
Simulation 3	366	430.99	3.24	425.90	428.49	430.47	433.42	438.91
Simulation 4	321	430.33	2.40	425.93	428.56	430.05	431.99	438.90
Simulation 5	607	431.10	2.60	425.92	428.96	430.87	433.00	438.84





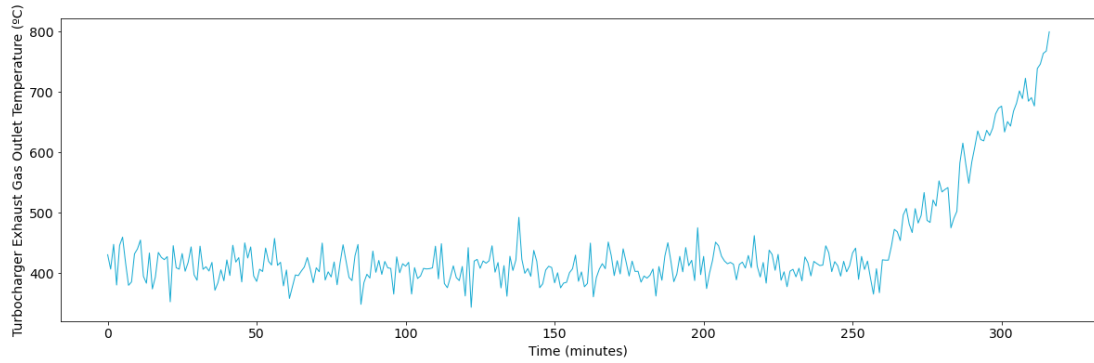


Fig. 7.16. (a) Example of a synthetic sequence (b-f) Example of a degradation simulation.

With regards to the LSTM, the architecture is defined based on heuristic evaluation and configuration analysis of analogous studies (e.g., Han et al. (2021)). Accordingly, the LSTM presents a total of 3 layers (64, 128, 192) and tanh activation function. In all three cases, the ratio of the validation set has been set to 0.20 and Adam optimiser has been applied to compile such models. For the training process, a total of 100 epochs and a batch size of 5 is considered.

After the training of all three models, the weights selection for the ensemble model is applied. Accordingly, the grid search optimisation algorithm is considered. Results of which determines that the weights selection is the following: $w_{Markov-CNN} = 0.3$, $w_{1D-CNN} = 0.5$, and $w_{LSTM} = 0.2$. Thus, it can be observed that the performance of the models can be sorted as follows: LSTM < Markov-CNN < 1D-CNN, thus being 1D-CNN. Nevertheless, it can be perceived that for certain scenarios either LSTM or Markov-CNN outperforms 1D-CNN (please see Fig. 7.17).

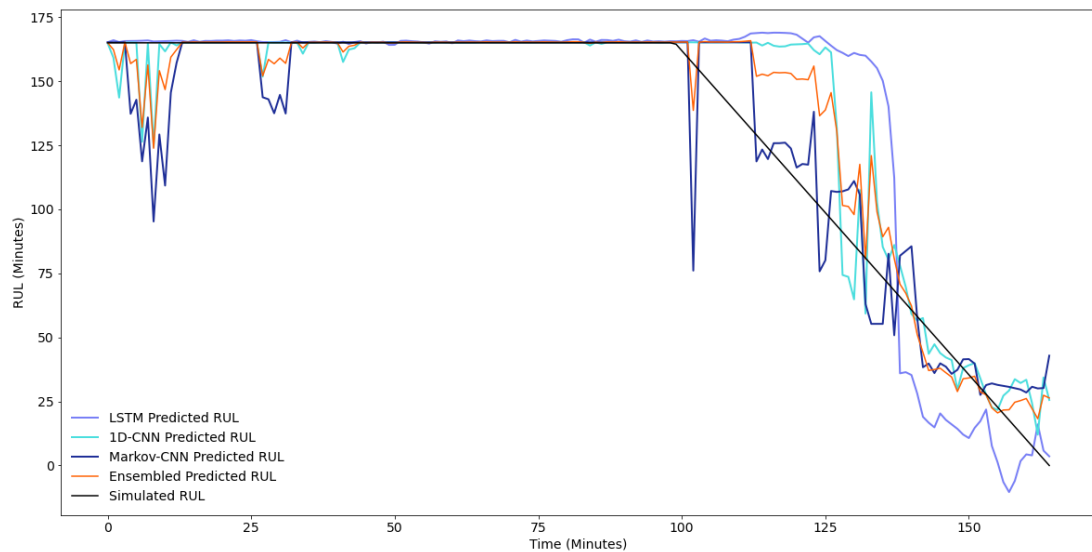


Fig. 7.17. RUL prediction for the turbocharger exhaust gas outlet temperature.

In Fig. 7.17 it can be perceived that Markov-CNN yields the most accurate predictions. Although all three models can predict adequately the RUL in operational conditions. However, it can be perceived that the estimated RUL in the degradation stage differs significantly. For instance, the LSTM model cannot identify the degradation occurrence in its first stage, although a higher precision is obtained when the failure is about to occur. A similar pattern is observed if the 1D-CNN is considered. With regards to Markov-CNN, this model presents more consistency than the preceding two models in both the early and medium stages. Nevertheless, the accuracy is reduced if the last stage of the degradation is considered. Furthermore, a downward spike at the beginning of the degradation stage is observed, thus indicating that the Markov-CNN may be unable to adequately capture the transition between the normal and

degradation phases. Accordingly, by considering an ensemble model, the RUL prediction is enhanced, and the resulting approach is more robust, as the limitation of each individual model can be minimised.

Analogous results have been determined in the remaining predictions (please see Table 7.7). Table 7.7 presents a total of 50 distinct RUL predictions performed during the test stage. Both the RMSE and the Maintenance Score are estimated to assess the performance of the models, these metrics being the most appropriate ones when evaluating the implementation of a data-driven model that aims to perform the RUL prediction task (Han et al., 2021). Overall, the integration of the three analysed models into an ensemble approach provides more robust results, as the utilisation of ensembles enhances the RUL prediction over any other analysed model in this study by reducing the variance of the prediction errors. The contrary can be observed when only the Markov-CNN architecture is considered, as, although this model outperformed the other models in some of the RUL predictions, their low level of accuracy in other predictions (e.g., predictions of RUL 22 and 44) cannot be diminished. Therefore, while time series imaging presents promising results, more efforts need to be made for considering this type of approach as a potential model for the prediction of RUL.

Table 7.7. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.

Simulated RUL	RMSE (Minutes)				Maintenance Score			
	Markov-CNN	LSTM	1D-CNN	Ensembled	Markov-CNN	LSTM	1D-CNN	Score Ensembled
RUL 1	16.15	27.23	19.12	15.54	1.26E+03	2.39E+04	4.55E+03	1.18E+03
RUL 2	69.27	59.40	59.28	46.08	8.18E+12	2.58E+09	3.16E+13	2.36E+08
RUL 3	55.43	60.12	36.17	36.12	1.42E+10	1.94E+15	1.87E+10	2.43E+09
RUL 4	36.23	28.49	18.74	18.76	1.05E+06	2.17E+05	5.23E+03	2.64E+03
RUL 5	18.29	32.61	26.42	18.91	2.10E+03	1.88E+04	3.51E+04	9.60E+02
RUL 6	99.20	80.18	92.55	72.88	6.62E+30	5.27E+10	3.38E+29	9.25E+17
RUL 7	35.08	57.38	66.50	45.93	2.94E+06	2.43E+08	7.63E+10	1.06E+07
RUL 8	46.45	17.02	23.88	23.81	2.34E+09	2.06E+04	5.34E+05	5.44E+05
RUL 9	48.35	34.52	28.41	20.18	4.30E+06	7.56E+06	6.67E+05	7.96E+03
RUL 10	28.01	23.63	15.65	15.24	5.30E+04	4.56E+04	7.17E+02	7.78E+02
RUL 11	72.80	11.39	10.81	22.02	1.04E+14	7.46E+03	3.17E+07	3.37E+05
RUL 12	79.34	69.21	74.35	56.76	4.92E+08	4.49E+10	7.29E+10	1.76E+07
RUL 13	65.94	63.20	62.14	53.30	2.44E+10	8.09E+07	2.84E+11	3.08E+08
RUL 14	58.97	90.67	78.13	70.43	7.49E+05	1.61E+07	1.23E+07	2.64E+06
RUL 15	43.99	26.69	19.16	22.61	1.36E+10	6.38E+09	2.90E+06	1.55E+07
RUL 16	120.62	65.07	59.10	52.73	2.40E+21	1.97E+08	5.06E+10	4.10E+09
RUL 17	84.05	62.16	41.07	41.17	6.45E+20	1.17E+22	8.77E+14	7.50E+14
RUL 18	31.34	66.15	68.44	49.07	1.12E+05	4.90E+07	7.85E+07	3.53E+05
RUL 19	76.81	44.16	38.65	38.48	4.52E+17	1.05E+06	9.32E+06	5.21E+07
RUL 20	58.54	28.88	18.39	24.10	1.40E+09	2.81E+07	2.68E+05	1.38E+05
RUL 21	73.15	48.94	27.07	32.04	6.99E+13	2.19E+10	2.76E+06	7.31E+07
RUL 22	131.04	89.02	71.86	67.57	7.30E+36	3.13E+11	9.49E+15	4.18E+14
RUL 23	64.87	51.57	30.24	31.15	1.24E+13	3.92E+15	3.20E+08	6.88E+06
RUL 24	71.71	26.50	44.64	41.53	3.14E+07	1.13E+06	9.83E+07	8.16E+06
RUL 25	156.30	79.65	76.31	83.85	1.79E+20	6.25E+15	1.49E+14	3.93E+14
RUL 26	109.60	37.79	63.53	54.65	9.89E+24	4.18E+07	9.90E+24	2.11E+15

RUL 27	102.35	35.81	43.31	43.94	3.13E+13	3.38E+04	3.14E+09	8.13E+06
RUL 28	94.95	64.74	59.50	53.68	5.77E+27	3.64E+10	1.05E+13	7.74E+08
RUL 29	75.36	78.44	60.37	52.96	1.82E+16	8.76E+18	2.54E+13	9.94E+11
RUL 30	39.49	39.32	36.12	26.85	1.23E+05	2.23E+05	5.43E+05	4.03E+03
RUL 31	129.75	89.98	84.93	89.19	6.65E+16	1.99E+22	2.00E+14	3.58E+13
RUL 32	81.39	51.95	54.31	50.01	5.20E+12	9.04E+08	4.74E+10	2.20E+08
RUL 33	32.88	15.86	13.83	15.56	1.03E+05	9.79E+04	6.19E+03	6.27E+03
RUL 34	78.56	96.20	63.54	52.26	9.90E+16	2.79E+17	4.97E+14	3.78E+09
RUL 35	88.13	89.21	78.84	67.30	6.85E+18	6.99E+16	9.59E+16	3.24E+15
RUL 36	39.51	15.00	18.70	18.12	2.97E+07	5.47E+02	7.46E+03	2.44E+03
RUL 37	109.22	14.38	36.43	46.00	1.55E+17	2.62E+06	1.56E+10	3.22E+10
RUL 38	119.33	67.27	63.29	55.37	2.55E+22	4.70E+07	5.12E+09	6.89E+07
RUL 39	52.86	46.59	49.54	42.04	4.82E+10	5.53E+10	4.42E+10	3.45E+10
RUL 40	20.84	40.09	31.35	21.17	4.05E+03	1.29E+06	1.03E+06	3.46E+03
RUL 41	85.49	47.19	44.02	45.50	5.78E+13	1.15E+11	1.91E+09	1.22E+08
RUL 42	71.49	52.62	42.91	34.25	3.01E+08	1.01E+11	5.33E+09	2.49E+06
RUL 43	45.26	66.44	69.69	47.70	2.82E+12	3.72E+09	1.03E+10	2.57E+06
RUL 44	165.47	59.22	44.69	57.56	2.22E+32	5.03E+11	2.51E+08	3.31E+10
RUL 45	96.15	18.41	16.59	34.81	7.64E+12	2.54E+04	6.50E+04	1.92E+06
RUL 46	55.77	42.21	47.14	39.38	3.21E+09	1.01E+07	3.17E+09	1.83E+06
RUL 47	147.58	41.53	34.44	57.97	7.96E+19	1.11E+16	6.42E+12	1.31E+14
RUL 48	35.09	73.51	75.83	51.03	1.30E+07	1.08E+10	2.97E+10	1.57E+06
RUL 49	69.37	70.49	61.31	48.43	1.82E+13	2.45E+18	2.39E+13	4.38E+08
RUL 50	69.83	61.07	52.88	46.47	3.61E+12	2.61E+11	1.83E+11	6.30E+08
Median	70.66	51.76	44.67	45.71	4.41E+12	5.74E+08	5.23E+09	3.48E+07

7.5. Case 8. MA framework

To finalise the validation process, a case study is presented in this section in order to analyse the overall performance of the proposed MA framework. Accordingly, the main engine of a bulk carrier is considered. Specifically, the exhaust gas outlet temperature parameter of one of the cylinders is introduced in this study due to its criticality in adequately monitoring the functioning of the cylinder. The analysis of other parameters can be consulted in *Appendix D. Main Results of Case Study 8. MA framework*.

More than 2,000 instances are analysed for such a parameter. Such instances have been collected in a 10-minute frequency. A graphical representation is presented in Fig. 7.18. The descriptive statistics are also presented in Table 7.8.

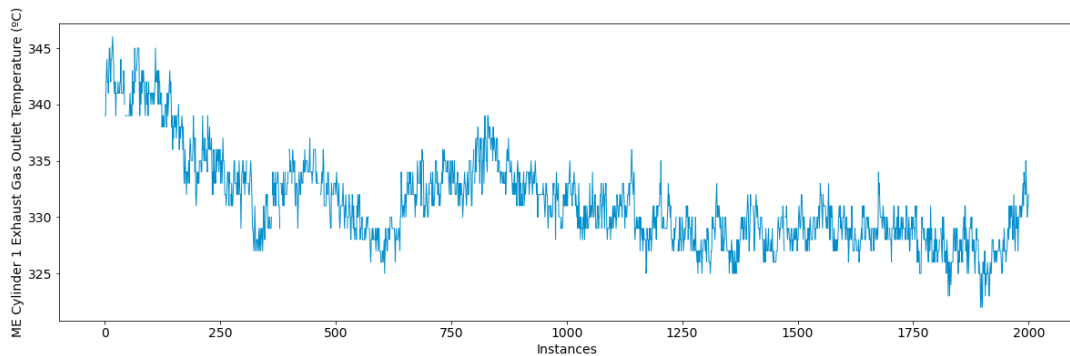


Fig. 7.18. Graphical representation of the cylinder 1 exhaust gas outlet temperature parameter.

Table 7.8. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
Cyl. 1 Exh. Gas Out. Temp	331.31	4.06	322	328	331	333	346

As part of the data pre-processing phase, the identification of operational states step has been implemented. In total, only one operational state has been identified, as perceived in Fig. 7.19. Unlike the case study presented in section 6.5. *Case Study 4. Novel Framework for the Identification of Steady States*, in which distinct operational profiles could be perceived, the case presented in this section only presents on operational profile. Thus, it is considered that that the results of the algorithm implemented for the identification of operational states are satisfactory, and therefore the number of operational profiled considered in this case study is 1.

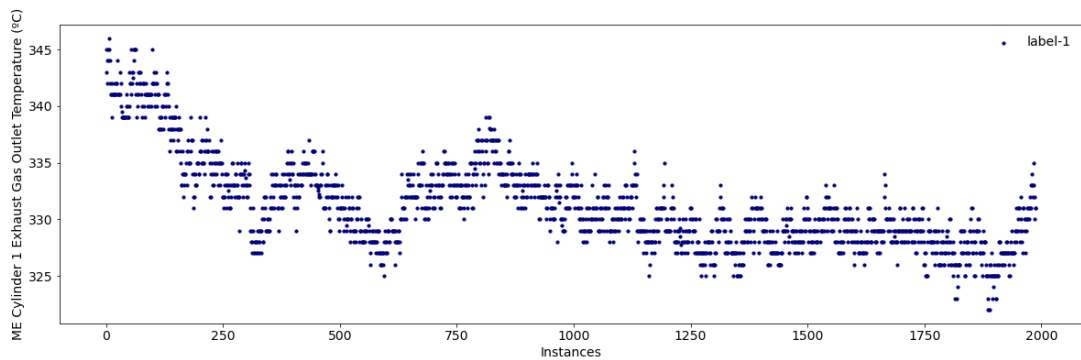
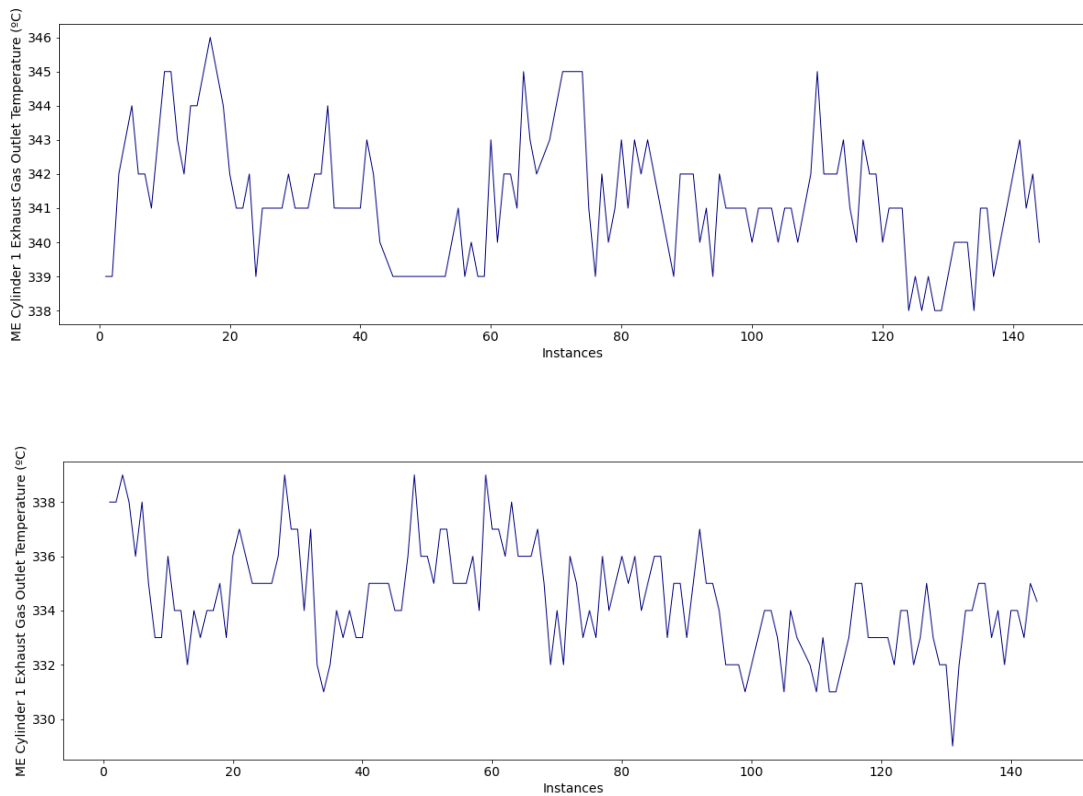


Fig. 7.19. Identification of the operational states for the monitored parameter.

Due to the characteristics of this dataset, the methodology presented for the performance of data imputation has not been required. However, due to the lack of fault data, both collective anomalies and degradation patterns are simulated for validation purposes and as part of the data augmentation step. Some examples are presented in Figs. 7.21 – 7.22. Examples of normal sequences are also introduced in Fig. 7.20.



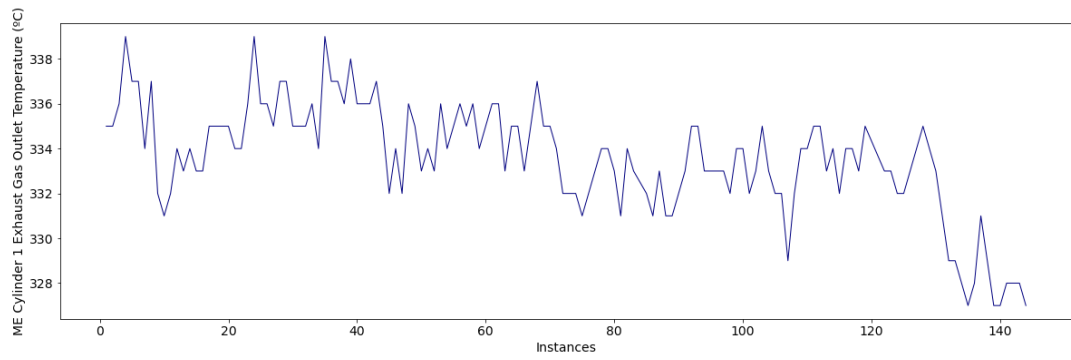
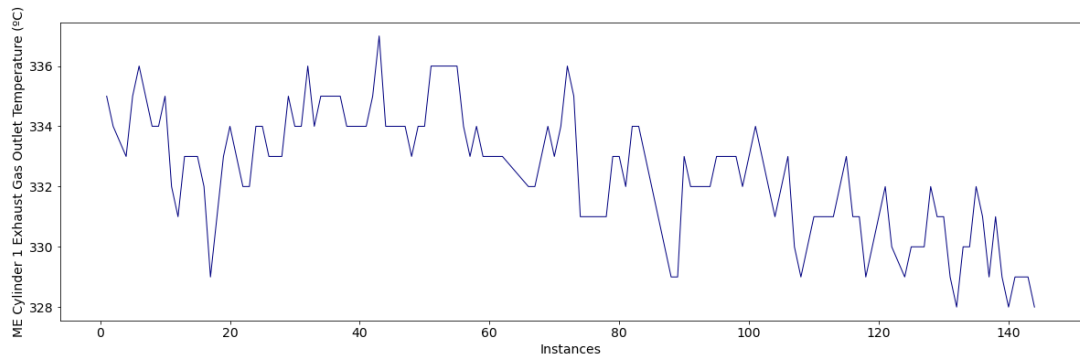
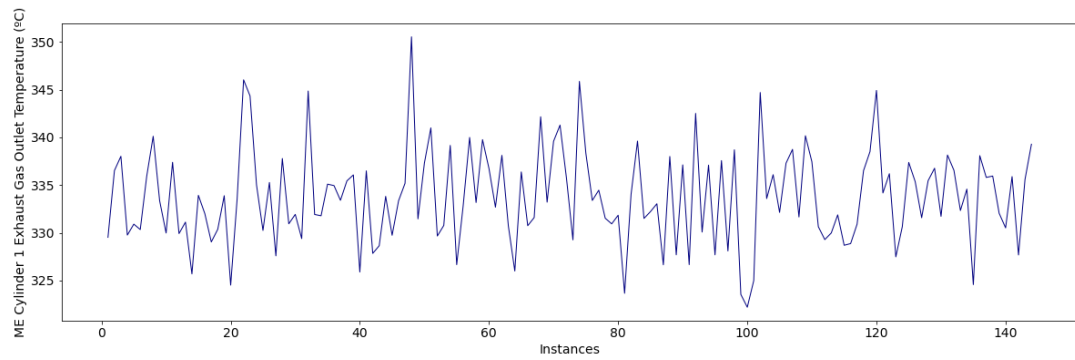


Fig. 7.20. Example of normal sequences.



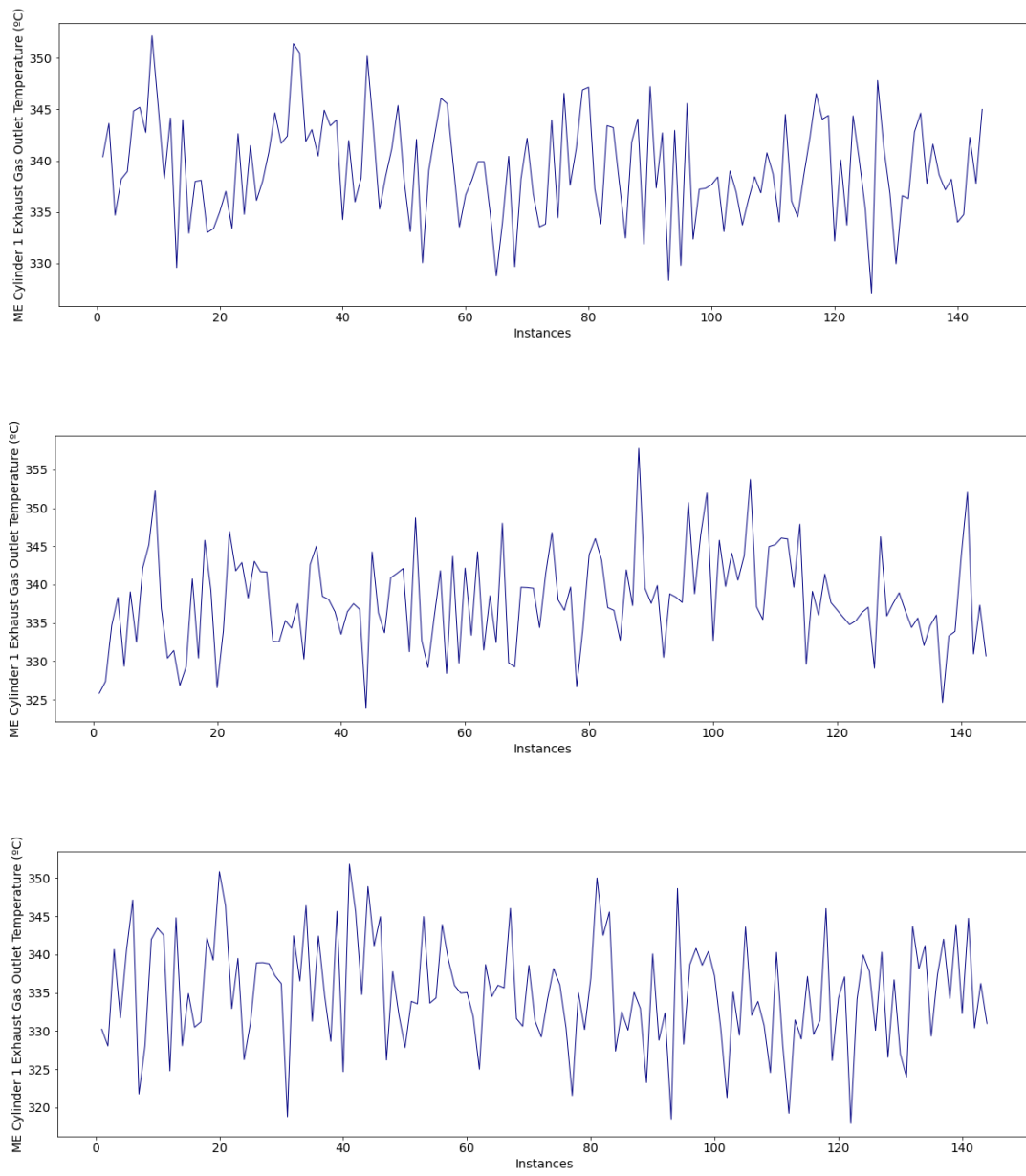


Fig. 7.21. Example of collective anomalies.

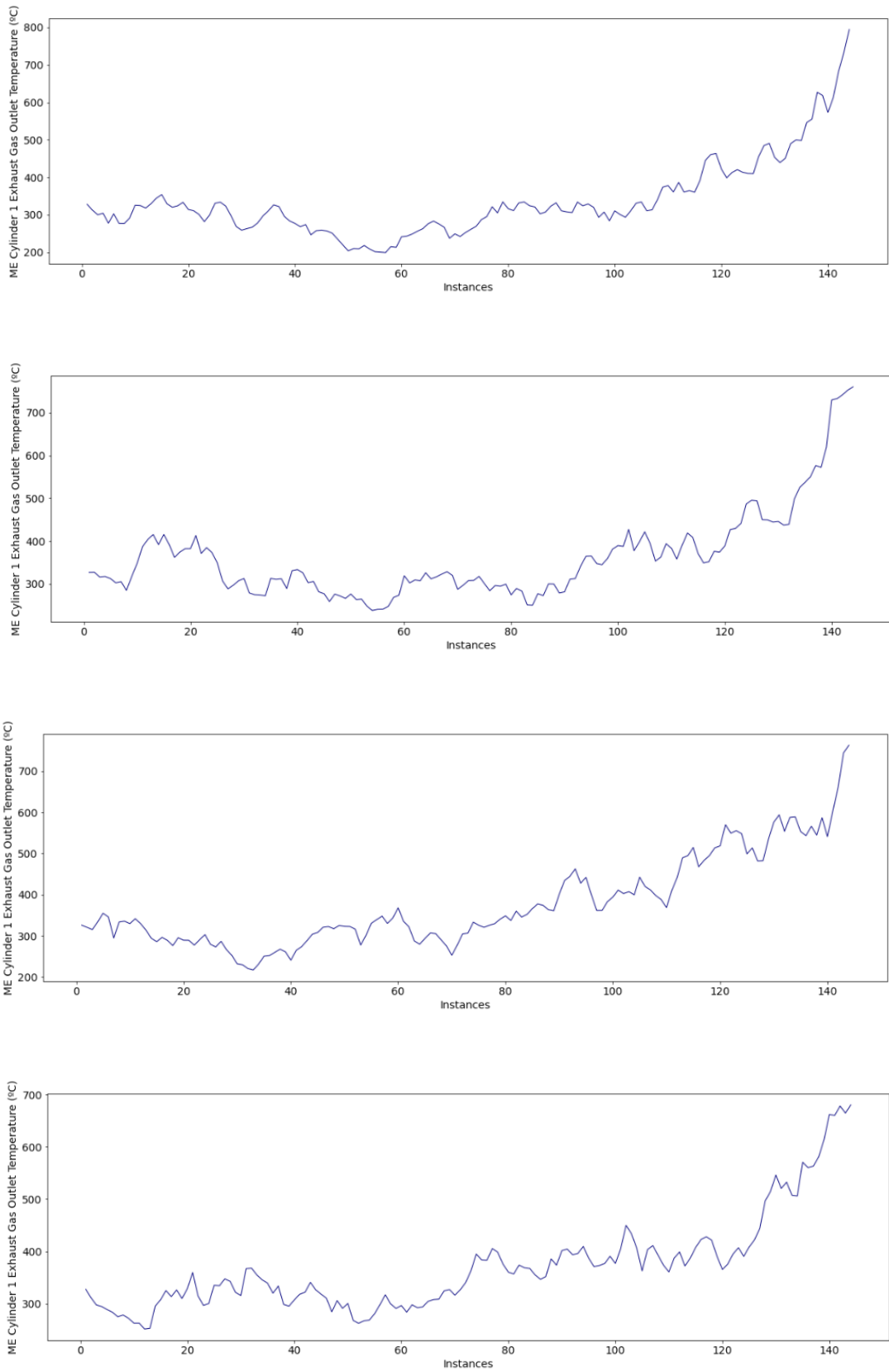


Fig. 7.22. Example of degradation patterns.

As perceived in the preceding figures, the sequences that contain collective anomalies and degradation patterns can be easily distinguished from the normal sequences. When the system is running under a normal steady state, the variability between instances is minimal. However, when the system starts to present an abnormal behaviour, such variability starts to be significant, and thus a collective anomalous context can be perceived. Lastly, when the systems are running under a failure state, an exponential growth in the sensor readings of the monitored parameter is presented until the system fails.

In order to be able to detect the preceding scenarios and avoid the system to present a failure state, and therefore avert the respective consequences of achieving such a state, the diagnosis module is implemented. The first part of this module consists of applying the fault detection phase, and thus being able to detect any anomalous sequence (please refer to section *5.3.1. Fault Detection* for a comprehensive explanation of such a step). For hyperparameter tuning, please refer to section *7.2. Case Study 5. Fault Detection as Part of the Diagnostic Analytics Module*. As a result, a LSTM-based VAE architecture with two layers in both encoder and decoder with units (128, 64) and (64, 128) is considered. The latent dimensions are set to 10 and the sequence length to 18, thus each input sequence presents information of the last 3 hours. The results of this phase are graphically represented in Figs. 7.23-7.28.

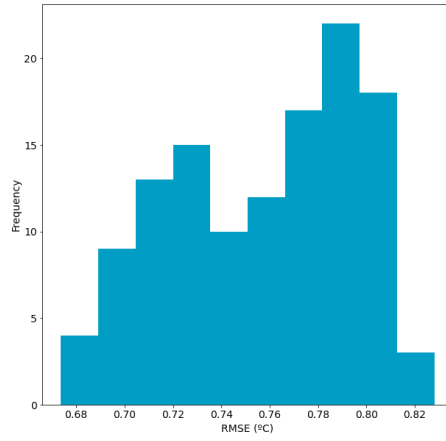


Fig. 7.23. Histogram of the reconstructed errors of the normal sequences (test set).

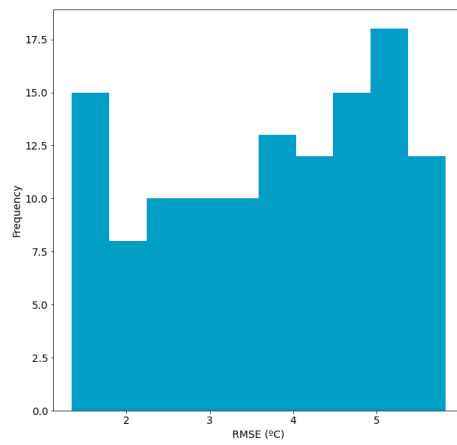


Fig. 7.24. Histogram of the reconstructed errors of the sequences with collective anomalies.

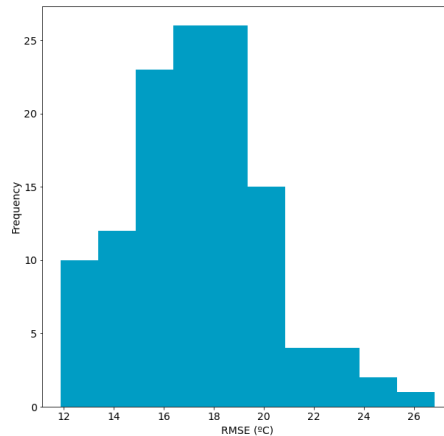


Fig. 7.25. Histogram of the reconstructed errors of the sequences with degradation patterns.

Figs 7.23-7.25 refer to the histograms of the reconstruction error of the analysed sequences. As it can be observed, the reconstruction errors of the sequences that contain collective anomalies and degradation patterns are much greater than the normal sequences. Accordingly, in this case the anomalous sequences can be easily detected by applying a regular threshold, thus being able to detect all anomalous sequences adequately. Figs 7.26-7.28 present examples of reconstructed sequences in which it can be similarly observed the reconstruction trend for each type of analysed sequence.

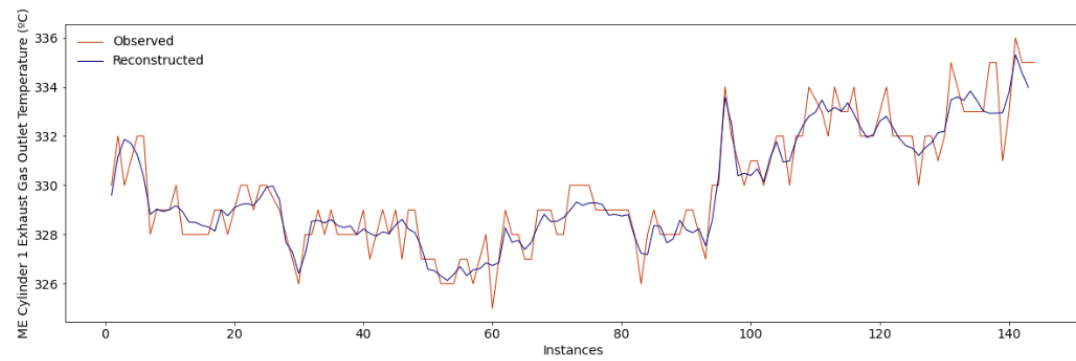
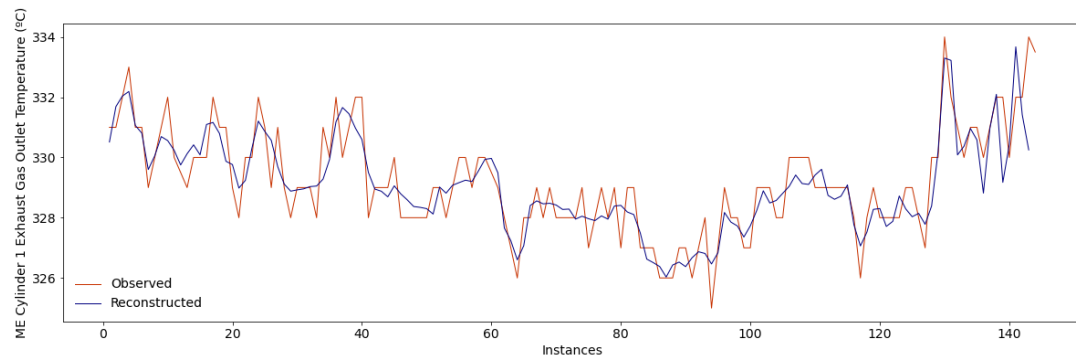
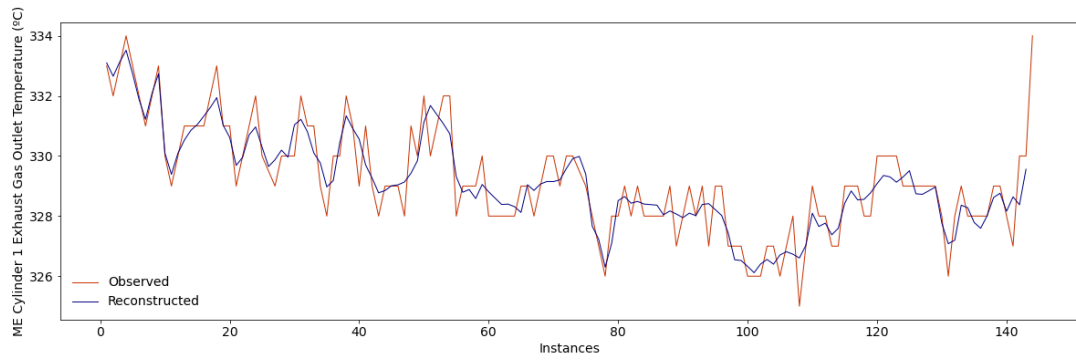
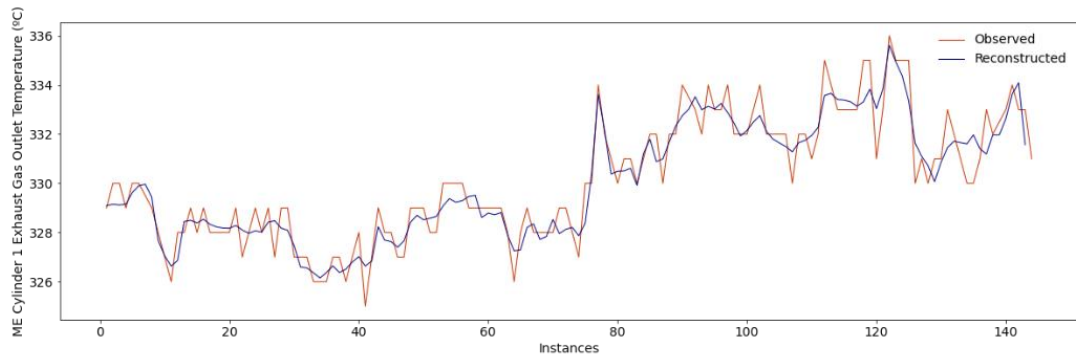


Fig. 7.26. Example of normal reconstructed sequences (test set).

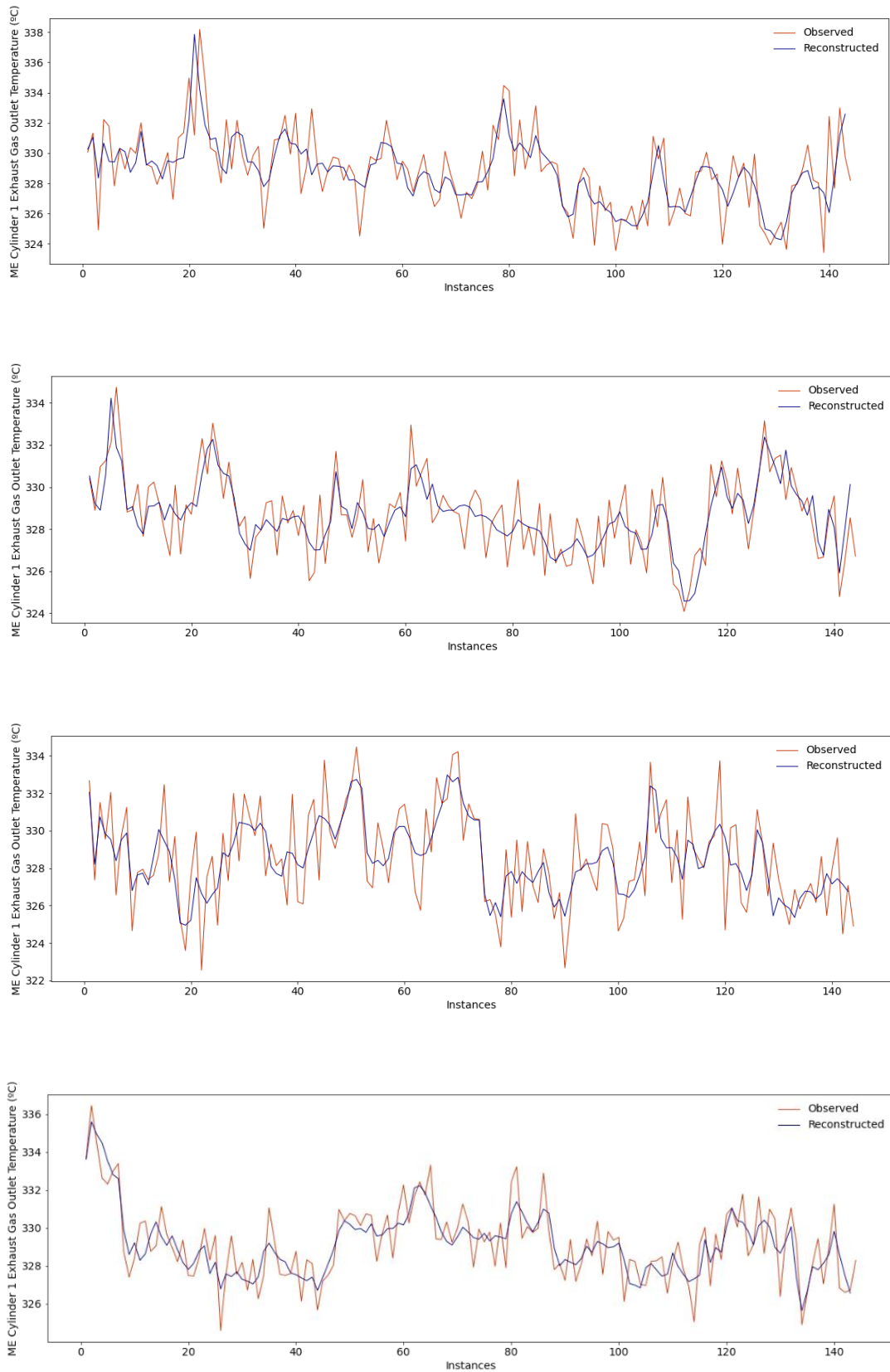


Fig. 7.27. Example of reconstructed sequences with collective anomalies.

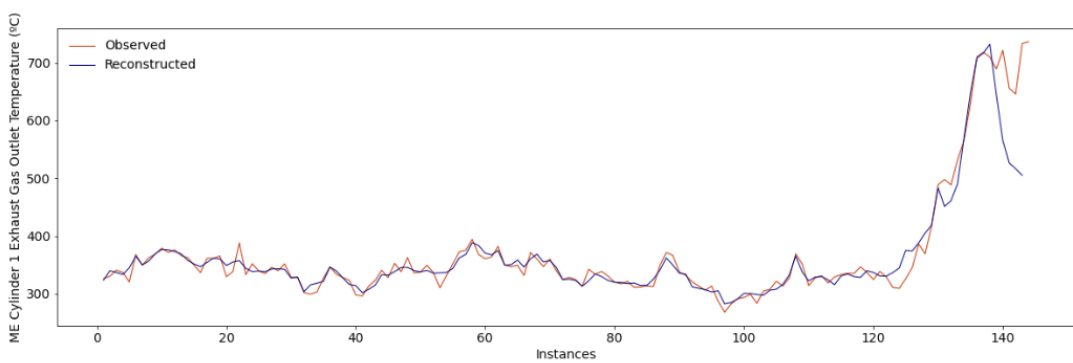
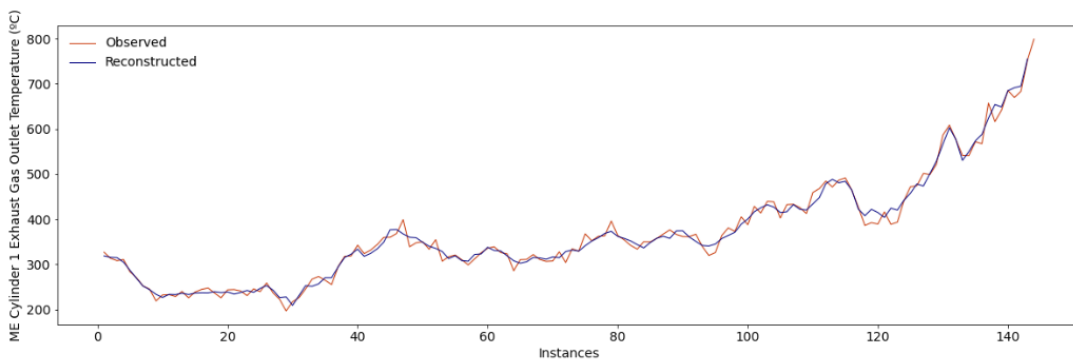
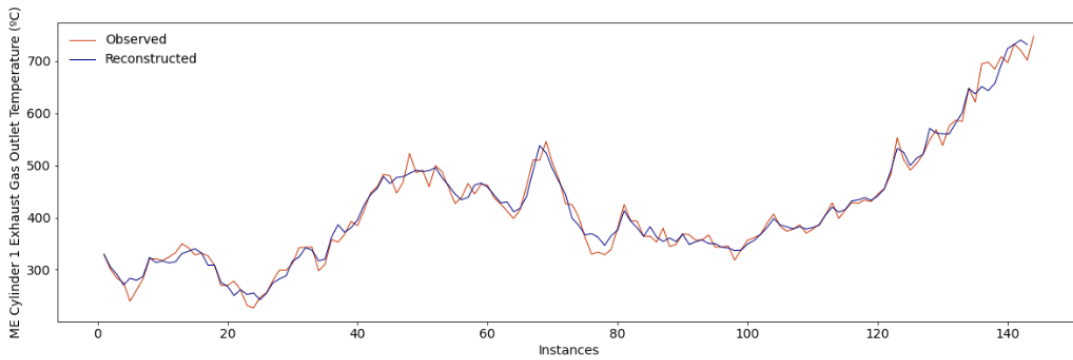
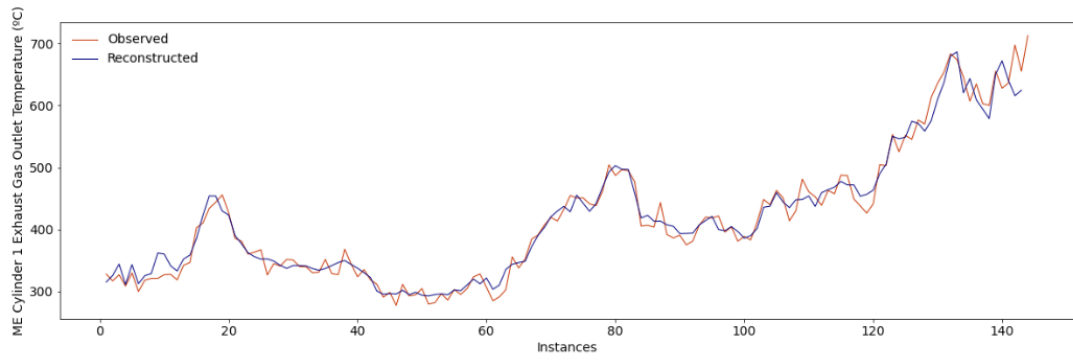


Fig. 7.28. Example of reconstructed sequences with degradation patterns.

Those sequences that have been detected as anomalous in the fault detection phase are presented as input in the fault identification phase to identify the characteristics of such abnormal patterns. In this case, the fault identification phase can be defined as a binary classification, as only two distinct abnormal patterns are being considered (collective anomalies and degradation patterns). As concluded in section 7.3. *Case Study 6. Fault Identification as Part of the Diagnostic Analytics Module*, the architecture Markov-CNN was the one that presented the most promising results. Accordingly, such an architecture is once again implemented for this case study. Thus, the first step to be implemented is related to the encoding of time series sequences into images. To proceed with such an encoding the first-order Markov chain is introduced, as comprehensively described in section 5.3.2. *Fault Identification*. Examples of encoded images are presented in Figs. 7.30-7.31. An example of time series imaging of a normal sequence is also presented in Fig. 7.29 for comparative purposes.

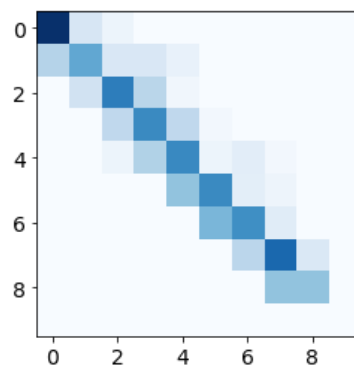


Fig. 7.29. Image of a normal sequence.

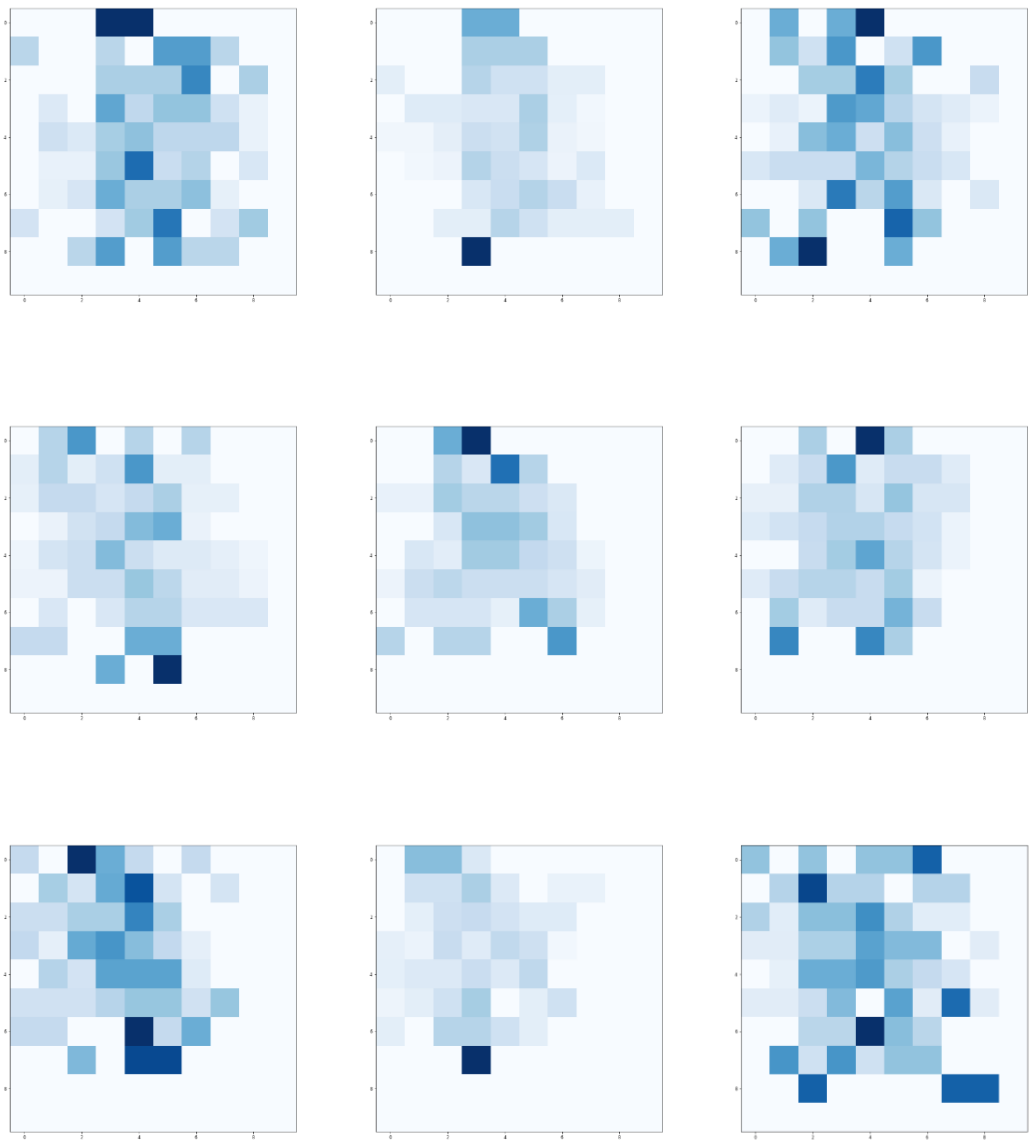


Fig. 7.30. Images with collective anomalies.

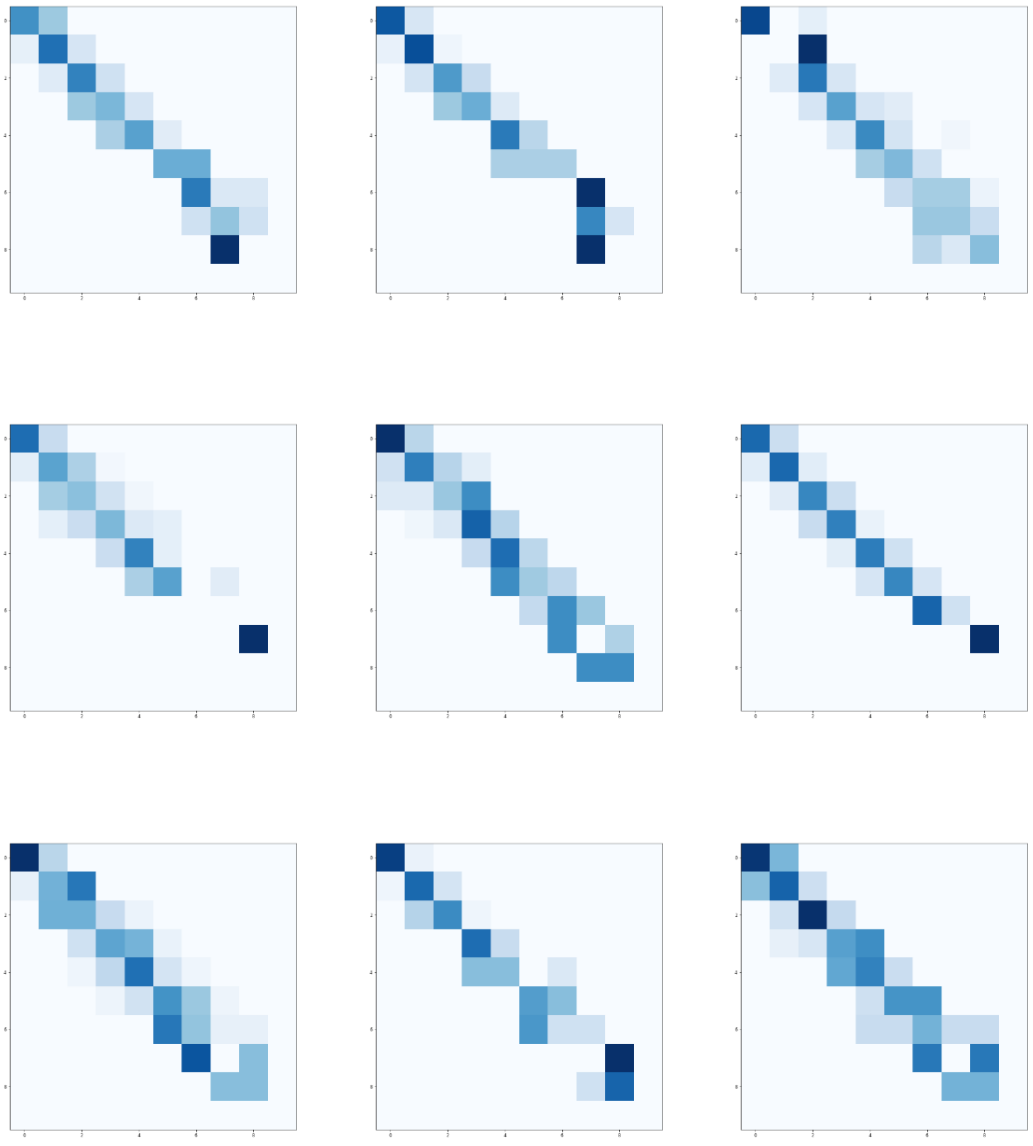


Fig. 7.31. Images with degradation patterns.

As observed if Fig. 7.29, the transition is usually performed between neighbour states when normal sequences are considered. Thus, if the current instance relates to state 2, it is highly probable that the subsequent instance will lie between states 1 and 3, thus being unlikely to be associated with state 8, for instance. For this reason, a diagonal of high intensity pixels is usually defined in images that represent normal

sequences. By contrast, there is no trend identified when images represent collective anomalies, as it can be perceived in Fig. 7.30. Distinct pixels with different intensity values are presented randomly along the dimensions of the image. However, when images with degradation patterns are analysed (see Fig. 7.31), an analogous trend as the one presented in Fig. 7.29 can be observed. Such a fact is once related with the transition between neighbour states. When the current instance refers to a high degradation state (e.g., state 8) it is highly unlikely that the subsequent instance will refer to an initial indication of degradation (e.g., state 1). Thus, the subsequent instance will lie between states 7-9. That said, although an analogous trend can be observed between the image with operational instance and the images with degradation patterns, several differences can be perceived. For instance, the diagonal of high intensity pixels is not as well defined with degradation patterns are considered due to the high variability presented. Furthermore, the highest level of intensity can be perceived in the last pixels of the diagonal of high intensity pixels.

By considering such differences, a CNN architecture is proposed to extract the deep features that can be utilised to determine the nature of the anomalies analysed. The hyperparameter tuning has been implemented as stated in section 7.3. *Case Study 6. Fault Identification as Part of the Diagnostic Analytics Module*. Accordingly, the considered CNN architecture is comprised of two convolutional layers with 192 filters and kernel size of 3x3. The pooling operation presents a 2x2 dimension. After the

feature extraction stage, a total of three fully connected layers with 64, 128, and 192 hidden units are defined. The dimensions of the images are set to 10x10.

As stated in the preceding paragraph, the two categories analysed in this case study can be easily distinguished. This aspect facilitated the achievement of the maximum accuracy score in this process, thus classifying all the images adequately. However, it has been perceived in section 7.3. *Case Study 6. Fault Identification as Part of the Diagnostic Analytics Module* that such an accuracy can decrease significantly when either considering a multi-class classification task or the classes cannot be distinguished as smoothly. Furthermore, further tests need to be performed with real-world scenarios to assess the full potential of such an approach.

To finalise this case study, the last module of the MA framework, 5.4.1. *Remaining Useful Life (RUL) Prediction Framework*, is analysed. After the implementation of the fault identification phase, those sequences that refer to degradation patterns that can indicate the evolution of the failure state are considered for the estimation of the RUL of the system. Fig. 7.32 presents some examples of RUL predictions.

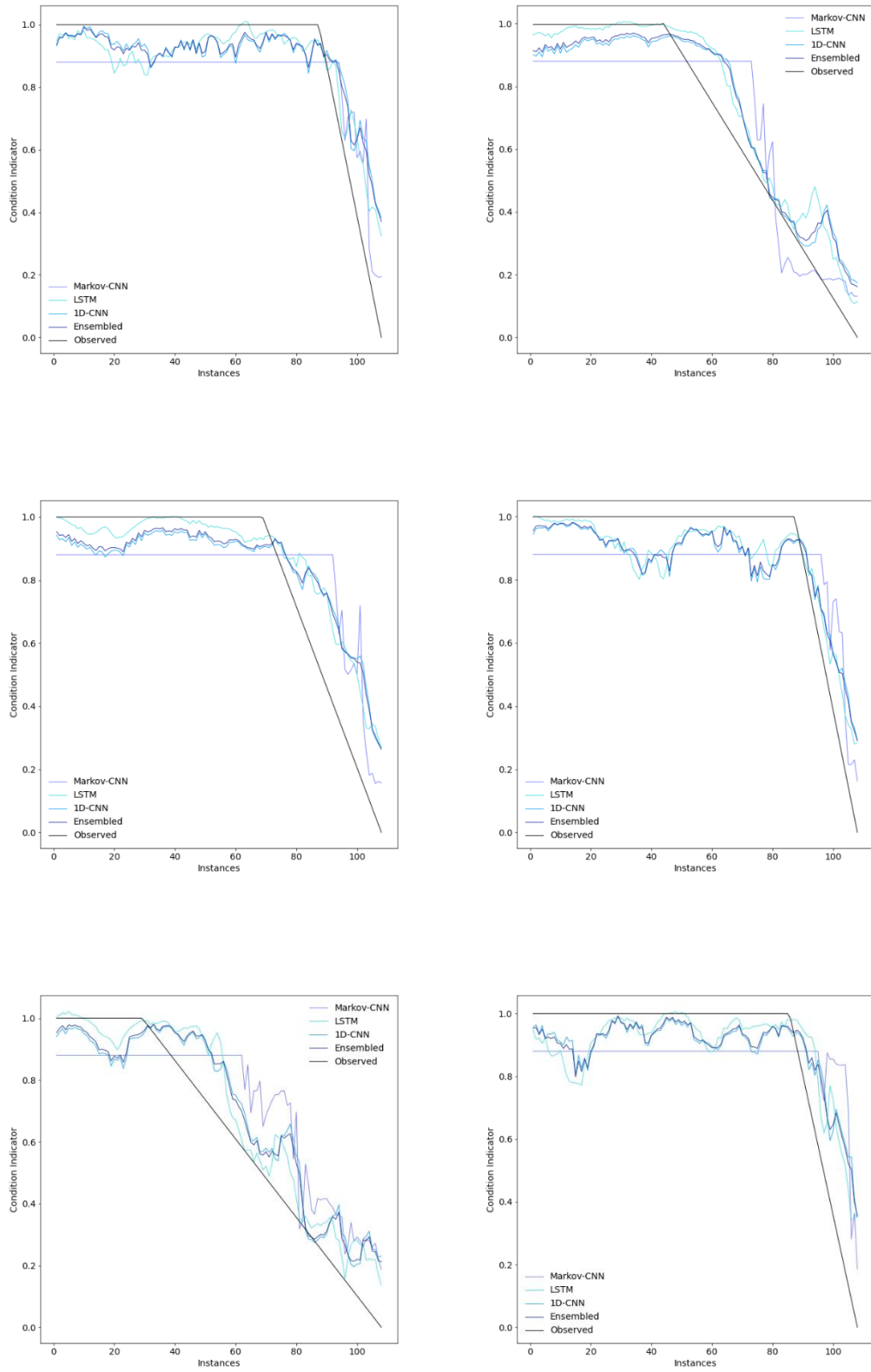


Fig. 7.32. Examples of condition indicator prediction.

As it can be perceived in Fig. 4.70 and Table B.2.2, there is no clear analysed model that outperforms the remaining ones. Accordingly, the consideration of an ensemble approach is usually the one recommended, as it is more robust and present more accuracy in the overall results than any individual model. However, such a fact is not usually certain in certain scenarios, as it can be perceived in *Appendix D. Main Results of Case Study 8. MA framework*. Moreover, it can also be perceived that the results obtained are less precise than the ones obtained in the preceding tasks. Such a fact can be related to the RUL uncertainty and/or the simulated data utilised for training purposes. Challenges that will be considered as part of future work and that will be further described in the subsequent chapter of this thesis

7.6. Chapter Summary

This chapter has introduced a summary of four case studies performed to validate each of the methodologies presented in the diagnostic and predictive analytics module. Results obtained have demonstrated that the application of data-driven methodologies for the implementation of both diagnostic and predictive analytics can effectively determine the current and future health of marine systems, thus facilitating the implementation of such methodologies to assist with decision-making strategies with regards to O&M activities. By ensuring the robustness and flexibility of O&M activities, the reliability and availability of marine systems can be guaranteed as well

as the reduction of the downtime and operational cost whilst enhancing ship/company profitability.

Chapter 8

Discussion and Conclusions

8.1. Chapter Overview

This is the last chapter of the present thesis, which provides a discussion of the results outlined and the validation of the novelties established. Accordingly, the fulfilment of the aim and objectives, the generated novelty, the reflection, the recommendation for future work, and the assumptions and challenges are comprehensively presented throughout this chapter.

8.2. Generated Novelties

A total of eight novelties have been generated as a result of the development of a novel MA framework. The architecture of such a MA framework has been defined according to both the critical literature review performed and the gaps identified. The development of an overall MA framework within the shipping industry was presented as the main generated novelty, as there is no evidence in the academic literature that a holistic framework constituted by diagnostic analytics (fault detection and identification) and predictive analytics (remaining useful life prediction) has been formalised, even though it has been demonstrated that the use of such frameworks can lead to a better planning and utilisation of available resources.

Furthermore, it has been identified that current datasets of marine systems present a significant number of missing values. If such values are not treated, the results derived from data analysis may be unreliable and inaccurate, thus leading to bias in further steps. Such a fact cannot only facilitate the obtention of poor models that are used in decision-making processes, but also encourage the under-utilisation of data. Accordingly, several efforts have been performed to enhance the current practices that are being implemented within the shipping industry, which essentially rely on the listwise deletion approach. Accordingly, a total of three contributions have been introduced in this thesis to promote better practices. The first contribution refers to the development of a comparative methodology that examined a total of twenty widely implemented machine learning and time series forecasting algorithms. This comparative study was aimed at the analysis of potential data imputation approaches that can be successfully implemented in smart maintenance within the shipping industry context, and thus consider the unique characteristics of such a domain. Based on this study, a hybrid data imputation was also presented in order to deal with one of the most challenging data pre-processing steps that needs to be considered when utilising marine systems' sensor data, which is the identification of steady operational states. To this end, the k -means algorithm was proposed, as it is one of the most widely used clustering techniques and it has been utilised in analogous studies. Furthermore, due to the need of univariate imputation techniques caused by, for instance, the lack of predictors, the first-order Markov chain was also analysed. The

final contribution in relation to data imputation relies on the analysis of deep learning methodologies, as to the best of the author's knowledge there is no evidence that such methodologies have been implemented for such a purpose within the shipping industry. Accordingly, a VAE-regressor with LSTM was analysed. By examining such approaches, an increase in the accuracy of the data imputation task is expected, thus leading to unbiased models that fuel the decision-making processes.

As stated previously, the identification of steady operational states is of paramount importance, as raw data usually contains non-operational states that adversely alter the results outlined when performing data-driven tasks. Thus, if such states are not adequately addressed, a decrease in both computational efficiency and model effectiveness can be perceived. Although some studies have been performed, in which, for instance, GMM with EM have been introduced, the studies did not assess their performance when the raw data included other states other than engine operating regions. Moreover, the development of techniques that automate this process is of preminent importance due to the current transition from historical analysis to real-time analysis caused by the recent advancements in autonomous shipping. Consequently, a framework to identify operational steady states of marine systems was proposed. Such a framework was comprised of a time series imaging approach based on the first-order Markov chain model and connected component analysis. By exploring new automated data-driven approaches, such as the one formalised in this

thesis, an enhancement in ship and systems availability, operability, and profitability is expected by supporting more sophisticated instant decision-making strategies.

As part of this need to formalise intelligent systems within the shipping industry for enabling smart maintenance, two more novelties have been introduced in relation to diagnostic analytics. Such a need stems from the lack of time series analysis and deep learning methodologies for determining the current health of marine systems. Accordingly, an approach comprised of a Deep Long Short-Term Memory-based Variational Autoencoder in tandem with multi-level Otsu's thresholding was introduced for the application of the fault detection phase. Moreover, a time series imaging approach for fault classification of marine systems was developed as part of the fault identification phase. Such an approach was comprised of image generation through the implementation of the first-order Markov chain and an image classification task, for which a total of two models have been analysed: ResNet50V2 and Convolutional Neural Networks. By optimising the determination of the current health of marine systems, an increase in data accessibility to enable innovative data-driven strategies is expected by demonstrating how these approaches can enhance current practices in relation to O&M activities, thus providing more cost-effective processes whilst ensuring safety and security.

The final generated novelty refers to the predictive analytics module. To this end, a degradation data simulation module was developed in tandem with an ensemble model comprised of three distinct deep learning architectures: Markov-Convolutional Neural Network, 1D-CNN, and LSTM. Consequently, the challenges that the sector is currently experiencing with regards to the lack of fault data and the lack of analysis and formalisation of deep learning technologies for the implementation of RUL could be addressed. By investing research efforts in more sophisticated prognostic-based maintenance methods, a minimisation of the risk while maximising the usefulness of a system is expected.

8.3. Original Contribution to the Industry

From the AS-IS analysis conducted in section *2.2. From Reactive to Smart Maintenance*, it could be perceived that there is not yet a clear technological solution within the shipping sector related to Maintenance Analytics and Prognostics and Health Management to enable Smart Maintenance to the best of the author's knowledge. However, there are already powerful tools in the market that are expected to revolutionise the sector that address at some extent some of the issues encountered in day-to-day practices. Examples of these are Danaos Web Enterprise Suite, Cassandra, Laros, METIS Ship Connect, and Mimic Intelligent Condition Monitoring. For this reason, the author of this thesis is developing an open-source web service to integrate the developed holistic MA framework. By implementing this type of actions,

it is expected that Smart Maintenance can be democratised within the shipping sector so that the industry can benefit from the research conducted and enhance the current practices.

8.4. Assumptions and Challenges

Due to the validation requirements for adequately assessing the proposed methodology, a number of assumptions were made, and various challenges were identified. Both the assumptions made and the challenges identified are presented in bullet points hereunder.

Assumptions

- The predictors utilised for validating the multivariate imputation techniques do not present corrupted values.
- The raw data utilised in the case studies refer to normal conditions. Such an assumption was required when training semi-supervised models, such as the one presented in the fault detection section (*2.3.2.1. Fault Detection*).
- As the first-order Markov chain was utilised to encode time series into images, it is assumed that the time series analysed present the Markov property.
- Due to the lack of fault data, it was assumed that certain faults of marine systems are presented in the form of collective anomalies and degradation patterns. Specifically, it is considered that the degradation pattern can be

described as a stochastic process by considering an exponential model with Brownian motion.

Challenges

- The main challenge presented in this thesis was the data required for validating the proposed methodology. Due to the sensitivity and confidentiality of the information, the author could not obtain fault data, and thus certain simulations needed to be performed as stated precedingly.
- Analogously, certain information with regards to the available data was not provided (e.g., full description of the system, maintenance logs, and operational profiles), thus hindering the process of analysis. Also, only main engine and diesel generators data could be obtained.
- Due to the large amounts of data needed for training deep learning models, the execution time of some of the analyses was extremely large. Due to such a fact, some of the analyses were unfeasible for the available computational power and needed to be limited.

8.5. Recommendation for Future Work

Due to the number of tasks that constitute the MA framework and their respective singularity, the directions for future work are sectioned by sub-modules. Moreover,

general future work directions are also presented in bullet points based on some of the limitations encountered as indicated hereunder.

Overall MA Framework

- Develop an open-source web service to democratise the novel methodology, so that other researchers and personnel from either academia or industry can benefit from the functionalities of the MA framework without requiring a full understanding of each of the techniques and approaches performed. A beta version of the code developed in Python will be available in an online repository.
- Add more functionalities to the existing MA framework. The introduction of a decision support system, the utilisation of computer vision techniques for structural and health monitoring of ships, and the employment of AR/VR approaches for supporting training and inspection activities are examples of areas of investigations that the author is already analysing for future research development.
- Deploy the MA framework in real time as Software as a Service (SaaS). Although most of the methodologies have been developed to be deployed in real time, further validations and enhancements need to be performed in this sense. The consideration of providing the existing MA framework as an open-source software is being considered by the author.

- Implement Explainable Artificial Intelligence (EAI) for enhancing transparency and eliminating bias. Specifically, the SHAP framework is expected to be analysed.
- Perform more validation with other available datasets and parameters that reflect real-world scenarios.

Data Pre-processing

Data Imputation

- Perform a comprehensive analysis of other deep learning models to assess their imputation performance in order to assist the lack of analysis and formalisation of deep learning methodologies in relation to the application of data imputation.
- Analyse the implication of corrupted data in the imputation performance of the analysed models. In performing multivariate imputation, it has been assumed that the predictors do not contain corrupted values. However, it has been perceived that raw data usually contain anomalies, and thus its implication in the imputation performance needs to be adequately addressed.
- Apply other data pre-processing steps that have not been implemented to evaluate if the imputation performance increases (e.g., feature selection, feature extraction, and time-series denoising).

- Assess the impact of collinearity between independent variables when marine systems are being considered. Such a collinearity can facilitate a decrease in the imputation performance, and thus needs to be adequately addressed. Accordingly, the implementation of more comprehensive correlation analyses and the estimation of relevant metrics, such as Variance Inflation Factor (VIF), need to be adequately analysed.
- Consider the analysis of distinct feature extraction techniques, such as Principal Component Analysis (PCA) and Partial Least Squares (PLS), and analyse their potential implications in the imputation performance.

Identification of Steady States

- Consider optimisation techniques for the selection of the different states of the transition matrix. Whilst the selection of such states through the implementation of heuristic evaluation has been satisfactory, the author considers that the identification task can be enhanced by optimally selecting such states.
- Evaluate the implication of different pre-processing steps prior to the implementation of the proposed methodology. For instance, it has been perceived that outliers, repeated values, and noise can have a negative impact in the adequate identification of the states, and thus these need to be adequately addressed in advance.

- Consider the performance of multiple iterations and the addition of ensemble methods to enhance the outcome of the proposed methodology.
- Apply additional metrics in the validation process. Due to the difficulties in utilising metrics for unsupervised approaches, further research needs to be performed to complement the visual analysis performed in this thesis with more tangible results.
- Consider more complex pooling methodologies to analyse if the performance effectiveness of the proposed methodology increases.

Diagnostic analytics

Fault Detection

- Analyse optimisation techniques to adequately select both the architecture of the deep neural network and the different hyperparameters of the applied models. The implemented grid search algorithm presented a significant computational cost for the analysis implemented, as only one potential area of a search space has been introduced. Therefore, further search spaces need to be considered to adequately perform the optimisation task. Moreover, as part of the future of this research, evolutionary algorithms are expected to be also analysed to evaluate if the computational cost is reduced while the level of accuracy is either maintained or enhanced.

- Consider further anomaly contexts. For this inquiry, the addition of gaussian noise has been considered to alter the analysed sequences. However, the analysis of other contexts is of preeminent importance, such as degradations and contextual anomalies, which may be distinguished while performing diagnostic analytics.
- Analyse other time series similarity methods, including such measurements as transform-based similarity and time domain similarity, prior to the implementation of image thresholding. Moreover, the extraction of statistical features may need to be considered as complementary to image thresholding as part of the fault diagnosis step.
- Implement ensemble methods for anomaly detection, as it is expected that, by combining several anomaly detection techniques, the performance of this task will be enhanced.
- Consider other performance metrics to assess unsupervised learning. Although assumptions have been made to perform a semi-supervised learning task, the results obtained show that the scenarios usually considered within the maritime industry are unsupervised. Accordingly, there is a need to study such metrics to ensure the efficient and effective performance of the proposed framework and enhance it.
- Explore weather and performance characteristics of the vessel to assess the possible enhancement of the proposed framework while providing support to ship owners, operators, and managers at a strategic level.

Fault Identification

- The first-order Markov chain was implemented in this thesis as a time series imaging approach. As such, a univariate approach was presented. Further validations to perform multivariate analysis is required. In this sense, some initial validations have been performed by the author as a part of a multivariate analysis by stacking all the individual transition matrices (one for each parameter) to present them as the input of the different image classification models being analysed. The author expects to present such results in subsequent studies. Moreover, the consideration of other multivariate approaches, such as multivariate Markov chains, are being studied.
- Study other time series imaging approaches and classification models. In this thesis, a comparative study has been performed in which a total of two time series imaging approaches and four classification models have been assessed. However, due to an increased interest in PHM within the shipping industry and other comparable sectors, such as manufacturing and aerospace, numerous state-of-the-art methods are being introduced that need to be considered to continue advancing the enhancement of fault classification tasks within the shipping industry. Moreover, other elements that both complement the models developed and enhance transparency and performance also need to be considered. Examples of these are the implementation of explainable artificial

intelligence or the consideration of evolutionary algorithms for applying hyperparameters optimisation.

Predictive Analytics

RUL Prediction

- Explore and implement other time series imaging methods, such as recurrence plots or Gramian Angular Fields (GAFs).
- Analyse optimisation techniques to adequately select the hyperparameters of the deep neural networks.
- Apply more sophisticated ensemble methods.
- Validate the proposed methodology by considering a multivariate approach, as, due to the lack of fault data, only a univariate approach has been considered instead.
- Consider other forms of simulating both time series and degradation data. In this approach, an exponential model with Brownian motion to simulate degradation data has been considered. However, the degradation of the machinery may not follow this pattern in some cases.
- Validate the proposed methodology by considering real-world data. More efforts need to be performed by both academia and industry to collaborate in sharing data for research purposes. By understanding data confidentiality and the sensitive information that can be extracted from them, it is essential to

provide a secure research environment to ensure that the proposed methodologies currently being tested with simulated data can be accurate and feasible for the industry and the real-world challenges that need to be addressed.

8.6. Fulfilment of Aim and Objectives

Throughout the application of a comprehensive critical literature review (please see *Chapter 2. Literature Review*), several gaps were identified within the maritime industry. An example of these is the lack of a holistic framework constituted by diagnostic analytics (fault detection and identification) and predictive analytics (remaining useful life prediction). Consequently, to continue promoting best practices and competitiveness within the shipping industry in terms of O&M activities management, asset optimisation, emissions management, and safety and security of the personnel, a novel MA framework has been presented in this thesis. The set of objectives defined in *section 1.3.2. Aim and Objectives* and how they were fulfilled are discussed hereunder.

Objective 1: The identification of the current gaps within the shipping sector with regards to data pre-processing, and both diagnostic and predictive analytics through the application of a critical literature review. The outcomes provided from the achievement of this objective have been summarised in *Chapter 2. Literature Review*.

As outlined in said chapter, the structure of the outcome has been presented as defined in the MA concept. Accordingly, both the diagnostic analytics and predictive analytics concepts were analysed. Furthermore, as data-driven methodologies were being considered, special attention was also given to a data pre-processing step to ensure data integrity and quality. Specifically, the data imputation and the identification of operational states were comprehensively assessed due to their criticality and lack of analysis and formalisation within the context of smart maintenance in the shipping industry context (please see sections *2.3.1.1. Data Imputation* and *2.3.1.2. Identification of Operational States* for further information). With regards to the diagnostic analytics model, both fault detection and fault identification were analysed. A bibliometric analysis and a taxonomy of anomaly detection studies for fault detection were performed due to the significant number of novel approaches and publications identified. Please refer to sections *2.3.2.1. Fault Detection*, *2.3.2.2 Fault Identification*, and *Appendix C. Bibliometric and Taxonomy results of anomaly detection studies* for a more comprehensive explanation. To finalise the critical literature review, the analysis of RUL prediction studies was evaluated as part of the predictive analytics study (please see *2.3.3. Predictive Analytics*). By exploring the preceding areas of investigation, the gaps in terms of the analysis and formalisation of smart maintenance within the shipping industry could be identified. Such gaps were summarised in section *2.4. Identified Gaps*.

Objective 2: The development of a maintenance analytics framework fuelled by Machine Learning and Deep Learning (DL) algorithms to generate the novelties presented based on the gaps identified. Contributions to the analysis and formalisation of data pre-processing, diagnostic analytics, and predictive analytics are expected. By considering the identified gaps in section 2.4. *Identified Gaps*, a total of eight novelties were determined, which were not only related to one of the analysed modules (data pre-processing, diagnostic analytics, and predictive analytics), but also were associated with the development of a holistic MA framework for the diagnosis and prognosis of marine systems (please refer to section 3.2. *Novelties* for further information). Accordingly, an overall architecture of the proposed MA framework was introduced in section 3.3. *Overview of the developed frameworks*. As described precedingly, such a framework was comprised of three main modules: 1) data pre-processing, 2) diagnostic analytics, and 3) predictive analytics. As part of the data pre-processing module, a total of four contributions were presented: 1) the development of a comparative methodology for data imputation, 2) the development of a novel hybrid imputation framework constituted by operational states' identification phase, univariate imputation, and multivariate imputation, 3) the analysis of a VAE-regressor with LSTM for data imputation, and 4) the development of a novel operational states' identification framework comprised of a first-order Markov chain model with connected component analysis. As part of the diagnostic analytics module, a total of two contributions were introduced: 1) the development of an anomaly detection intelligent system for fault diagnosis of marine systems, constituted by a

LSTM-based VAE in tandem with multi-level Otsu's thresholding, and 2) the development of time series approaches for the identification of faults of marine systems. Lastly, as part of the predictive analytics module, one contribution was proposed: the development of a RUL prediction approach for fault prognosis of marine machinery, which aims to analyse the effectiveness of time series imaging and ensemble modelling for the prediction of RUL within the maritime industry while addressing the challenge of fault data unavailability. A detailed explanation of the developed MA framework is presented in *Chapters 3: Part I. Introduction to the Maintenance Analytics Methodology, 4. Methodology: Part II. The Data Pre-processing Module, and 5. Methodology: Part III. The Diagnostic analytics and Predictive Analytics Modules.*

Objective 3: The implementation of a total of 8 case studies to assess the performance of the developed methodology and demonstrate its effectiveness to address the generation of the novelties based on the gaps identified. To validate the developed MA framework, a total of 8 case studies, one for each novelty, were implemented. The first case study aimed to validate the methodology introduced in section 4.2. *Comparative Methodology of Univariate and Multivariate Imputation Techniques.* The case study referred to a total of seven machinery system parameters obtained from sensors installed on a main engine of a cargo vessel. Case study 2 aimed to validate the methodology introduced in section 4.3. *Hybrid Data Imputation Framework.* The case study referred to a total of four parameters, obtained from sensors installed on the

main engine of a cargo vessel. Case study 3 aimed to validate the methodology introduced in section 4.4. *Analysis of LSTM-based Variational Autoencoders for Regression for Data Imputation*. The case study referred to marine machinery system parameters obtained from sensors installed on a diesel generator of a tanker ship. Case study 4 aimed to validate the methodology introduced in section 4.5. *A Novel Framework for the Identification of Steady States*. The case study referred to the analysis of three diesel generators of a tanker ship. Case study 5 aimed to validate the methodology introduced in section 5.3.1. *Fault Detection*. The case study referred to a total of fourteen parameters obtained from sensors installed on a diesel generator of a tanker ship. Case study 6 aimed to validate the methodology introduced in section 5.3.2. *Fault Identification*. The case study referred to one feature that has been collected from sensors coupled to a diesel generator of a tanker ship. Case study 7 aimed to validate the methodology introduced in section 5.4. *Remaining Useful Life (RUL) Prediction Framework*. The case study referred to the turbocharger of a diesel generator of a tanker ship. Case study 8 aimed to validate the overall methodology introduced in section 3.3. *Overview of the developed frameworks*. The case study referred to the main engine of a bulk carrier. The outcomes of these case studies were presented in *Chapters 6. Case Studies and Results: Part I. Data Pre-processing Module and 7. Case Studies and Results: Part II. Diagnostic and predictive analytics Module*.

Objective 4: The introduction of future research guidelines to continue contributing to the establishment of Smart Maintenance in the shipping industry based on the discussion of the obtained results. Such an objective is being achieved in the present chapter of the thesis by introducing the fulfilment of aim and objectives, generated novelty, recommendation for future work, and assumptions and challenges.

8.7. Chapter Summary

In this final chapter, the fulfilment of the main objectives defined in the first chapter of this thesis has been discussed. Subsequently, the novelties generated have been summarised and the reflections have been presented. To continue enhancing the current version of the developed MA framework, future research and recommendations have also been introduced. To finalise, the main assumptions and challenges have been described. The subsequent pages of this thesis refer to the bibliography and appendices.

Bibliography

Abadi et al., 2016. TensorFlow: A System for Large-Scale Machine Learning.

Proceedings of the 12th USENIX Symposium on Operating Systems Design

and Implementation, pp. 1-21, url:

<https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi>.

Allianz Global Corporate & Specialty (AGCS), 2018. Safety and Shipping Review

2018. An annual review of trends and developments in shipping losses and safety.

Agrawal S., Sarkar S., Srivastava G., Maddikunta P. K. R., Gadekallu T. R., 2021.

Genetically optimized prediction of remaining useful life. Sustainable Computing: Informatics and Systems 31, pp. 1-8, doi:

<https://doi.org/10.1016/j.suscom.2021.100565>.

Aheleroff S., Xu X., Lu Y., Aristizabal M., Velásquez J. P., Joa B., Valencia Y., 2020.

IoT-enabled Smart appliances under industry 4.0: a case study. Advanced Engineering Informatics 43, pp. 1-14, doi:

<https://doi.org/10.1016/j.aei.2020.101043>.

Ahmed J., Gharakheili H. H., Raza Q., Russell C., Sivaram V., 2019. Real-Time

Detection of DNS Exfiltration and Tunneling from Enterprise Networks. 2019

IFIP/IEEE Symposium on Integrated Network and Service Management (IM), pp. 649-653, url: <https://ieeexplore.ieee.org/document/8717806>.

Ahsan S., Alemu Lema T., 2017. Remaining Useful Life Prediction of Gas Turbine Engine using Autoregressive Model. MATEC Web of Conferences 131, pp. 1-6, doi: <https://doi.org/10.1051/mateconf/201713104014>.

Alaoui-Belghiti A., Chevallier S., Monacelli E., 2019. Unsupervised Anomaly Detection Using Optimal Transport for Predictive Maintenance. Artificial Neural Networks and Machine Learning - ICANN 2019: Text and Time Series. ICANN 2019. Lecture Notes in Computer Science 11730, pp. 686-697, doi: https://doi.org/10.1007/978-3-030-30490-4_54.

Amruthnath N., Gupta T., 2018. A research study on unsupervised machine learning algorithms for early fault detection in predictive maintenance. 2018 5th International Conference on Industrial Engineering and Applications (ICIEA), pp. 355-361, doi: <https://doi.org/10.1109/IEA.2018.8387124>.

Anitha J., Pandian S. I. A., Agnes S. A., 2021. An efficient multilevel color image thresholding based on modified whale optimization algorithm. Expert Systems with Applications 178, pp. 1-19, doi: <https://doi.org/10.1016/j.eswa.2021.115003>.

Aslam S., Michaelides M. P., Herodotou H., 2020. Internet of Ships: A Survey on Architectures, Emerging Applications, and Challenges. IEEE Internet of

Things Journal 7-10, pp. 9714-9727, doi:
<https://doi.org/10.1109/JIOT.2020.2993411>.

AUTOSHIP. Last accessed 31st January 2022: <https://www.autoship-project.eu/>.

Azimi I., Pahikkala T., Rahmani A. M., Niela-Vilén H., Axelin A., Liljeberg P., 2019. Missing data resilient decision-making for healthcare IoT through personalization: A case study on maternal health. Future Generation Computer Systems 96, pp. 297-308, URL <https://doi.org/10.1016/j.future.2019.02.015>.

Balakrishnan S. M., Sangaiah A. K., 2018. Chapter 6 – aspect oriented modelling of missing data imputation for data imputation for Internet of Things (IoT) based healthcare infrastructure. Intelligent Data-Centric, pp. 135-145, doi: <https://doi.org/10.1016/B978-0-12-813314-9.00006-2>.

Bashir F., Wei H., 2018. Handling missing data in multivariate time series using a vector autoregressive model-imputation (VAR-IM) algorithm. Neurocomputing 276, pp. 23-30, URL <https://doi.org/10.1016/j.neucom.2017.03.097>.

Berghout T., Mouss L-L., Kadri O., Saïdi L., Benbouzid M., 2020. Aircraft engines Remaining Useful Life prediction with an adaptive denoising online sequential Extreme Learning Machine. Engineering Applications of Artificial intelligence 96, pp. 1-10, doi: <https://doi.org/10.1016/j.engappai.2020.103936>.

- Bokde N., Beck M. W., Martínez Álvarez F., Kulat K., 2018. A novel imputation methodology for time series based on pattern sequence forecasting. *Pattern Recognition Letters* 116, pp. 88-96, URL <https://doi.org/10.1016/j.patrec.2018.09.020>.
- Borith T., Bakhit S., Nasridinov A., Yoo K., 2020. Prediction of Machine Inactivation Status Using Statistical Feature Extraction and Machine Learning. *Appl. Sci.* 10, pp. 1-18, doi: <https://doi.org/10.3390/app10217413>.
- Bose S. K., Kar B., Roy M., Gopalakrishnan P. K., Basu A., 2019. ADEPOS: Anomaly Detection based Power Saving for Predictive Maintenance using Edge Computing. 2019 24th Asia and South Pacific Design Automation Conference, pp. 1-10, doi: <https://doi.org/10.1145/3287624.3287716>.
- Brandsæter A., Vanem E., Glad I. K. (2017). Cluster Based Anomaly Detection with Applications in the Maritime Industry. 2017 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC), pp. 328-333, doi: <https://doi.org/10.1109/SDPC.2017.69>.
- Brandsæter A., Vanem E., Glad I. K. (2019). Efficient on-line anomaly detection for ship systems in operation. *Expert Systems With Applications* 121, pp. 418-437, doi: <https://doi.org/10.1016/j.eswa.2018.12.040>.
- Cai C., Weng X., Zhang C., 2017. A novel approach for marine diesel engine fault diagnosis. *Cluster Computing* 20, pp. 1691-1702, doi: <https://doi.org/10.1007/s10586-017-0748-0>.

- Cao Y. Ding Y., Jia M. Tian R., 2021. A novel temporal convolutional network with residual self-attention mechanism for remaining useful life prediction of rolling bearings. *Reliability Engineering and System Safety* 215, pp. 1-13, doi: <https://doi.org/10.1016/j.ress.2021.107813>.
- Carrega A., Cipollini F., Oneto L., 2019. Simple continuous optimal regions of the space of data. *Neurocomputing* 349, pp. 91-104, doi: <https://doi.org/10.1016/j.neucom.2019.03.081>.
- Cheliotis M., Gkerekos C., Lazakis I., Theotokatos G., 2019. A novel data condition and performance hybrid imputation method for energy efficient operations of marine systems. *Ocean Engineering* 188, pp. 1-14, doi: <https://doi.org/10.1016/j.oceaneng.2019.106220>.
- Cheliotis M. F., 2020. A Compound Novel Data-Driven and Reliability-Based Predictive Maintenance Framework for Ship Machinery Systems. Doctor of Philosophy. University of Strathclyde. Glasgow, UK.
- Cheliotis M., Lazakis I., Theotokatos G., 2020b. Machine learning and data-driven fault detection for ship systems operations. *Ocean Engineering* 216, pp. 1-17, doi: <https://doi.org/10.1016/j.oceaneng.2020.107968>.
- Cheliotis M., Lazakis I., Cheliotis A., 2022. Bayesian and machine learning-based fault detection and diagnostics for marine applications. *Ships and Offshore Structures*, pp. 1-14, doi: <https://doi.org/10.1080/17445302.2021.2012015>.

Cheng X., Ellefsen A. L., Li G., Holmeset F. T., Zhang H., Chen S., 2019. A Step-wise Feature Selection Scheme for a Prognostics and Health Management System in Autonomous Ferry Crossing Operation. 2019 IEEE International Conference on Mechatronics and Automation (ICMA), pp. 1877-1882, doi: <https://doi.org/10.1109/ICMA.2019.8816219>.

Cheng X., Yang X., Chen Z., 2019b. Research on FPGA filtering method of Gearbox fault signal. IOP Conference Series: Materials Science and Engineering 490, doi: <https://doi.org/10.1088/1757-899x/490/5/052024>.

Ching, W. Huang, X. Ng, M. Siu, T., 2006. Markov Chains. Models, Algorithms and Applications. 2nd ed. Singapore: Springer, pp. 1-19.

Chivers B. D., Wallbank J., Cole S. J., Sebek O., Stanley S., Fry M., Leontidis G., 2020. Imputation of missing sub-hourly precipitation data in a large sensor network: A machine learning approach. Journal of Hydrology 588, pp. 1-12, URL <https://doi.org/10.1016/j.jhydrol.2020.125126>.

Collet F. et al., 2015. Keras, url: <https://keras.io>.

Chong A., Lam K. P., Xu W., Karaguzel O. T., Mo Y., 2016. Imputation of missing values in building sensor data. Building Performance Modeling Conference, pp. 1-9.

- Coraddu A., Lim S., Oneto I., Pazouki K., Norman R., Murphy A. J. (2019). A novelty detection approach to diagnosing hull and propeller fouling. *Ocean Engineering* 176, pp. 65-73, doi: <https://doi.org/10.1016/j.oceaneng.2019.01.054>.
- Cui Y., Bangalore P., Tjernberg L. B., 2018. An Anomaly Detection Approach Based on Machine Learning and SCADA Data for Condition Monitoring of Wind Turbines. 2018 IEEE International Conference on Probabilistic Methods Applied to Power Systems (PMAPS), pp. 1-6, doi: <https://doi.org/10.1109/PMAPS.2018.8440525>.
- Dalheim Ø. Ø., Steen S., 2020. Preparation of in-service measurement data for ship operation and performance analysis. *Ocean Engineering* 212, pp. 1-17, doi: <https://doi.org/10.1016/j.oceaneng.2020.107730>.
- Dalheim Ø. Ø., Steen S., 2020b. A computationally efficient method for identification of steady state in time series data from ship monitoring. *Journal of Ocean Engineering and Science* 5, pp. 333-345, doi: <https://doi.org/10.1016/j.joes.2020.01.003>.
- Danaos, 2022. Maritime Software Solutions | Danaos Management, accessed 5 December 2022, url: <https://www.danaos.gr/>.
- DeepSea, 2022. Cassandra, 360° Technical fleet optimisation. Accessed 5 December 2022, url: <https://www.deepsea.ai/cassandra/>.

Department for International Trade, 2019. Promoting the UK's world-class global maritime offer: Trade and Investment 5-year plan, pp. 1-52.

Department for Transport, 2019. Technology and Innovation in UK Maritime: The case of Autonomy, pp. 1-60.

Di Maio F., Hu J., Tse P., Pecht M., Tsui K., Zio E., 2012. Ensemble-approaches for clustering health status of oil sand pumps. *Expert Systems with Applications* 39, pp 4847-4859, doi: <https://doi.org/10.1016/j.eswa.2011.10.008>.

Djedidi O., Djeziri M. A., Benmoussa S., 2021. Remaining useful life prediction in embedded systems using an online auto-updated machine learning based modeling. *Microelectronics Reliability* 119, pp. 1-10, doi: <https://doi.org/10.1016/j.microrel.2021.114071>.

DNV GL, 2014. DNV GL strategic research & innovation position paper 6-2014. Beyond condition monitoring in the maritime industry, pp. 1-32.

Doğru A., Bouarfa S., Arizar R., Aydoğlan R., 2020. Using Convolutional Neural Networks to Automate Aircraft Maintenance Visual Inspection. *Aerospace* 7, pp. 1-22, doi: <https://doi.org/10.3390/aerospace7120171>.

Ducharlet K., Travé-Massuyès L., Le Lann M. V., Miloudi Y., 2020. A Multi-phase Iterative Approach for Anomaly Detection and Its Agnostic Evaluation. *Trends in Artificial Intelligence Theory and Applications. Artificial*

Intelligence Practices. IEA/AIE, pp. 505-517, doi:
https://doi.org/10.1007/978-3-030-55789-8_44.

Ellefsen A. L., Cheng X., Holmeset F. T., Æsøy V., Zhang H., Ushakov, 2019. Automatic Fault Detection for Marine Diesel Engine Degradation in Autonomous Ferry Crossing Operation. 2019 IEEE International Conference on Mechatronics and Automation (ICMA), pp. 2195-2200, doi:
<https://doi.org/10.1109/ICMA.2019.8816600>.

Ellefsen A. L., Bjørlykhaug E., Æsøy V., Zhang H., 2019b. An Unsupervised Reconstruction-Based Fault Detection Algorithm for Maritime Components. IEEE Access 7, pp. 16101-16109, doi:
<https://doi.org/10.1109/ACCESS.2019.2895394>.

Ellefsen A. L., Bjørlykhaug E., Æsøy V., Ushakov S., Zhang H., 2019c. Remaining useful life predictions for turbofan engine degradation using semi-supervised deep architecture. Reliability Engineering and System Safety 183, pp. 240-251, doi: <https://doi.org/10.1016/j.res.2018.11.027>.

Ellefsen A. L., Han P., Cheng X., Holmeset F. T., Æsøy V., Zhang H., 2020. Online Fault Detection in Autonomous Ferries: Using Fault-Type Independent Spectral Anomaly Detection. IEEE Transactions on Instrumentation and Measurement 69, pp. 8216-8225, doi:
<https://doi.org/10.1109/TIM.2020.2994012>.

- Erhan L., Ndubuaku M., Di Mauro M., Song W., Chen M., Fortino G., Bagdasar O., Liotta A., 2021. Smart anomaly detection in sensor systems: A multi-perspective view. *Information Fusion* 67, pp. 64-79, doi: <https://doi.org/10.1016/j.inffus.2020.10.001>.
- European Space Agency, 2022. Foresight Marine, accessed 5 December 2022, url: <https://business.esa.int/projects/foresight-marine>.
- Fahim S. R., Sarker S. K., Muyeen S. M., Sheikh M. R. I., Das S. K., Simoes M., 2021. A Robust Self-Attentive Capsule Network for Fault Diagnosis of Series-Compensated Transmission Line. *IEEE Transactions on Power Delivery* 36, pp. 3846-3857, doi: <https://doi.org/10.1109/TPWRD.2021.3049861>.
- Fahim S. R., Sarker Y., Sarker S. K., Sheikh M. R. I., Das S. K., 2020. Self-attention convolutional neural network with time series imaging based feature extraction for transmission line fault detection and classification. *Electric Power Systems Research* 187, pp. 1-12, doi: <https://doi.org/10.1016/j.epsr.2020.106437>.
- Fahim S. R., Islam Sarker M. R., Arifuzzaman M., Hosen M. S., Sarker S. K., Das S. K., 2020b. A novel Approach to Fault Diagnosis of High Voltage Transmission line – A Self Attentive Convolutional Neural Network Model. 2020 IEEE Region 10 Symposium, pp. 1329-1332, doi: <https://doi.org/10.1109/TENSYMP50017.2020.9230660>.
- Fahim S. R., Muyeen S. M., Sarker Y., Sarker S. K., Das S. K., 2021b. An Agreement Based Dynamic Routing Method for Fault Diagnosis in Power Network with

Enhanced Noise Immunity. 2021 31st Australasian Universities Power Engineering Conference, pp. 1-5 doi: <https://doi.org/10.1109/AUPEC52110.2021.9597762>.

Fahim S. R., Niloy S., Shatil A. H., Hazari M. R., Sarker S. K., Das S. K., 2021c. An Unsupervised Protection Scheme for Overhead Transmission Line with Emphasis on Situations during Line and Source Parameter Variation. International Conference on Robotics, Electrical, and Signal Processing Techniques, pp. 758-762, doi: <https://doi.org/10.1109/ICREST51555.2021.9331170>.

Farbiz F., Miaolong Y., Yu Z., 2020, A Cognitive Analytics based Approach for Machine Health Monitoring, Anomaly Detection, and Predictive Maintenance. 2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA), pp. 1104-1109, doi: <https://doi.org/10.1109/ICIEA48937.2020.9248409>.

Fekade B., Maksymyuk T., Kyryk M., Jo M, 2018. Probabilistic recovery of incomplete sensed data in IoT. IEEE Internet of Things Journal 5, pp. 2282-2292, URL <https://ieeexplore.ieee.org/document/7987674>.

Fernandes M., Canito A., Corchado J. M, Marreiros G., 2019. Fault Detection Mechanism of a Predictive Maintenance Based on Autoregressive Integrated Moving Average Models. 16th International Conference in Distributed

Computing and Artificial Intelligence (DCAI), pp. 171-180, doi:
https://doi.org/10.1007/978-3-030-23887-2_20.

Gkerekos C., Theotokatos G., Bujorianu L. M., Boulougouris E., Vassalos D.,
Carballedo B., McCluskey S., Coats T., Sloan S., 2019. Digitalisation in the
maritime sector: a stakeholders' pulse check. Paper presented at Marine
Industry 4.0, Rotterdam, Netherlands, pp. 1-7.

Gkerekos C., Lazakis I., Theotokatos G., 2019b. Machine learning models for
predicting ship main engine Fuel Oil Consumption: a comparative study.
Ocean Engineering 188, pp. 1-14, doi:
<https://doi.org/10.1016/j.oceaneng.2019.106282>.

Grandini M., Bagli E., Visani G., 2020. Metrics for Multi-Class Classification: An
Overview. Computer Science. ArXiv, pp. 1-17, URL:
<https://arxiv.org/pdf/2008.05756.pdf>.

Gribbestad M., Hassan M. U., Hameed I. A., 2021. Transfer Learning for Prognostic
and Health Management (PHM) of Marine Air Compressors. Journal of Marine
Science and Engineering 9, pp.1-20, doi: <https://doi.org/10.3390/jmse9010047>.

Hadeed S. J., O'Rourke M. K., Burgess J. L., Harris R. B., Canales R. A., 2020.
Imputation methods for addressing missing data in short-term monitoring of
air pollutants. Science of the Total Environment 730, pp. 1-7, URL
<https://doi.org/10.1016/j.scitotenv.2020.139140>.

- Han P., Ellefsen A. L., Li G., Æsøy V, Zhang H., 2021. Fault Prognostics Using LSTM Networks: Application to Marine Diesel Engine. *IEEE Sensors Journal*, pp. 1-8, doi: <https://doi.org/10.1109/JSEN.2021.3119151>.
- Han P., Ellefsen A. L., Li G., Helmeset F. T., Zhang H., 2021b. Fault Detection With LSTM-Based Variational Autoencoder for Maritime Components. *IEEE Sensors Journal* 21, pp. 21903-21912, doi: <https://doi.org/10.1080/17445302.2021.2012015>.
- Harrou F., Dairi A., Taghezouit B., Sun Y., 2019. An unsupervised monitoring procedure for detecting anomalies in photovoltaic systems using a one-class Support Vector Machine. *Solar Energy* 179, pp. 48-58, doi: <https://doi.org/10.1016/j.solener.2018.12.045>.
- He K., Zhang X., Ren S., Sun J., 2016. Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778, doi: <https://doi.org/10.1109/CVPR.2016.90>.
- He K., Zhang X., Ren S., Sun J., 2016b. Identity Mapping in Deep Residual Networks. *Computer Vision – ECCV 2016. ECCV 2016. Lecture Notes in Computer Science*, vol 9908. Springer, Cham, pp. 630-645, doi: https://doi.org/10.1007/978-3-319-46493-0_38.
- Hegde H. Shimpi N., Panny A., Glurich I., Christie P., Acharya A., 2019. MICE vs PPCA: Missing data imputation in healthcare. *Informatics in Medicine Unlocked* 17, pp. 1-8, URL <https://doi.org/10.1016/j.imu.2019.100275>.

- Helbing G., Ritter M., 2018. Deep Learning for fault detection in wind turbines. *Renewable and Sustainable Energy Reviews* 98, pp. 189-198, doi: <https://doi.org/10.1016/j.rser.2018.09.012>.
- Hochreiter S., Schmidhuber, J., 1997. Long short-term memory 9(8): pp. 1735-1780, doi: <https://doi.org/10.1162/neco.1997.9.8.1735>.
- Hodžić A., Škulj D., Čaušević, A., 2020. Data-driven Anomaly Detection for Railway Propulsion Control Systems. *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, pp. 4351-4356, doi: <https://doi.org/10.1109/IECON43393.2020.9255026>.
- Hou L., Zhang J., Du B., 2020. A Fault Diagnosis Model of Marine Diesel Engine Fuel Oil Supply System Using PCA and Optimized SVM. *Journal of Physics: Conference Series* 1576, pp. 1-9, doi: <https://doi.org/10.1088/1742-6596/1576/1/012045>.
- Hsu J., Wang Y., Lin K., Chen M., Hsu J. H., 2020. Wind Turbine Fault Diagnosis and Predictive Maintenance Through Statistical Process Control and Machine Learning. *IEEE Access* 8, pp. 23427-23439, doi: <https://doi.org/10.1109/ACCESS.2020.2968615>.
- Hyndman R. J., Athanasopoulos G., 2020. *Forecasting: principles and practice*. 3rd ed. Melbourne: Otexts, URL <https://otexts.com/fpp3/>.

- Ichimura Y., Dalaklis D., Kitada M., Christodoulou A., 2022. Shipping in the era of digitalization: Mapping the future strategic plans of major maritime commercial actors. *Digital Business* 2(1), pp. 1-11, doi: <https://doi.org/10.1016/j.digbus.2022.100022>.
- Imbassahy D. W. D., Marques H. C., Rocha G. C., Martinetti A., 2020. Empowering Predictive Maintenance: A Hybrid Method to Diagnose Abnormal Situations. *Appl. Sci.* 10, pp. 1-27, doi: <https://doi.org/10.3390/app10196929>.
- Inagaki K. Hayamizu S., Tamura S., 2021. Using Deep-Learning Approach to Detect Anomalous Vibrations of Press Working Machine. *Sensors and Instrumentation, Aircraft/Aerospace, Energy Harvesting & Dynamic Environments Testing* 7, pp. 229-232, doi: https://doi.org/10.1007/978-3-030-47713-4_21.
- Izonin I., Kryvinska N., Tkachenko R., Zub K., 2019. An approach towards missing data recovery within IoT smart system. *Procedia Computer Science* 155, pp. 11-18, doi: <https://doi.org/10.1016/j.procs.2019.08.006>.
- James Fisher and Sons plc, 2022. Mimic: intelligent condition monitoring solutions and services, accessed 5 December 2022, url: <https://jfmimic.co.uk>.
- Jaramillo Jimenez V., Bouhmala N., Gausdal A. H., 2020. Developing a predictive maintenance model for vessel machinery. *Journal of Ocean Engineering and Science* 5, pp. 358-386, doi: <https://doi.org/10.1016/j.joes.2020.03.003>.

- Jasiulewicz-Kaczmarek M., Gola A., 2019. Maintenance 4.0 Technologies for Sustainable Manufacturing - an Overview. IFAC-PapersOnLine 52-10, pp. 91-96, doi: <https://doi.org/10.1016/j.ifacol.2019.10.005>.
- Kang Z., Catal C., Tekinerdogan B., 2021. Remaining Useful Life (RUL) Prediction of Equipment in Production Lines Using Artificial Neural Networks. Sensors 21, pp. 1-20, doi: <https://doi.org/10.3390/s21030932>.
- Karatuğ Ç., Arslanoğlu Y., 2022. Development of condition-based maintenance strategy for fault diagnosis for ship engine systems. Ocean Engineering 256, pp. 1-8, doi: <https://doi.org/10.1016/j.oceaneng.2022.111515>.
- Karim R., Westerberg J., Galar D., Kumar U., 2016. Maintenance Analytics - The New Know in Maintenance. IFAC-PapersOnLine 49-28, pp. 214-219, doi: <https://doi.org/10.1016/j.ifacol.2016.11.037>.
- Kiangala K. s., Wang Z., 2020. An Effective Predictive Maintenance Framework for Conveyor motors Using Dual Time-Series Imaging and Convolutional Neural Network in an Industry 4.0 Environment. IEEE Access 8, pp. 121033-121049, doi: <https://10.1109/ACCESS.2020.3006788>.
- Kim Y., Steen S., Muri H., 2022. A novel method for estimating missing values in ship principal data. Ocean Engineering 251, pp. 1-25, doi: <https://doi.org/10.1016/j.oceaneng.2022.110979>.

- Kingma D. P., Welling M., 2013. Auto-encoding variational Bayes. arXiv.org, pp. 1-14, <https://arxiv.org/abs/1312.6114>.
- Kolokas N., Vafeiadis T., Ionnidis D., Tzovaras D., 2019. Anomaly Detection in Aluminium Production with Unsupervised Machine Learning Classifiers. 2019 IEEE International Symposium of INnovations in Intelligent SysTems and Applications (INSTA), pp. 1-6, doi: <https://doi.org/10.1109/INISTA.2019.8778419>.
- Kolokas N., Vafeiadis T., Ionnidis D., Tzovaras D., 2020. A generic fault prognostic algorithm for manufacturing industries using unsupervised machine learning classifiers. Simulation Modelling Practice and Theory 103, pp. 1-13, doi: <https://doi.org/10.1016/j.simpat.2020.102109>.
- Kotu V., Deshpande B., 2019. Chapter 12 - time series forecasting. Data Science. 2nd ed. Burlington: Morgan Kaufmann, URL <https://doi.org/10.1016/B978-0-12-814761-0.00012-5>.
- Kuhn M., Johnson K., 2016. Applied Predictive Modeling. 5th ed. New York: Springer Science + Business Media LLC, pp. 27-58.
- Kyriakou C., Christodoulou S. E., Dimitriou L., 2019. Smartphone-based Pothole Detection Utilizing Artificial Neural Networks. Journal of Infrastructure Systems 25, pp. 1-8, doi: [https://doi.org/10.1061/\(ASCE\)IS.1943-555X.0000489](https://doi.org/10.1061/(ASCE)IS.1943-555X.0000489).

- Lachin J. M., 2016. Fallacies of last observation carried forward analyses. Clin Trials, 13(2). pp. 161-168, doi: <https://doi.org/10.1177/1740774515602688>.
- Langone R., Alzate C., De Ketelaere B., Vlasselaer J., Meert W., Suykens J. A. K, 2015. LS-SVM based spectral clustering and regression for predicting maintenance of industrial machines. Engineering Applications of Artificial Intelligence 37, pp. 268-278, doi: <https://doi.org/10.1016/j.engappai.2014.09.008>.
- Lawler, G., 2006. Introduction to Stochastic Processes. 2nd ed. Boca Raton: Taylor & Francis Group, LLC, pp.1-12.
- Lazakis I., Dikis K., Michala, A.L., Theotokatos, G., 2016. Advanced ship systems condition monitoring for enhanced inspection, maintenance and decision making in ship operations. Transportation Research Procedia 14, pp. 1679-1688, doi: <https://doi.org/10.1016/j.trpro.2016.05.133>.
- Lazakis I., Gkerekos, C., Theotokatos, G., 2018a. Investigating an SVM-driven, one-class approach to estimating ship system condition. Ship Offshore Structures 14, pp. 432-441, doi: <https://doi.org/10.1080/17445302.2018.1500189>.
- Lazakis I., Raptodimos Y., Varelas, T., 2018b. Predicting ship machinery system condition through analytical reliability tools and artificial neural networks. Ocean Engineering 152, pp. 404-415, doi: <https://doi.org/10.1016/j.oceaneng.2017.11.017>.

- Lei Y., Li N., Guo L., Li N., Yan T., Lin J., 2018. Machinery health prognostics: A systematic review from data acquisition to RUL prediction. *Mechanical Systems and Signal Processing* 104, pp. 799-834, doi: <https://doi.org/10.1016/j.ymssp.2017.11.016>.
- Liao P-S., Chen T-S., Chung P-C., 2001. A fast algorithm for multilevel thresholding. *Journal of Information Science and Engineering* 17, pp. 713-727, url: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.85.3669>.
- Li X., Zhang T., Liu Y., 2019. Detection of Voltage Anomalies in Spacecraft Storage Batteries Based on a Deep Belief Network. *Sensors* 19, pp. 1-21, doi: <https://doi.org/10.3390/s19214702>.
- Li X., Chen P., Jing L., He Z., Yu G., 2020. SwissLog: Robust and Unified Deep Learning Based Log Anomaly Detection for Diverse Faults. 2020 IEEE 31st International Symposium on Software Reliability Engineering (ISSRE), pp. 92-103, doi: <https://doi.org/10.1109/ISSRE5003.2020.00018>.
- Li X., Kang Y., Li F., 2020b. Forecasting with time series imaging. *Expert Systems with Applications* 160, pp. 1-13, doi: <https://doi.org/10.1016/j.eswa.2020.113680>.
- Li Y., Huang X., Ding P., Zhao C., 2021. Wiener-based remaining useful life prediction of rolling bearings using improved Kalman filtering and adaptive modification. *Measurement* 182, pp. 1-17, doi: <https://doi.org/10.1016/j.measurement.2021.109706>.

- Liu G., Bao H., Han B., 2018. A Stacked Autoencoder-Based Deep Neural Network for Achieving Gearbox Fault Diagnosis. *Mathematical Problems in Engineering* 2018(5), pp. 1-10, doi: <https://doi.org/10.1155/2018/5105709>.
- Liu C., Gryllias K., 2020. A semi-supervised Support Vector Data Description-based fault detection method for rolling element bearings based on cyclic spectral analysis. *Mechanical Systems and Signal Processing* 140, pp. 1-24, doi: <https://doi.org/10.1016/j.ymssp.2020.106682>.
- Liu Y., Dillon T., Yu W., Rahayu W., Mosafa F, 2020b. Missing value imputation for Industrial IoT sensor data with large gaps. *IEEE Internet of Things Journal* (Early Access), URL <https://ieeexplore.ieee.org/abstract/document/8976165>.
- Liu Y., Pang Z., Karlsson M., Gong S., 2020c. Anomaly detection based on machine learning in IoT-based vertical plant wall for indoor climate control. *Building and Environment* 183, pp. 1-13, doi: <https://doi.org/10.1016/j.buildenv.2020.107212>.
- Lloyd's List, 2021. Outlook 2021. Shipping accelerates towards an uncertain future, pp. 1-66.
- London Economics, NLA International, Marine South East, 2021. Consultancy Research into the UK Maritime Technology Sector, pp. 1-151.
- Lutz M., Vogt S., Berkhout V., Faulstich S., Dienst S., Gück C., Ortega A., 2020. Evaluation of Anomaly Detection of an Autoencoder Based on Maintenance

Information and Scada-Data. Energies, pp. 1-18, doi:
<https://doi.org/10.3390/en13051063>.

Madakyaru M., Harrou F., Sun Y., 2019. Monitoring Distillation Column Systems Using Improved Nonlinear Partial Least Squares-Based Strategies. IEEE Sensors Journal 19, pp. 11697-11705, doi:
<https://doi.org/10.1109/JSEN.2019.2936520>.

Makridis G., Kyriazis D., Plitsos S., 2020. Predictive maintenance leveraging machine learning for time-series forecasting in the maritime industry. 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), pp.1-8, doi: <https://doi.org/10.1109/ITSC45102.2020.9294450>.

Marine Accident Recommendations and Statistics (MAIB), 2020. Annual report, pp. 1-79.

Martin-del-Campo S., Sandin F., Schnabel S, 2019. Kinematic Frequencies of Rotating Equipment Identified with Sparse Coding and Dictionary Learning. Annual Conference of the PHM Society 11, pp. 1-8, doi:
<https://doi.org/10.36001/phmconf.2019.v11i1.837>.

Masmoudi O., Jaoua M., Jaoua A., Yacount S., 2021. Data Preparation in Machine Learning for Condition-based Maintenance. Journal of Computer Science 17, pp. 525-538, doi: <https://doi.org/10.3844/jcssp.2021.525.538>.

METIS, 2022. METIS ship connect, accessed 5 December 2022, url:
<https://www.metis.tech/metis-ship-connect/>.

Michałowska K., Riemer-Sørensen S., Sterud C., Hjellset O. M., 2021. Anomaly Detection with Unknown Anomalies: Application to Maritime Machinery 54, pp. 105-111, doi: <https://doi.org/10.1016/j.ifacol.2021.10.080>.

Miki D., Demachi K., 2020. Bearing fault diagnosis using weakly supervised long short-term memory. Journal of Nuclear Science and Technology 9, pp. 1091-1100, doi: <https://doi.org/10.1080/00223131.2020.1761473>.

Morariu C., Morariu O., Răileanu S., Borangiu T., 2020. Machine learning for predictive scheduling and resource allocation in large scale manufacturing systems. Computers in Industry, pp. 1-13 doi: <https://doi.org/10.1016/j.compind.2020.103244>.

Mulongo J., Atemkeng M., Ansah-Narh T., Rockefeller R., Nguenngang G. M., Garuti M. A., 2019. Anomaly Detection in Power Generation Plants Using Machine Learning and Neural Networks. Applied Artificial Intelligence 34, pp. 64-79, doi: <https://doi.org/10.1080/08839514.2019.1691839>.

Munir M., Siddiqui S. A., Chattha M. A., Dengel A., Ahmed S., 2019. FuseAD: Unsupervised Anomaly Detection in Streaming Sensors Data by Fusing

Statistical and Deep Learning Models. *Sensors* 19, pp. 1-15, doi:
<https://doi.org/10.3390/s19112451>.

Nair V., Hinton G. E., 2010. Rectified linear units improve restricted boltzmann machines. *ICML'10: Proceedings of the 27th International Conference on International Conference on Machine Learning*, pp. 807-814.

Noor N. M., Abudllah M. M. A. B., Yahaya A. S., Ramli N. A., 2014. Comparison of linear interpolation method and mean method to replace the missing values in environmental data set. *Material Science Forum* 803, pp. 278-281, doi:
<https://doi.org/10.4028/www.scientific.net/MSF.803.278>.

Offiong N. M., Wu Y., Memon F. A., 2020. Predicting failures in electronic water taps in rural sub-Saharan African communities: an LSTM-based approach. *Water Science & Technology* 82, pp. 2776–2785, doi:
<https://doi.org/10.2166/wst.2020.542>.

Oikonomou S., 2021. Development of an advanced artificial intelligent reliability analysis tool to enhance ship operations and maintenance activities. Master of Philosophy. University of Strathclyde. Glasgow, UK.

Oliveira D. F. N., Vismari L. F., de Almeida J. R., Cugnasca P. S., Camargo J. B., Marreto E., Doimo D. R., de Almeida L. P. F, Gripp R., Neves M. M., 2019. Evaluating Unsupervised Anomaly Detection Models to Detect Faults in Heavy Haul Railway Operations. 2019 18th IEEE International Conference On

Machine Learning and Applications (ICMLA), pp. 1016-1022, doi:
<https://doi.org/10.1109/ICMLA.2019.00172>.

Pedregosa F et al. Scikit-learn: Machine Learning in Python. JMLR 12 (2011): 2825-2830.

Perera L. P., Mo B., 2016. Data analysis on marine engine operating regions in relation to ship navigation. Ocean Engineering 128, pp. 163-172, doi:
<https://doi.org/10.1016/j.oceaneng.2016.10.029>.

Pereira J., Silveira M., 2018. Unsupervised Anomaly Detection in Energy Time Series Data Using Variational Recurrent Autoencoders with Attention. 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 1275-1282, doi: <https://doi.org/10.1109/ICMLA.2018.00207>.

Pratama I., Permanasari A. E., Ardiyanto I., Indrayani R., 2016. A review of missing values handling methods on time-series data. International Conference on Information Technology Systems and Innovation (ICITSI), pp. 1-6, URL <https://ieeexplore.ieee.org/document/7858189>.

Prisma Electronics, 2022. Laros by Prisma Electronics, accessed 5 December 2022, url: <https://www.laros.gr/>.

Privault N., 2013. Understanding Markov Chains. Examples and Applications. Singapore: Springer, pp. 77-94.

- Priya Stella Mary I., Arockiam L., 2017. Imputing the missing data in IoT based on the spatial and temporal correlation. IEEE International Conference on Current Trends in Advanced Computing (ICCTAC), pp. 1-4, URL <https://ieeexplore.ieee.org/document/8249990>.
- Quatrini E., Costantino F., Di Gravio G., Patriarca R., (2020). Machine learning for anomaly detection and process phase classification to improve safety and maintenance activities. Journal of Manufacturing Systems 56, pp. 117-132, doi: <https://doi.org/10.1016/j.jmsy.2020.05.013>.
- Rahimzadeh M., Attar A., 2020. A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2. Informatics in Medicine Unlocked 19, pp. 1-9, doi: <https://doi.org/10.1016/j.imu.2020.100360>.
- Ramadhan M. S., Hassan K. A., 2021. Remaining useful life prediction using an integrated Laplacian-LSTM network on machinery components. Applied Soft Computing 112, doi: <https://doi.org/10.1016/j.asoc.2021.107817>.
- Ramchandran A., Sangaiah A. K., 2018. Chapter 11 – Unsupervised Anomaly Detection for High Dimensional Data – and Exploratory Analysis. Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications. Intelligent Data-Centric Systems, pp. 233-251, doi: <https://doi.org/10.1016/B978-0-12-813314-9.00011-6>.

Raptodimos Y., Lazakis I., 2018. Using artificial neural network self-organising map for data clustering of marine engine condition monitoring applications. *Ships Offshore Structures* 13, pp. 649-656, doi: <https://doi.org/10.1080/17445302.2018.1443694>.

Raptodimos Y., 2018b. Combination of Reliability Tools and Artificial Intelligence in a Hybrid Condition Monitoring Framework for Ship Machinery Systems. Doctor of Philosophy. University of Strathclyde. Glasgow, UK.

Raptodimos Y., Lazakis I., 2019. Application of NARX neural network for predicting marine engine performance parameters. *Ships Offshore Structures* 15, pp. 443-452, doi: <https://doi.org/10.1080/17445302.2019.1661619>.

Riazi M., Zaiane O., Takeuchi T., Maltais A., Günther J., Lipsett M., 2019. Detecting the Onset of Machine Failure Using Anomaly Detection Methods. International Conference on Big Data Analytics and Knowledge Discovery, pp. 3-12, doi: https://doi.org/10.1007/978-3-030-27520-4_1.

Roy M. Bose S. K., Kar B., Gopalakrishnan P. K., Basu A., 2018. A Stacked Autoencoder Neural Network based Automated Feature Extraction Method for Anomaly detection in On-line Condition Monitoring. 2018 IEEE Symposium Series on Computational Intelligence (SSCI), doi: <https://doi.org/10.1109/SSCI.2018.8628810>.

Russakovsky O., Deng J., Su H., Krause J., Satheesh S., Ma S., Huang Z., Karpathy A., Khosla A., Bernstein M., Berg A. C., Fei-Fei L., 2015. ImageNet Large

- Scale Visual Recognition Challenge. *International Journal of Computer Vision* 115, pp. 211-252, doi: <https://doi.org/10.1007/s11263-015-0816-y>.
- Sahin, A. Sen, Z. 2001. First-order Markov chain approach to wind speed modelling. *Journal of Wind Engineering and Industrial Aerodynamics* 89, pp. 263-269, doi: [https://doi.org/10.1016/S0167-6105\(00\)00081-7](https://doi.org/10.1016/S0167-6105(00)00081-7).
- Santolamazza A., Cesarotti V., Introna V., 2018. Anomaly detection in energy consumption for Condition-Based maintenance of Compressed Air Generation systems: an approach based on artificial neural networks. *IFAC-PapersOnLine* 51, pp. 1131-1136, doi: <https://doi.org/10.1016/j.ifacol.2018.08.439>.
- Senemmar S., Zhang J., 2021. Deep Learning-based Fault Detection, Classification, and Locating in Shipboard and Power Systems. 2021 IEEE Electric Ship Technologies Symposium, pp. 1-6, doi: <http://doi.org/10.1109/ESTS49166.2021.9512342>.
- Shamshad, A. Wan, W. Bawadi, M. Sanusi, S., 2005. First and Second Order Markov Chain Models for Synthetic Generation of Wind Speed Time Series. *Energy*, 5 (30), pp. 1-21. doi: <https://doi.org/10.1016/j.energy.2004.05.026>.
- Shang C., Yang F., Huang B., Huang D., 2018. Recursive Slow Feature Analysis for Adaptive Monitoring of Industrial Processes. *IEEE Transactions on Industrial Electronics* 65, pp. 8895-8905, doi: <https://doi.org/10.1109/TIE.2018.2811358>.

- Shi W., Lu N., Jiang B., Zhi Y., Xu Z., 2019. An Unsupervised Anomaly Detection Method Based on Density Peak Clustering for Rail Vehicle Door System. 2019 Chinese Control and Decision Conference (CCDC), pp. 1954-1959, doi: <https://doi.org/10.1109/CCDC.2019.8833427>.
- Shi D., Ma H., He D., Gou Y., 2020. A remaining useful life estimation model of drop system based on data driven and Bayesian theory. Structures 28, pp. 1-8, doi: <https://doi.org/10.1016/j.istruc.2020.09.002>.
- Starczewski A., Krzyżak A., 2015. Performance Evaluation of the Silhouette Index. Artificial Intelligence and Soft Computing. ICAISC 2015. Lecture Notes in Computer Science, vol 9120. Springer, Cham, doi: https://doi.org/10.1007/978-3-319-19369-4_5.
- Sun A. Y., Zhong Z., Jeong H., Yang Q., 2019. Building complex event processing capability for intelligent environmental monitoring. Environmental Modelling & Software 116, pp. 1-6, doi: <https://doi.org/10.1016/j.envsoft.2019.02.015>.
- Taheritanjani S., Schoenfeld R., Bruegge B., 2019. Automatic Damage Detection of Fasteners in Overhaul Processes. 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE), pp. 1289-1295, doi: <https://doi.org/10.1109/COASE.2019.88430497>.
- Tan Y., Tian H., Jiang R., Lin Y., Zhang J., 2020. A comparative investigation of data-driven approaches based on one-class classifiers for condition monitoring

of marine machinery system. *Ocean Engineering* 201, pp. 1-12, doi:
<https://doi.org/10.1016/j.oceaneng.2020.107174>.

Tan Y., Zhang J., Tian H., Jiang D., Guo L., Wang G., Lin Y., 2021. Multi-label classification for simultaneous fault diagnosis of marine machinery: A comparative study. *Ocean Engineering* 239, pp. 1-11, doi:
<https://doi.org/10.1016/j.oceaneng.2021.109723>.

Tan W. M., Teo T. H., 2021b. Remaining Useful Life Prediction Using Temporal Convolution with Attention. *AI* 2, pp. 1-23, doi:
<https://doi.org/10.3390/ai2010005>.

Tang W., Roman D., Dickie R., Robu V., Flynn D., 2020. Prognostics and Health Management for the Optimization of Marine Hybrid Energy Systems. *Energies* 13, pp. 1-29, doi: <https://doi.org/10.3390/en13184676>.

Thetius-Inmarsat, 2021. A changed world. The state of digital transformation in a post COVID-19 maritime industry, pp. 1-52.

Theotokatos G., Stoumpos S., Bolbot V., Boulougouris E., 2020. Simulation-based investigation of a marine dual-fuel engine. *Journal of Marine Engineering & Technology* 19, pp. 1-13, doi: <https://doi.org/10.1080/20464177.2020.1717266>.

Thirukovalluru R., Dixit S., Sevakula R. K., Verma N. K., Salour A., 2016. Generating feature sets for fault diagnosis using denoising stacked auto-encoder. 2016

IEEE International Conference on Prognostics and Health Management (ICPHM), pp. 1-7, doi: <https://doi.org/10.1109/ICPHM.2016.7542865>.

Velasco-Gallego C., Lazakis I., 2020. Real-time data-driven missing data imputation for short-term sensor data of marine systems. A comparative study. *Ocean Engineering* 218, pp. 1-23, doi: <https://doi.org/10.1016/j.oceaneng.2020.108261>.

Xiang S., Qin Y., Zhu C., Wang Y., Chen H., 2020. Long short-term memory neural network with weight amplification and its application into gear remaining useful life prediction. *Engineering Applications of Artificial Intelligence* 91, pp. 1-11, doi: <https://doi.org/10.1016/j.engappai.2020.103587>.

Xiao L., Tang J., Zhang X., Bechhoefer E., Ding S., 2021. Remaining useful life prediction based on intentional noise injection and feature reconstruction. *Reliability Engineering and System Safety* 215, pp. 1-14, doi: <https://doi.org/10.1016/j.res.2021.107871>.

Xu H. Song P., Liu B., 2019. A Vibration Signal Anomaly Detection Method Based on Frequency Component Clustering and Isolated Forest Algorithm. 2019 IEEE 2nd International Conference on Automation, Electronics and Electrical Engineering (AUTEEE), pp. 49-53, doi: <https://doi.org/10.1109/AUTEEE48671.2019.9033363>.

- Xue L., Gao S., 2019. Unsupervised anomaly detection system for railway turnout based on GAN. *Journal of Physics: Conference Series* 1345, pp. 1-5, doi: <https://doi.org/10.1088/1742-6596/1345/3/032069>.
- Yang Y., Liao Y., Meng G., Lee J., 2011. A hybrid feature selection scheme for unsupervised learning and its application in bearing fault diagnosis. *Expert Systems with Applications* 38, pp. 11311-11320, doi: <https://doi.org/10.1016/j.eswa.2011.02.181>.
- Yao Y., Wang J., Xie M., Hu L., Wang J., 2020. A new approach for fault diagnosis with full-scope simulator based on state information imaging in nuclear power plant. *Annals of Nuclear Energy* 141, pp. 1-9, doi: <https://doi.org/10.1016/j.anucene.2019.107274>.
- Yao D., Li B., Liu H., Yang J., Jia L., 2021. Remaining useful life prediction of roller bearings based on improved 1D-CNN and simple recurrent unit. *Measurement* 175, pp. 1-14, doi: <https://doi.org/10.1016/j.measurement.2021.109166>.
- Yuan J., Liu X., 2013. Semi-supervised learning and condition fusion for fault diagnosis. *Mechanical Systems and Signal Processing* 38, pp. 615-627, doi: <https://doi.org/10.1016/j.ymssp.2013.03.008>.
- Wang Z., Oates T., 2015. Encoding Time Series as Images for Visual Inspection and Classification Using Tiled Convolutional Neural Networks. *AAAI Workshop*, pp. 1-7.

- Wang S., Wang J., Wang R., 2020. A novel scheme for intelligent fault diagnosis of marine diesel engine using the multi-information fusion technology. IOP Conference Series Materials Science and Engineering 782, pp. 1-12, doi: <https://doi.org/10.1088/1757-899X/782/3/032022>.
- Wang Q., Zhang G., Huang Y., Zhang Y., 2021. Imaging Multivariate Time-Series to Improve Fault Detection: Application on Tennessee Eastman Process. 2021 7th Annual International Conference on Network and Information Systems for Computers, pp. 196-201, doi: <https://10.1109/ICNISC54316.2021.00045>.
- Wang K., Hu Q., Zhou M., Zun Z., Qian X., 2021b. Multi-aspect applications and development challenges of digital twin-driven management in global smart ports. Case Studies on Transport Policy 9, pp. 1298-1312, doi: <https://doi.org/10.1016/j.cstp.2021.06.014>.
- Wärtsilä, 2020. Combustion engine for power generation: introduction, viewed 27 June 2020, URL <https://www.wartsila.com/energy/learn-more/technical-comparisons/>.
- Zeng F., Li Y., Jiang Y., Song G., 2021. A deep attention residual neural network-based remaining useful life prediction of machinery. Measurement 181, pp. 1-12, doi: <https://doi.org/10.1016/j.measurement.2021.109642>.
- Zhang C., Sun J. H, Tan K. C., 2015. Deep Belief Networks Ensemble with Multi-objective Optimization for Failure Diagnosis. 2015 IEEE International

Conference on Systems, Man, and Cybernetics, pp. 1-6, doi:
<https://doi.org/10.1109/SMC.2015.19>.

Zhang S., Kang R., Lin Y., 2021. Remaining useful life prediction for degradation with recovery phenomenon based on uncertain process. Reliability Engineering and System Safety 208, pp. 1-9, doi: <https://doi.org/10.1016/j.ress.2021.107440>.

Zhang P., Gao Z., Cao L., Dong F., Zou Y., Wang K., Zhang Y., Sun P., 2022. Marine Systems and Equipment Prognostics and Health Management: A Systematic Review from Health Condition Monitoring to Maintenance Strategy. Machines 10, pp. 1-53, doi: <https://doi.org/10.3390/machines10020072>.

Zhao Q., Adeli E., Honnrorat N., Leng T., Pohl K. M., 2019. Variational AutoEncoder For Regression: Application to Brain Aging Analysis. Application to Brain Aging Analysis. Med Image Comput Comput Assist Interv 11765: 823-831, doi: https://doi.org/10.1007/978-3-030-32245-8_91.

Zhao D., Wang T., Chu F., 2019b. Deep convolutional neural network-based planet bearing fault classification. Computers in Industry 107, pp. 59-66, doi: <https://doi.org/10.1016/j.compind.2019.02.001>.

Zhao Z., Cerf S., Birke R., Robu G., Bouchenak S., Mokhtar S. B., Chen L., Y., 2019. Robust Anomaly Detection on Unreliable Data. 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 630-637, doi: <https://doi.org/10.1109/DSN.2019.00068>.

Zio E., 2022. Prognostics and Health Management (PHM): Where are we and where do we (need to) go in theory and practice. Reliability Engineering and System Safety 218, pp. 1-16, doi: <https://doi.org/10.1016/j.ress.2021.108119>.

Appendix A

SWOT Analysis of Maintenance Activities

A.1. Strengths

- Maintenance transition through time from a reactive manner to a more proactive approach.
- A planned maintenance system (PMS) is mandatory according to IMO, which allows shipowner and operator to plan, perform, and document vessel maintenance and intervals complying with Class and manufacturer requirements.
- Test and maintenance are performed according to the guidelines of manufacturers, and Classification Societies (IACS, KR, ABS, DNVGL, LR, BV, and RINA), having the feedback of the shipping company (standards).

A.2. Weaknesses

- Lack of spare parts control and management.
- Paper-based records and visual inspections.
- Scheduled maintenance is often performed too early or too late.

- Infant mortality is sometimes introduced to components in mid-life due to invasive inspection or calendar maintenance.
- CMMS do not perfectly match the particularities of each company. Thus, they prefer to create their own system.
- Maintenance costs represent between 15% and 40% of production costs.
- A lot of time is spent on unnecessary maintenance.
- Downtime is extremely costly.
- Expensive CAPEX and OPEX.
- Training takes significant time and resources.
- CBM technology and PMS are not being utilised by global fleet of vessels.
- Most shipowners do not have the in-house competences or resources to implement an effective condition monitoring system.
- Personnel overburdened with day-to-day operations routine.

A.3. Opportunities

- Industrial IoT technology. It is thought IIoT will increasingly make it easier to drive growth to the company.
- Predictive maintenance and fault correction, which reduce downtime and wastage, leading to positive impact on environmental factors. Prognosis.
- New upgrades in Decision Support Systems (DSS).

- Real-time information utilization, whose targets is to evaluate the state of degraded ship systems and machinery.
- Utilization of Remotely Operated Vehicles (ROV) for inspection (image recorded and processing).
- Information and management of critical components integration. Information Technology (IT) and Operation Technology (OP)
- Forecasting for spare parts demand.
- Maintenance analytics (ML, AI, etc. → New tools + technologies).
- Development of appropriate software and new standards to ensure the systems are interconnected and the data can be handled.
- Comprehensive database can be built if the data can be shared between all the stakeholders.
- Use of mesh networks.
- On-shore support for diagnostics, prognostics, and guidance in emergencies (remote maintenance).
- Development of a robust system/framework for verification, certification, and assurance.
- Development of standards methods for ensuring asset data is in form conducive to ML algorithms.
- Mobile devices utilization and implementation of AR to provide a 3D view of equipment and respective information.
- Dynamic and stochastic scheduling with integration of decision-making tools.

- Application of a balanced performance measurement system including financial, operational, and performance indicators.
- Ordered list based on multi-criteria model to assess equipment criticality.
- Implementation of a multi-criteria technique to classify spare parts. Spare parts management based on reliability knowledge.

A.4. Threats

- Lack of cooperation between stakeholders. There is a lack of trust due to information asymmetries.
- Cost of installation, capital investment, and lack of trust in new technologies.
- Data confidentiality and security.
- Train the staff in new technologies (staff cooperation).
- Massive amounts of data to process and unreliable data.
- The implementation of new technologies requires a strong knowledge base about ship functions, systems, and components.
- Investments in information technology without considering the actual conditions and maintenance strategies.
- Insurance claims → conservative market.
- Lack of adequately trained personnel, which implies greater complexity in new technologies implementation.
- Failures uncertainty. The relative number of random failures is very high.

Appendix B

Literature Review Summary Tables

Table B.1. Literature review summary about data imputation techniques.

Reference	Methodology	Utilisation	Limitations
Pratama et al. 2016	Review study of conventional and modern imputation procedures.	-	-
Chong et al. 2016	Comparative study of five imputation methods: <ul style="list-style-type: none">• Linear regression.• Weighted k-NN.• SVM.• Mean imputation.• Replacing incomplete values with 0.	Data imputation in time-series sensor data.	<ul style="list-style-type: none">• Linear regression is a multivariate imputation method, and thus its accuracy may decrease if the predictors are not highly correlated with the response. Furthermore, the imputation is only accurate if the mentioned correlation is linear.• Weighted k-NN is another example of a multivariate imputation method, and thus its accuracy hinges on the correlation between the predictors and the response. Moreover, the number of neighbours, k, needs to be estimated, and this may lead to

			<p>either under-fitting or over-fitting if k is not optimally selected.</p> <ul style="list-style-type: none"> • SVM also hinges on the correlation between the predictors and the response, its performance may vary based on the kernel selection, and its computational cost is large. • Mean imputation distorts the distribution of the variable and the relationship between variables by reducing estimates of correlation towards zero. • Replacing incomplete values with 0 also distorts the distribution of the variable, which can result in large errors when the incomplete values to impute are far from zero.
Noor et al. 2014	Two imputation methods were implemented: <ul style="list-style-type: none"> • Linear interpolation. • Mean imputation. 	Data imputation in annual hourly monitoring records.	<ul style="list-style-type: none"> • Although linear interpolation was the method that presented the most accurate results, it presents some limitations, such as being inaccurate for non-linear functions. • Mean imputation disrupts the inherent structure of the data and degrades the performance of the statistical modelling, as it can lead to large errors in the matrix correlation.
Balakhirshnan and Sangaiah 2018	Automated framework to impute missing values by applying the context and linear mean (MCL) method	Data imputation in temperature sensor data.	Considering a missing value at time t , this method can only be used if the occurrences at time $t - 1$ and $t + 1$ are available.

Azimi et al. 2019	Multiple imputation approach, which utilises short-term data and context and lifestyle data.	Data imputation in a seven-month monitored data.	The proposed method leads to less accurate results when the incomplete data window is small. Moreover, the bias of the estimates may be large if the correlation between context and missing values is not high.
Priya Stella Mary and Arockiam 2017	Methodology based on the assumption that data collected from sensors presents a highly spatial and temporal correlation.	Data imputation in 5-minute frequency records of air quality sensors data.	The model is only accurate when the relationship between the sensors is strong.
Fekade et al. 2018	Probabilistic method to impute missing values from IoT devices by utilising data from analogous sensors. k -means algorithm is applied to identify neighbours' sensors. Then, Probabilistic Matrix Factorization (PMF) is utilised inside each partition to impute missing values.	Data imputation in data collected from different sensors located in different rooms of a laboratory.	As PMF is utilised, the complexity increases exponentially with increases in the matrix size. Over-fitting may also occur when the technique is trying to minimise an error that results in a loss of generality. In addition, imputations may not be possible to implement if there are not neighbour sensors available.
Bashir and Wei 2018	Method that utilises Vector Autoregressive model (VAR) by combining the Prediction Error Minimisation (PEM) with an EM algorithm. The overall method is named Vector Autoregressive Imputation Method (VAR-IM).	Data imputation in a dataset including electrocardiogram signals of 290 patients.	VAR-IM requires the time series to be stationary. Moreover, its performance may not be more accurate than other data imputation methods analysed when the percentage of incomplete values in the dataset is low.

Izonin et al. 2019	Data imputation method based on the use of the Ito decomposition and the AdaBoost algorithm.	Data imputation in real data recorded using a certified analyser.	The proposed approach yields a lesser performance if the data presents anomalies, or the size of the sample is not adequately large to train the model.
Liu et al. 2020	Univariate data imputation method that utilises STL decomposition, named Itr-MS-STLDecImp.	Data imputation in real-world time series data collected from a Syngas compressor in a real manufacturing plant.	It is only accurate when dealing with large gaps of data and when the time series presents trend and seasonality.
Bokde et al. 2018	Method named imputePSF, which is an adjustment of the Pattern Sequence based Forecasting (PSF) algorithm.	Data imputation in traffic speed time series from a loop detector, in time series of water flow rates generated from hydraulic simulations with EPANET, and nottem dataset that is a twenty-year time series of the monthly average air temperature at Nottingham Castle, England.	It is only accurate when the time series presents periodic components, and thus it is not recommended when the time series presents either noisy trends or non-cyclical patterns.
Hegde et al. 2019	Comparative study of: <ul style="list-style-type: none"> • Probabilistic principal component analysis (PPCA). • Multiple imputation using chained equations (MICE). 	Data imputation in 116 dental variables.	Both techniques are multivariate imputation methods, and thus, if the predictors are not highly correlated with the response, the imputation may not be accurate.
Hadeed et al. (2020)	Comparative study of: <ul style="list-style-type: none"> • Univariate methods (Mean, Median, Last Observation Carried Forward, Kalman Filter, Random, Markov). 	Data imputation in 20 household with complete 24-hour monitoring data for PM _{2.5} .	<ul style="list-style-type: none"> • Univariate imputation techniques may fail to capture expected diurnal or temporal events.

	<ul style="list-style-type: none"> • Multivariate methods (Predictive Mean Matching, Row Mean Method). 		<ul style="list-style-type: none"> • Multivariate imputation may present low performances if there are significant differences between households.
--	---	--	---

Chivers et al. (2020)	A two-step approach (a binary classification step in tandem with a regression analysis).	Data imputation in data from a temperate oceanic climate at sub-hourly temporal resolution.	The comparison between the implemented machine learning models demonstrated their different performances. Ensemble decision tree methods performed well in the classification step, whereas the neural networks performed well in the regression analysis. In no cases k -NN technique was implemented due to the complex and weakly correlated relationship between predictor features and target. Furthermore, the deep learning network of 20 hidden layers was not used either due to over-fitting.
-----------------------	--	---	---

Cheliotis et al. 2019	Hybrid imputation method combining k -NN and MICE algorithms with first-principle knowledge.	Data imputation in 8 sensors coupled to the turbocharger and to the main engine of a chemical tanker.	As mentioned previously along this table, both k -NN and MICE techniques are multivariate imputation methods, and thus, if the predictors are not highly correlated with the response, the imputation may not be accurate. Also, the number of neighbours, k , needs to be estimated, and this may lead to either under-fitting or over-fitting if k is not optimally selected.
-----------------------	--	---	---

Table B.2. Advantages and limitations of the methodologies reviewed in the maritime industry context.

Reference	Methodology	Advantages	Limitations
Brandsæter et al. (2017) and Brandsæter et al. (2019).	Algorithm that implements AAKR as signal reconstruction and SPRT as residual analysis. The algorithm was altered and enhanced by adding three novel modifications: cluster-based memory vector selection method, modified distance measure between the query vector and the memory vector, and credibility estimation.	<ul style="list-style-type: none"> • Robust and fast to implement. • It is effective for the case presented. 	<ul style="list-style-type: none"> • Only one type of fault is presented in the study. • There is no evidence that the framework can be implemented in real time. • The reconstruction method does not consider the characteristics of time series data.
Lazakis et al. (2019).	One-class Support Vector Machine.	<ul style="list-style-type: none"> • It is effective for the case presented. • The methodology can be implemented to a diverse set of machinery. 	<ul style="list-style-type: none"> • There is no evidence that the framework can be used in real time. • It does not consider the characteristics of time-series data. • Inconvenient for large sets of data.
Cheng X. et al. (2019b).	FPGA.	<ul style="list-style-type: none"> • It is effective to denoise vibration signals that contain strong noise. • It can be applied in real time. 	<ul style="list-style-type: none"> • Its application is specific, and thus cannot be considered as a generic anomaly detection framework.
Ellefsen et al. (2019) b.	VAE.	<ul style="list-style-type: none"> • SA can be implemented as a feature selection to remove irrelevant and redundant input features. 	<ul style="list-style-type: none"> • Similar to Ellefsen et al. (2020), there is no evidence that the algorithm can be automatically implemented in real time.

Ellefsen et al. VAE.
(2020).

Cheliotis et al. Combination between EB models and
(2020b). EWMA.

- The algorithm achieved an average accuracy of 97.88% in the experiment presented, which indicates the accuracy of implementing VAE in this context.
 - The algorithm can be used to detect degradations of different natures.
 - It is possible to derive the reconstruction of the data to analyse the underlying cause of the fault to apply isolation.
 - It is highly transferable and can be applied in a variety of different cases.
 - The framework demonstrated to be effective for detecting the faults that were simulated.
 - The framework does not face the challenges of using real data.
 - ASF may lead to inaccuracies due to the different engine loads.
 - Only one fault type is considered.
 - The framework cannot be applied automatically in real time.
 - The framework does not face the challenges of using real data.
 - There is no evidence this framework can be applied in real time.
 - It does not consider the characteristics of time-series data.
 - Other machine learning and time series forecasting models may be more accurate than the models selected for this study.
-

Table B.3. Summary of fault identification of marine systems studies.

Reference	Methodology	Case Study	Main Results
Wang et al. (2020)	A supervised phase was introduced in the fault diagnosis framework for the identification of the running state of the system. This supervised phase implemented a Back Propagation Neural Network (BPNN)	The proposed methodology was validated by considering a marine diesel engine. The data considered ran under five common working conditions: 1) normal, 2) carbon deposition of the injector nozzle, 3) air leakage of the exhaust valve, 4) wear of high-pressure oil pump, and 5) damage of the piston ring. Regarding the parameters, a total of eight were considered (e.g., mean effective pressure, scavenging pressure, and rotation speed)	High accuracy was perceived in identifying both working and high-pressure oil pump wear exhaust valve leakage conditions. However, it was considered that the diagnostics of both the nozzle carbon deposition and piston ring damage conditions required an enhancement
Cai et al. (2017)	A total of three steps were presented. The first step referred to the structuring of the system into subsystems to reduce complexity in the analysis. Subsequently, a classification model based on SVM was applied to perform the monitoring of	The diesel engine system was considered. As subsystems, the 1) fuel, 2) lubrication, 3) intake and exhaust, and 4) cooling subsystems were identified. A historical fault database was available for validation purposes. For instance, the fuel system	An accuracy of 96% was obtained when the diesel engine fuel system fault dataset was considered

	operating states and fault diagnosis. The third and final step referred to the application of the association rule mining algorithm to analyse the relationships among the fault characteristics at distinct levels	fault dataset contained a total of 75 groups of prediction data in 5 working conditions with 15 groups of prediction data of each working state	
Hou et al. (2020)	An optimisation of the SVM algorithm with a three-dimensional Arnold mapping introduced into the Particle Swarm Optimization algorithm was considered. The Principal Component Analysis (PCA) method was also implemented prior to the implementation of the model to reduce the computational complexity	The fuel oil supply system of a marine diesel engine was considered. A total of ten faults were available for validation purposes	A comparative study between the established methodology and the Multi-Layer Perceptron (MLP) with BP algorithm added momentum factor and adaptive learning, and Radial Basis Function Network was performed. The established methodology outperformed the remaining methods in terms of Correct Diagnosis Ratio. However, in terms of execution time, it was perceived a significant difference between the established method, which achieved an execution time of 78.27 seconds, and the remaining analysed methods, which presented 0.98 and 0.53 seconds respectively

Senemmar and Zhang (2021)	Three methodologies were presented: 1) deep neural network, 2) Gated Recurrent Unit (GRU), and 3) Long Short-Term Memory (LSTM)	Fault data from an 8-bus shipboard power system were simulated	The GRU-based model was considered as the most effective DL model, as it achieved an accuracy of 99%
Tan et al. (2020)	A total of 6 one-class classifiers were investigated: 1) One Class Support Vector Machine (OCSVM), 2) Support Vector Data Description (SVDD), 3) Global k -Nearest Neighbors (GKNN), 4) Local Outlier Factor (LOF), 5) Isolation Forest (IF), and 6) Angle-Based Outlier Detection (ABOD)	A real-data validated numerical simulator developed for a Frigate characterized by a combined diesel-electric and gas propulsion plat was considered	The performance of the six analysed methods were ranked as follows: ABOD > OCSVM \approx SVDD > GKNN > IF \approx LOF
Tan et al. (2021)	A total of five multi-class classifiers were considered for simultaneous fault diagnosis: 1) Binary Relevance (BR), 2) Classifier Chains (CC), 3) Multi-Label k -Nearest Neighbor (MLKNN), 4) Binary Relevance k -Nearest Neighbor (BRKNN), 5) Multi-Label Twin Support Vector Machine (MLTSVM)	Analogous to Tan et al. (2020), a dataset generated from a data simulator of a Frigate was considered	The performance of the five analysed methods were ranked as follows: BR > CC > BRKNN > MLKNN > MLTSVM

Table B.4. Summary of time series imaging approaches in fault identification task studies.

Reference	Methodology	Time Imaging Approach	Series	Case Study	Sector/Domain of Application
Fahim et al. (2021a)	Self-attentive weight-sharing capsule network	GAF		A Western-System-Coordinating Council WSCC 9-bus and 3-machine test model modified with the series capacitor was considered	Transmission line domain
Fahim et al. (2021b)	Stacked denoising-autoencoder	GAF		Time-series oscillographs data	Transmission line domain
Fahim et al. (2020a)	self-attention convolutional neural network	GAF		228,690 data samples of both three-phase voltage and current signals	Transmission line domain
Fahim et al. (2020b)	self-attention convolutional neural network	GAF		Input signals namely voltage, current and combined voltage and current signal, under various sampling frequencies	Transmission line domain
Yao et al. (2020)	The image feature was extracted by implementing Kernel Principal Component Analysis (KPCA). Then, the	Full-scope Simulator based on the State		Nuclear plant	Nuclear
Appendix B		344			Christian Velasco-Gallego

resulting image features were classified by analysing five Information Imaging
typical classifiers: 1) SVM, 2) k -NN, 3) Linear (FSFSSII)
Discriminant Analysis, 4) Decision Tree Analysis, and 5)
logistic regression

Appendix C

Bibliometric and Taxonomy Results of Anomaly Detection Studies

C.1. Bibliometric Analysis

The implementation of anomaly detection methodologies can enhance operations and maintenance activities in numerous sectors, such as manufacturing, construction, energy, and maritime, by predicting, preventing, and detecting machine failures that may result in a loss of productivity, thus increasing both costs and unplanned downtime. It is unsurprising, therefore, that there are more than 41,000 results when searching papers about anomaly detection in Scopus database. Trends indicate that academic publications in relation to this concept are increasing expeditiously, which makes the application of regular literature reviews unfeasible. Accordingly, a bibliometric analysis is performed to extract knowledge to be considered for further development of anomaly detection techniques applied for maintenance analytics. Specifically, due to the complexity of obtaining fault data of marine systems, the application of either unsupervised or semi-supervised is the only further analysed at this stage. To that end, the following query was used to obtain a first estimation of papers to be included in the analysis:

(“anomaly detection” OR “outlier detection” OR
“novelty detection” OR “fault diagnosis” OR
“fault detection”) AND (“maintenance”) AND
(“unsupervised” OR “semi-supervised”)

A total of 141 publications are identified from Scopus database. Only papers published from 2010 to 2020 are considered to limit the analysis to the most recent studies. Of all the categories of document types, only those types that refer to novel anomaly detection approaches are considered (conference papers, articles, conference reviews, reviews, and book chapters). Therefore, after applying this initial filter, 131 out of 141 are analysed further. Prior to the data analysis step, pre-processing is performed to remove redundant articles that are either out of the scope or are duplicated (e.g., a study has been presented in an analogous manner in a conference paper and in an article). After completing this pre-processing step, a total of 90 out of 131 publications are objects of study.

Table C.1. The 10 most cited articles in the anomaly detection for maintenance analytics practices field.

Ranks	References	DOI	TC	Source
1	Langone et al. (2015)	10.1016/j.engappai.2014.09.008	61	Eng Appl Artif Intell
2	Yang et al. (2011)	10.1016/j.eswa.2011.02.181	52	Expert Sys Appl
3	Thirukovalluru et al. (2016)	10.1109/ICPHM.2016.7542865	48	IEEE Int Conf Progn Health Manag, ICPHM

4	Helbing and Ritter (2018)	10.1016/j.rser.2018.09.012	40	Renewable Sustainable Energy Rev IEEE Trans Ind Electron
5	Shang et al. (2018) Di Maio et al.	10.1109/TIE.2018.2811358	40	
6	(2012)	10.1016/j.eswa.2011.10.008	38	Expert Sys Appl Int Conf Ind Eng Appl, ICIEA
7	Amruthnath and Gupta (2018)	10.1109/IEA.2018.8387124	34	
8	Harrou et al. (2019)	10.1016/j.solener.2018.12.045	26	Sol Energy IEEE Int Conf Syst, Man, Cybern, SMC
9	Zhang et al. (2015) Yuan and Liu	10.1109/SMC.2015.19	25	
10	(2013)	10.1016/j.ymssp.2013.03.008	25	Mech Syst Signal Process

Table C.1 describes the 10 most cited articles in the anomaly detection for maintenance analytics practices field. The results were analysed based on the Total Citations (TC) metric. Langone et al. (2015) demonstrated the applicability of implementing Least Squares Support Vector Machines (LS-SVMs) for fault diagnosis. Prior to the implementation of LS-SVMs, Kernel Spectral Clustering (KSC) was performed as an unsupervised approach on sensor data coming from a vertical seal and fill machine to identify both normal operating conditions and abnormal contexts. Subsequently, a Nonlinear Auto-Regressive (NAR) model was illustrated in the LS-SVM framework. Results demonstrated the capability of the presented framework to identify degradations affecting the machine. Yang et al. (2011) presented a hybrid feature selection scheme for unsupervised learning in fault diagnosis. A bearing fault diagnosis application was utilised to highlight the robustness and the accuracy of the presented approach. Additionally, results were compared with widely implemented

feature selection algorithms, such as PCA-based feature selection and forward search feature selection. Thirukovalluru et al. (2016) examined traditional handcrafted features and compared them with features learned by Deep Neural Networks (DNNs) for implementing fault diagnosis. To perform this analysis a total of five datasets were analysed, which referred to air compressor monitoring, drill bit monitoring, steel plate monitoring, and two bearing fault monitoring data. Results demonstrated both good feature representation and accurate classification performance when utilising DNNs. Helbing and Ritter (2018) discussed recent applications of Artificial Neural Networks (ANNs) and DL approaches in the wind turbines sector. Results indicated the prevalence of unsupervised methodologies within this sector, although supervised methodologies demonstrated promising results, due to the challenges that need to be addressed in relation to quality and accessibility as well as labelling and class imbalance of operational data. Shang et al. (2018) developed a recursive slow feature analysis as a new process monitoring and fault diagnosis approach. A case study on a real crude heating furnace system was implemented to demonstrate the efficacy of the presented methodology. Di Maio et al. (2012) compared two unsupervised ensemble methods (fuzzy C-means and hierarchical trees). Predictions of multiple classifiers were combined to reduce variance of both results and bias. Data collected from several slurry pumps were utilised to determine the effectiveness of the developed model to identify the health status. Amruthnath and Gupta (2018) tested the accuracy, performance, and robustness of a total of 5 unsupervised learning algorithms (PCA T2 statistic, Hierarchical clustering, K-Means, Fuzzy C-Means clustering, and model-

based clustering). Simple vibration data collected from an exhaust fan was utilised to analyse the implemented algorithms. Harrou et al. (2019) introduced a fault diagnosis approach for monitoring photovoltaic systems. An anomaly detection approach was developed by implementing a model based on the one-diode model to mimic the characteristics of the photovoltaic array and, subsequently, apply a OCSVM to residuals from the simulation model to detect faults. Real data from a 9.54 kWp grid-connected plant was implemented as a case study to highlight the superior performance of the proposed approach in relation to other five binary clustering schemes analysed in the study. Zhang et al. (2016) analysed Deep Learning Network (DBN) to perform classification. Precisely, an ensemble of DBNs with Multi-Object Evolutionary Algorithm based on Decomposition (MOEA/D) is implemented to detect failure degradation. A turbofan engine degradation simulation dataset provided from NASA was utilised for analysis. Yuan and Liu (2013) introduced manifold regularization based semi-supervised learning to implement fault diagnosis. Two vibration signals of Buma pump and CWRU bearing datasets were utilised to test the proposed methodology. Results demonstrated the feasibility and efficiency of the approach.

The analysis of the most cited articles demonstrated that there is no clear algorithm that outperforms all possible scenarios that can be considered in relation to the system and the type of fault being analysed. DL methodologies are gaining attention within recent years, thus indicating their applicability as anomaly detection approaches.

However, there are various challenges that are yet to be tackled in relation to these algorithms, an example of which is the lack of trust in "black-box" models by the industry. Although known for their undeniable lack of transparency, there is no doubt whatsoever about their extraordinarily accurate predictions. Therefore, the incorporation of explainable artificial intelligence models is of paramount importance to guarantee the enhancement of operations and maintenance activities while ensuring transparency, interpretability, and explainability. Another relevant challenge is the lack of available data, as industry is exceedingly reserved due to the sensitive information that can be extracted. Therefore, some data, such as fault data, are deeply laborious to obtain, significantly slowing the research process. Consequently, the cooperation between industry and academia is of preeminent importance to make data available, thus facilitating the implementation of DL methodologies, which require a large amount of data to train and optimally select their respective parameters. In addition, as Internet of Ships is in its infant phase, there is a lack of data quality due to unreliable outcomes caused by certain anomalies and missing values that are originating from device failure, network collapse, and human error (Balakrishnan and Sangaiah, 2018; Izonin et al., 2019; Noor et al., 2014). Accordingly, the adequate implementation of data pre-processing steps, such as data imputation, is essential to guarantee reliable data-driven models. Although data imputation is a compelling pre-processing step that has gained popularity recently, there is a lack of formalisation and analysis, thus far, within the shipping industry. In accordance with this aspect,

the deployment of these novel models within the shipping industry is also yet to be adequately formalised.

Aligned with the analysis performed in the preceding paragraphs, a co-occurrence analysis of keywords is also implemented. From Fig. C.1, some fundamental aspects already outlined in the preceding paragraphs are identified. Regarding the methodologies implemented, OCSVM, Decision Trees, Neural Networks, Auto Encoders, Feature Extraction, and clustering algorithms were the most applied within the studies. Specially, deep learning and support vector machines prevail among other unsupervised and semi-supervised algorithms. Bearings' datasets, Structural Health Monitoring (SHM) data, industrial and manufacturing equipment sensor data are the major typologies of datasets in which state-of-the-art anomaly detection approaches have been modelled for fault diagnosis. To complement the graphical representation described in Fig. C.1, the 20 most frequent keywords are listed in Table C.2.

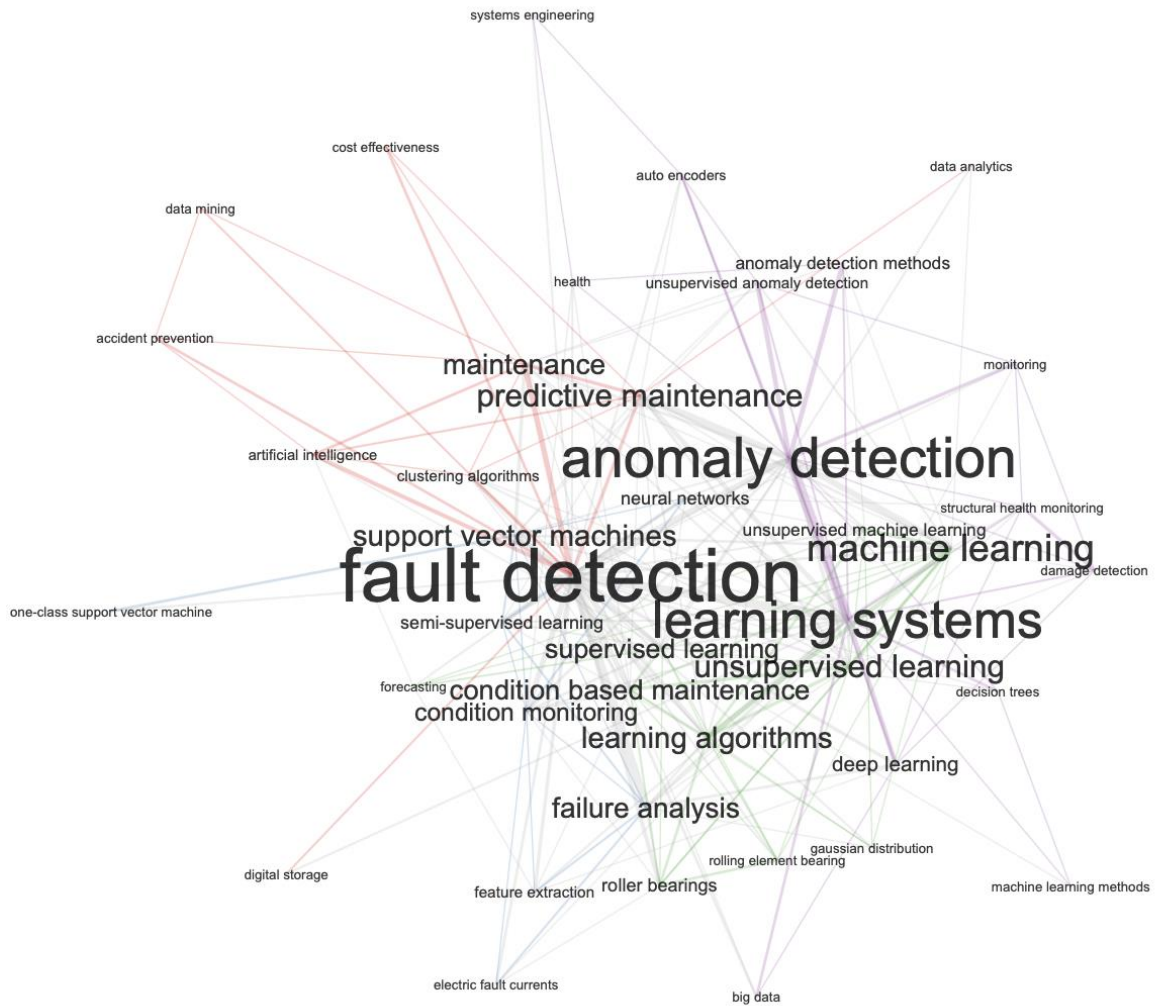


Fig. C.1. Co-occurrence of keywords.

Table C.2. Most frequent keywords.

Author's keywords		Keywords Plus	
Words	Freq.	Words	Freq.
anomaly detection	23	fault detection	45
fault detection	16	anomaly detection	33
machine learning	14	learning systems	26
predictive maintenance	13	machine learning	18
unsupervised learning	12	predictive maintenance	15
fault diagnosis	9	failure analysis	14
condition-based maintenance	6	maintenance	14
condition monitoring	5	unsupervised learning	14
structural health monitoring	5	support vector machines	12
damage detection	4	condition based maintenance	11

deep learning	4	condition monitoring	11
semi-supervised learning	4	deep learning	11
gaussian mixture model	3	learning algorithms	11
isolation forest	3	supervised learning	11
supervised learning	3	anomaly detection methods	8

To finalise the bibliometric analysis, the frequencies of the algorithms presented throughout the years are examined based on the indicator Keyword Plus to determine past and present trends, and thus establish future perspectives in relation to the application of unsupervised and semi-supervised anomaly detection techniques for fault diagnosis. As identified in Fig C.2. and stated previously, the number of publications has increased significantly. Therefore, the number of algorithms implemented in papers published in 2010-2015 is not substantial. SVM and ANN are highlighted within this period. Some studies related to feature extraction were also identified, although this concept has been further investigated in the recent years. Deep learning is unquestionably the area that is gaining the most attention in recent years and the number of publications that consider such approaches are increasing expeditiously. Auto Encoders (AE), adversarial networks, and Convolutional Neural Networks (CNNs) are just some examples of deep learning models that have been analysed in academia to develop fault diagnosis frameworks.

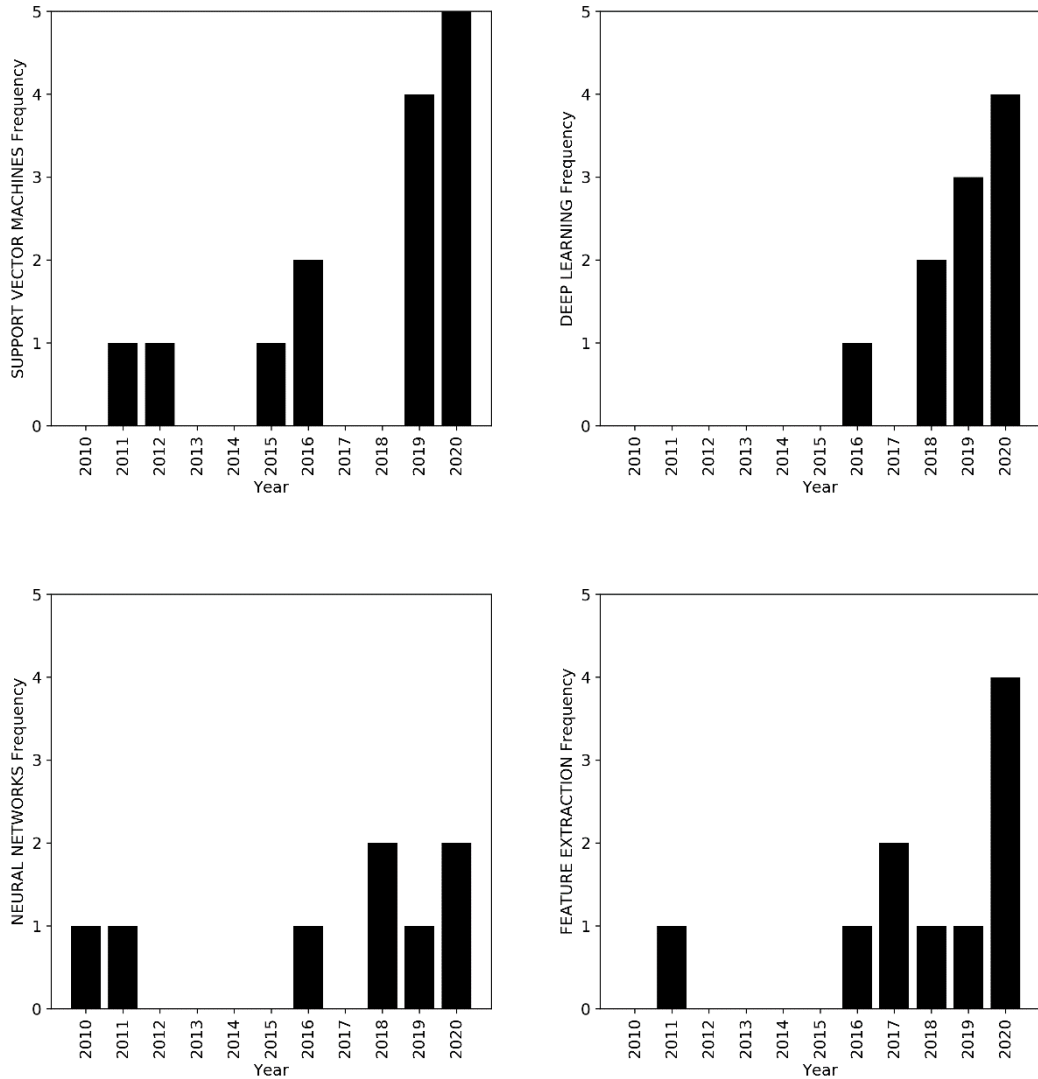


Fig. C.2. Frequencies of the most implemented algorithms throughout the years (2010-2020) based on the Keyword Plus indicator.

C.2. Taxonomy of Anomaly Detection Studies

To complement the bibliometric analysis about anomaly detection within the maintenance analytics context, and the critical literature review about unsupervised and semi-supervised anomaly detection techniques in the maritime industry, a taxonomy of anomaly detection papers is presented in this section. Taxonomy

analysis has been widely implemented within the maritime industry to review diverse concepts. An example of which is the review presented in Zis et al. (2020), which performed a critical literature review on weather routing and voyage optimisation problems. The review included an analysis of the main methodologies implemented for weather routing and a taxonomy of research based on relevant academic papers.

By prioritising recent and most cited publications, a total of 45 anomaly detection papers within the maintenance analytics context are classified according to the parameters listed hereunder.

- **Applied methodology.** The methodology presented in the paper is briefly described.
- **Implementation of case studies.** If a case study is performed to highlight the presented methodology, it is shortly characterised.
- **Type of data utilised.** The characteristics of the data are indicated.
- **Nature of the faults.** The faults utilised to evaluate the presented methodology are outlined.
- **Methodology deployed in real time.** The possibility of deploying the presented methodology in real time is detailed.
- **Industrial sector.** As anomaly detection is an interdisciplinary concept, the industrial sector in which the method is applied is specified.

Due to the dimensions of the resulting table, the detailed results of this taxonomy are represented in Table C.3. However, a summary based on the resulting taxonomy is expressed in the subsequent paragraphs.

Firstly, if the applied methodologies aspect is considered, DL models lead the ranking, as they have been applied in more than 30 case studies. This reaffirms the conclusions outlined from the bibliometric analysis, in which it could be observed that the current and future trends regarding anomaly detection approaches were constituted by DL algorithms. Autoencoders and LSTM networks are examples of the most common models implemented in the analysed studies. Clustering algorithms, such as k -means and GMM, were also widely applied within this context. Regarding supervised classifiers, Isolation Forest (IF) was the most implemented algorithm due to its effectiveness to be applied in real time, as it requires less memory than other ML and DL algorithms. For this reason, in most of the cases, in which this model was applied in predictive frameworks, its deployment was in real time. However, as previously stated, this is a challenge that needs to be addressed when considering DL algorithms as anomaly detection techniques. Although most of the studies highlighted the outstanding accuracy of DL models to detect anomalies, many of them were not holistic, and thus their unfeasible application in real-world scenarios was promoted. This is because diverse studies utilised simulated data to assess the performance of the methodologies, as the collection of the data needed to adequately train and test the models is extremely complicated. Therefore, the challenges that need to be

addressed when dealing with real-world scenarios were not considered accordingly. All the following are indispensable examples to be further analysed due to the consequences that they can have in the performance of the DL methodologies when are applied in real-world scenarios:

- The lack of fault data.
- The need of explainable models due to the lack of transparency of DL algorithms.
- The quality of the dataset in relation to corrupted values due to either sensors' failure or human error, outliers, and high noise when dealing with sensor data.
- The implementation of data imputation due to the large number of missing values that datasets contain.
- The lack of synchronisation between sensors.
- The necessity of an integration framework due to the collection of data from various sources.

In this sense, a more comprehensive description of all the prior and subsequent phases implemented to obtain the expected model performance need to be outlined in the studies presented. Therefore, it is required to promote transparency of the best practices being applied and provide a holistic predictive framework to be utilised by the industry.

If the industrial sector is considered, the manufacturing and industry is the one leading the list (see Fig. C.3). This is because this sector encompasses those studies that refer to machinery systems, such as gas turbines and robot arms, which require zero downtime during their operational state to guarantee the expected production/functioning; reducing the risks and costs associated with downtime and systems' failure. Correspondingly, sectors such as aircraft and space, railway, and wind turbines are also leading the list on account of this fact.

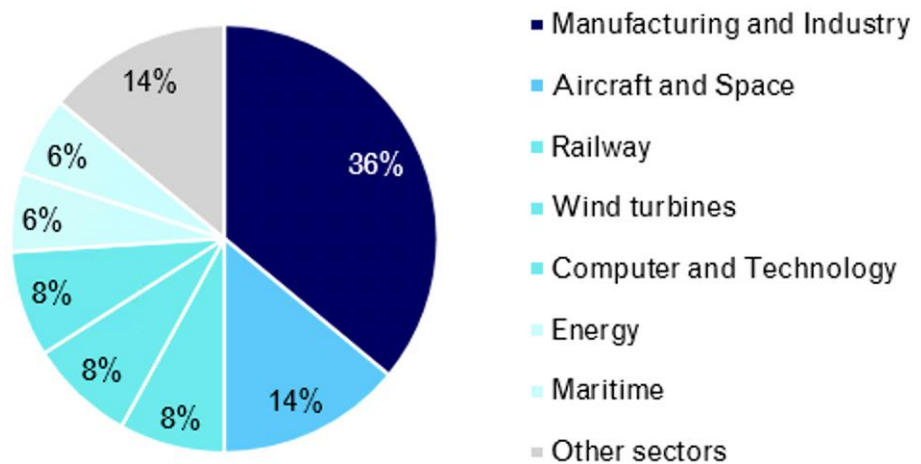


Fig. C.3. Studies classified by industrial sector.

Three articles were identified within the maritime sector, one of which referred to Structural Health Monitoring (SHM) and two of them were related to marine machinery, the latter being within the scope of the studies analysed in the critical

literature review about unsupervised and semi-supervised anomaly detection techniques within the maritime industry. However, these were not assessed as they were not retrieved from the query implemented. This consideration is of paramount importance, as it indicates a limitation to query-based analysis and the influence of the selection of keywords and abstract development to avoid biased results. For example, Carrega et al. (2019) presented a generic title and abstract focusing on the technique implemented and, in failing to describe comprehensively the case study to highlight the proposed methodology, the article was subsequently omitted from the analysis when the query was applied. In essence, as the number of publications is increasing expeditiously and the implementation of regular literature reviews is becoming unfeasible, which yields the implementation of data-driven review approaches, the selection of keywords and refining of both the title and the abstract are critical to be in accordance with the desired scope and target audience.

Table C. 3. Taxonomy of anomaly detection studies.

Reference	Applied methodology	Implementation of case studies	Type of data utilised	Nature of the faults	Methodology deployed in real time	Industrial sector
Ahmed et al. (2019)	Isolation Forest.	Anomaly detection in DNS queries.	Benign dataset of top rank primary domains from two enterprise networks.	Simulation of malicious DNS queries.	The methodology can be deployed in real time.	Computer and technology.
Alaoui-Belghiti et al. (2019)	Acoustic signals were compared with reference signals through the calculation of Sinkhorn distances.	Industrial bench test.	Noise of an industrial test.	Two faulty mechanical parts were considered: the sound of a light and a cyclical low-pitched sound.	There is no evidence that the proposed methodology can be implemented in real time.	Manufacturing and Industry.
Borith (2020)	Decision tree, k -NN, random forest, and linear SVM.	Automobile part manufacturer.	17 features and 86,400 from an automobile part manufacturer.	Dataset divided into active and non-active states.	The proposed methodology can be deployed in real time.	Manufacturing and Industry.
Bose et al. (2019)	ADEPOS framework that applied ELM-B as the machine learning algorithm.	Experiments on bearings.	NASA bearing dataset.	Faults in bearings.	There is no evidence that the proposed methodology can	Aircraft and Space.

					be implemented in real time.	
Carrega et al. (2019)	Multi-purpose algorithm for unsupervised or semi-supervised learning to detect optimal operating regions.	Frigate.	Sensor data collected on a frigate, characterised by a combined diesel, electric, and gas naval propulsion system.	The decay and speed states were simulated.	There is no evidence that the presented methodology can be deployed in real time.	Maritime.
Chen et al. (2020)	Sliding-Window Convolutional Variational Autoencoder (SWCVAE).	Industrial robot use-case.	Normal operation time series data.	Only one anomaly scenario was presented in the case study.	The proposed methodology is partially online, as the process of training model is offline.	Manufacturing and Industry.
Coraddu et al. (2019)	Two anomaly detection methods based on Support Vector Machines (SVMs) and k -Nearest Neighbour (k -NN).	Hull condition of the Research Vessel The Princess Royal.	Real time data collected when the vessel was in operation.	Two different hull conditions of the Princess Royal were considered: clean and fouled.	The proposed models can be adopted for real-time applications directly onboard.	Maritime.
Cui et al. (2018)	Nonlinear Autoregressive Neural Networks	2MW wind turbine located in Sweden.	Data collected from the wind turbine.	Failure in the gearbox bearing.	There is no evidence that the proposed methodology can	WInd turbines.

	with eXogenous inputs (NARX).				be deployed in real time.	
Doğru et al. (2020)	Mask Region Convolutional Neural Networks (MASK R-CNN).	Aircraft.	Images of aircraft dents from different sources.	Aircraft dents.	There is no evidence that the proposed framework can be deployed in real time.	Aircraft and Space.
Ducharlet et al. (2020)	SuMeRI approach, which applied successive specialised anomaly detection methods in an iterative way. One-Class SVM, Linear Regression, and Nerual Networks were considered for the case presented.	Utilisation of two public datasets and a real case study dataset.	The first dataset referred to Numenata Anomaly Benchmark and it measured the temperature on a machine. The second contained the HTTP requests in the KDD Cup 1999 dataset. The real case dataset referred to temperature measurements in a building that was equipped with an intelligent auto-regulating system.	System failures were presented in the first dataset. In the second dataset the data were labelled as either normal or an attack, in which the type of attack was also indicated. In the real case the dataset the data were unlabelled, although it was known that the dataset contained anomalies.	There is no evidence that the proposed methodology can be implemented in real time.	Manufacturing and Industry.

Farbiz et al. (2020)	Cognitive analytics-based framework.	Industrial robot use-case.	Data obtained from robot controller data, energy data, and vibration sensor data.	All data collected referred to normal operation data.	Although the methodology indicates that it can be deployed in real time, further studies are required to assess the accuracy of the prediction of anomalies, as no fault data was utilised to evaluate the model in the case study.	Manufacturing and Industry.
Fernandes et al. (2019)	Autoregressive Integrated Moving Average (ARIMA) models.	CNC machines from a mechanical metallurgy factory.	Data collected from two lathes and two vertical mills by installing sensors and interfacing with the machines' firmware to consume the information provided by the inner sensors.	All data referred to normal operation data, as CNC machines were very rarely inclined to fail due to a strict schedule of preventive maintenance.	Although it is not specifically indicated in the paper, the proposed methodology can be deployed in real time with some adjustments.	Manufacturing and Industry.

Hodžić et al. (2020)	Principal Component Analysis (PCA) and binary classification decision tree algorithm were utilised as unsupervised anomaly detection methods. Supervised learning is not within the scope of this analysis	Railway propulsion system.	Data collected from the Real-Time Simulator (RTS) lab, which included sensors for measuring current, speed, voltage, reference power, and two different temperatures.	Data collected were not labelled, and thus the nature of faults was unknown. Unsupervised anomaly detection was performed in tandem with domain experts analysis to label the data and train the supervised learning model.	There is no evidence that the methodology can be deployed in real time, as domain experts needed to analyse the data and appropriately label it while performing the unsupervised learning method.	Railway.
Hsu et al. (2019)	Density-based spatial clustering of applications with noise algorithm was used to classify abnormal-state wind turbine data from normal-state data. Subsequently, random forest and decision tree techniques were implemented to	Wind turbines located in Taiwan.	Sensor data.	Control charts were used to detect four categories of wind turbine faults: rotary blades, gearboxes, generators, and hydraulic oil systems.	There is no evidence that the proposed methodology can be deployed in real time.	Wind turbines.

construct the predictive models for wind turbines anomalies.

Imbassahy (2020)	Generic Anomaly Detection Hybridization Algorithm (GADHA).	Aircraft.	Data collected from landing gear.	Abnormal situations typically found in landing-gears operations.	There is no evidence that the proposed framework can be deployed in real time.	Aircraft and Space.
Inagaki et al. (2021)	Two methods were evaluated (autoencoder and Gaussian Mixture Model (GMM)).	Press machine.	Data collected from the press machine while operating.	A sudden anomaly was identified in the test dataset. Consequently, the machine stopped.	There is no evidence that the proposed methodology can be deployed in real time.	Manufacturing and Industry.
Kolokas et al. (2019)	Isolation Forest (IF) and Elliptic Envelope (EE)	Anode production.	Data that included variables, such as temperatures, speeds, and flow rates collected from the cooler.	1052 fault incidents were registered and utilised to assess the accuracy of the	There is no evidence that the presented methodology can	Manufacturing and Industry.

				proposed methodology.	be deployed in real time.	
Kolokas et al. (2020)	Isolation Forest (IF)	Data related to aluminium and plastic production.	The data were divided into two general categories: the input data, which referred to analysed signals where anomalies were detected, and the target data, which referred to stop data.	A total of 1870 stops were identified, which could be either voluntary (e.g. maintenance activities) or involuntary (faults or breakdowns). Some of these stops referred to machine breakdown, mold breakdown, lack of cooling, lack of raw material, and auxiliary equipment failure.	It can be deployed in real time, although the model may need to be retrained due to the changing process conditions.	Manufacturing and Industry.
Kyriakou et al. 2019	Artificial Neural Network (ANN)	Detection of potholes.	Sensor data were collected for 31 parameters, such as speed, accelerations, and rotations.	Roadway pothole anomalies at known locations were used to train the model.	There is no evidence that the proposed methodology can be implemented in real time.	Civil.

Li et al. (2019)	Deep Belief Network.	Spacecraft storage batteries.	Data originated from satellite X.	Decreased in the voltage of the storage battery of satellite X in the discharge mode.	The proposed methodology can be implemented in real time.	Aircraft and Space.
Li et al. (2020)	Attn-based Bi-LSTM	Log-based anomaly detection.	Log datasets ranging from distributed systems, supercomputers, operating systems, mobile systems, etc.	Sequence order changes, which referred to sequential log anomalies, and time interval changes, which referred to performance issues.	The proposed methodology can work in a nearly real-time mode.	Computer and Technology.
Liu and Gryllias (2020)	Support Vector Data Description (SVDD) with Negatives Samples (NSVDD).	Experiments on bearings.	NASA Intelligent Maintenance Systems (IMS) dataset.	Faults in bearings.	There is no evidence that the proposed methodology can be implemented in real time.	Aircraft and Space.
Liu et al. (2018)	Stacked AutoEncoders (SAEs)	Gearbox fault diagnosis.	Dataset based on vibration signals collected on a specially designed bench which consisted of a one phase input and three-phase output motor, a gearbox, the shaft	Multifault gearbox experimental dataset.	There is no evidence that the proposed framework can be deployed in real time.	Manufacturing and Industry.

supporting seats, a flexible coupling, and a magnetic powder brake.

Liu et al. (2020c)	Evaluation of regression methods (e.g. linear regression, lasso regression, and ridge regression) and neural networks (ANN, LSTM, and bi-LSTM).	Vertical plant wall placed in an elderly home located in Sweden.	CO2 concentration level and temperature variations in a green room lab data.	Some point anomalies were identified due to noise. Two anomalous test sets were manually generated for point anomaly and contextual anomaly scenarios.	The proposed methodology can be implemented in real time.	Building.
Lutz et al. (2020)	Autoencoders.	Wind turbines located in the same wind farm in the German North Sea.	Operational and event data.	Scada-events provided the information if the wind turbine was in downtime, although it did not specify why the downtime occurred.	There is no evidence that the proposed framework can be deployed in real time.	Wind turbines.

Madakyaru et al. (2019)	Integrated statistical mechanism merging the benefits of Partial Least Squares (PLS), Adaptive Neural Network Fuzzy Inference Systems (ANFIS) modeling and the k -Nearest Neighbors (k -NN)-based data mining scheme for nonlinear process monitoring.	Laboratory scale bubble cap distillation column.	Distillation data temperatures, feed, and coolant fluid temperatures.	The model was tested by perturbing the column around its nominal operating condition. The feed flow was perturbed with the magnitude of ± 50 while keeping the reflux flow constantly at the nominal condition. Another anomaly was introduced by perturbing the reflux flow with the magnitude of ± 40 while keeping the flow rate constant.	There is no evidence that the proposed methodology can be implemented in real time.	Chemical.
Makridis et al. (2020)	XGBoost, LSTM, One Class SVM, and Permutation Entropy Check to perform regression.	Vessels.	Sensor data collected from various vessels.	Data collected from defected ships.	There is no evidence that the methodology can be implemented in real time.	Maritime.

Martin-del-Campo et al. (2019)	Convolutional sparse coding with dictionary learning.	Two case studies were presented. Case study 1: controlled experiment with the data taken from the bearing data center at Case Western Reserve University. Case study 2: data from a real-world condition monitoring system. The data was collected from the monitoring system installed in wind turbines located in northern Sweden.	The first case study considered data collected from a well-known controlled experiment. This experiment consisted of a motor, a torque transducer, and a dynamometer. The second case study utilised data recorded from the high-speed shaft of a wind turbine.	The first dataset presented faults seeded in the bearings. Analogously, the second dataset also presented a bearing failure.	There is no evidence that the proposed methodology can be deployed in real time.	Wind turbines.
Miki and Demachi (2020)	Long Short-Term Memory Neural Network.	Experiments on bearings.	Dataset published by the Bearing Data Center of Case Western Reserve University (CWRU).	Failures such as ball failure, inner-ring failure, and outer-ring failure of various defect sizes.	There is no evidence that the proposed framework can be deployed in real time.	Nuclear.
Morariu et al. (2020)	Long Short-Term Memory (LSTM) Neural Networks (NNs).	Pick and place operation (basic operation performed by industrial robots and used in assembly).	Set of pick and place robot operations, measured at two different motion speed settings.	Abnormal operations characterised by either higher energy consumption or	The methodology can be implemented in real time.	Manufacturing and Industry.

					longer execution time.	
Mulongo et al. (2019)	Support Vector Machines (SVMs), k -Nearest Neighbors (k -NN), Logistic Regression (LR), and MultiLayer Perceptron (MLP).	Fuel consumption.	TeleInfra dataset, which contained different power types used by the operator obtained from a telecommunication base station.	The dataset contained anomalies in the fuel consumed.	There is no evidence that the proposed methodology can be deployed in real time.	Energy.
Munir et al. (2019)	FuseAD, method that fused statistical and deep-learning-based models for time-series anomaly detection.	Open-source time-series anomaly detection benchmark and open-source streaming anomaly detection benchmark introduced by Numenta.	Yahoo Webscope and NAB datasets.	Anomaly labels were editorially or synthetically generated by the publisher and were provided with the dataset.	There is no evidence that the proposed methodology can be implemented in real time.	Computer and technology.
Offiong et al. (2020)	Long Short-Term Memory (LSTM).	Solar-powered water taps.	Dataset acquired from real solar-powered water taps.	The dataset contained a large amount of missing values and incorrect values. It also presented high noise. Anomalies such as chipset	There is no evidence that the proposed framework can be deployed in real time.	Water science.

				errors were also observed.		
Oliveira et al. (2019)	Evaluation of Isolation Forest and Autoencoders	Railway line.	Real measurements collected from thermal, acoustic and impact sensors installed in a heavy haul railway line.	Faults related to the system context, such as abnormalities manifested due to overheating or faults related to vibration and acousting emissions.	The presented methodology was a comparative study to assess the evaluation of two specific approaches, and thus the deployment of these techniques in real time was not within the scope of the paper.	Railway.
Pereira and Silveira (2018)	Variational recurrent autoencoder. The encoder and decoder were parametrised with	Solar photovoltaic.	Solar energy generation curves representing different patterns and behaviours, such as normal sequence used	Faults and spike anomalies were annotated.	The proposed methodology can be deployed in real time.	Energy.

	recurrent neural networks to take into account the temporal dependencies of time series data.			as ground truth and a brief shading.		
Quatrini et al. (2020)	Decision forests algorithm and decision jungle algorithm.	Pharmaceutical plant.	Dataset constituted of 16 parameters that referred to either the process or the product.	Incorrect trends of relevant parameters for some specific phases.	There is no evidence that the proposed methodology can be implemented in real time.	Manufacturing and Industry.
Riazi et al. (2019)	A total of eight models were trained: angle-based outlier detection, histogram-based outlier detection, isolation forest, k -means, k -NN, Local Outlier Factor (LOF), OC-SVM, PCA.	Belt-driven single degree of freedom robot arm.	The arm was equipped with four sensors: two full-bridge strain gauge configurations and two encoders. Parameters such as cycle mode, motor shaft angle, and motor output torque were considered in the study.	5 Typical failures for this system were identified and simulated. Examples of failures are: the belt was loosened to two different levels of either 120N or 100N, an increase of the ambient temperature to 40°C, and sand was scattered on the surface.	There is no evidence that the proposed methodology can be implemented in real time.	Manufacturing and Industry.

Roy et al. (2018)	Automated feature extraction method for on-line condition monitoring based on the stack of the traditional autoencoder and an On-line Sequential Extreme Learning Machine (OSELM) network.	Experiments on bearings.	NASA Intelligent Maintenance Systems (IMS) dataset.	Faults in bearings.	The proposed methodology can be deployed in real time.	Aircraft and Space.
Santolamazza et al. (2018)	Artificial Neural Networks (ANNs).	Compressed air generation system in an industrial plant.	Data collected from system constituted by four air compressors. The inputs of the ANNs were the low pressure flow rate, the medium pressure flow rate, the external air temperature, and the state of single compressors.	Several typologies of faults were assessed through the implementation of a control chart.	The proposed methodology can be deployed in real time.	Energy.
Shi et al. (2019)	Density Peak Clustering (DPC) algorithm.	Railway vehicle.	Operation data of 8 doors in one carriage of North Extension of Guangzhou Metro Line 3.	Simulation was conducted with the abnormal making ratio 12.5\%.	The proposed methodology can be applied in real time.	Railway.

Sun et al. (2019)	Isolation Forest.	Oil field located in Natchez, Mississippi, US.	Pressure and raw distributed temperature sensing data.	Anomalies related to perturbations and controlled release events.	The methodology can be deployed in real time.	Environmental.
Taheritanjani et al. (2019)	In relation to unsupervised anomaly detection techniques, the following techniques were assessed: One-Class SVM, One-Class NN, Isolation Forest, Local Outlier Factor, Patch Based Autoencoder, and Small Part Autoencoder.	Fasteners.	2019 images of 12 different bolt types each with intact and damaged samples.	A test set was utilised with 207 pictures of damaged fasteners and 213 intact fasteners. Different types of damages were assessed.	There is no evidence that the presented methodology can be deployed in real time.	Aircraft and Space.

Xiao et al. (2020)	<p>BC-CEEMDAN-GRU. Characteristics were reconstructed by using Basic Characteristics based Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (BC-CEEMDAN). The most sensitive features were selected by employing a linear combination of monotonicity and correlation criteria. The selected features were the input of the Gated Recurrent Unit (GRU) neural network.</p>	Experiments on bearings.	IEEE-PHM-2012-Challenge and XJTU-SY datasets.	The datasets contained several faults.	There is no evidence that the proposed methodology can be deployed in real time.	Manufacturing and Industry.
--------------------	--	--------------------------	---	--	--	-----------------------------

Xu et al. (2019)	Isolated forest.	Rolling bearing equipment.	Vibration signal data generated during actual production.	Outer ring failure, inner ring failure and ball failure.	There is no evidence that the methodology can be implemented in real time.	Manufacturing and Industry.
Xue et al. (2019)	Combination of one-dimensional convolution with Generative Adversarial Networks (GAN).	Railway switch.	Normal and abnormal data collected from a railway switch.	The nature of the faults were not deeply examined in the study.	The proposed methodology can be applied in real time.	Railway.

Appendix D

Main Results of Case Study 8. MA

Framework

The subsequent sections of the present appendix relate to each of the components/sub-systems, and their respective parameters, analysed (see a comprehensive description of the system considered in section 7.5. *Case 8. MA framework*).

D.1. Main Engine Water Cooling System

The parameter analysed refers to the cooling fresh water inlet pressure. A graphical representation of such a parameter is expressed in Fig. D.1.1. The descriptive statistics is also introduced in Table E.1.1.

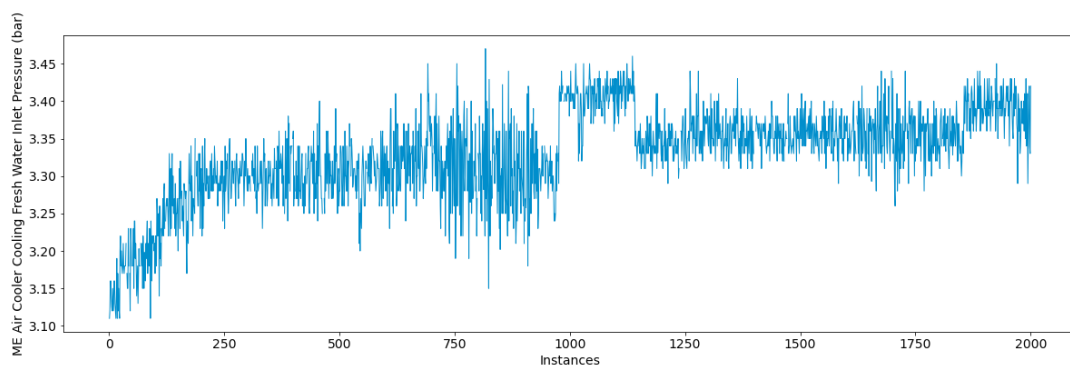


Fig. D.1.1. Graphical representation of the Cooling Fresh Water (CFW) inlet pressure parameter.

Table D.1. 1. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
CFW inlet press. (bar)	3.33	0.06	3.11	3.3	3.34	3.37	3.47

As part of the data pre-processing phase, the identification of operational states step has been implemented (see section 4.5. *A Novel Framework for the Identification of Steady States* for a comprehensive explanation of such a step). In total, only one operational state has been identified, as perceived in Fig. D.1.2.

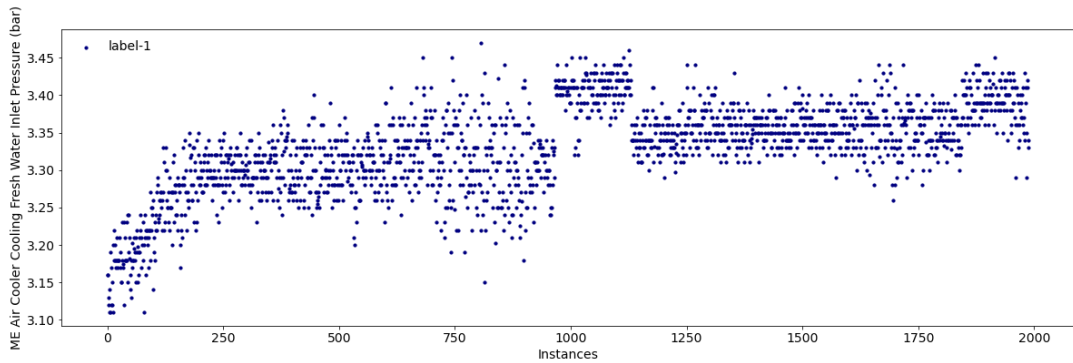
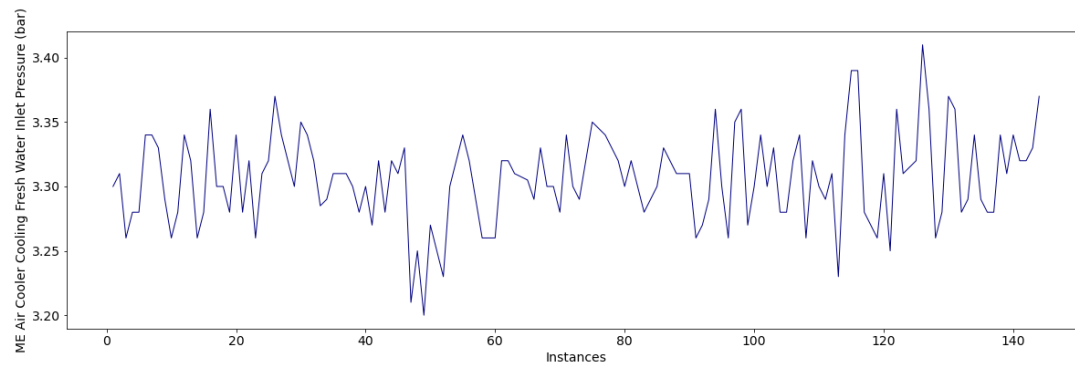
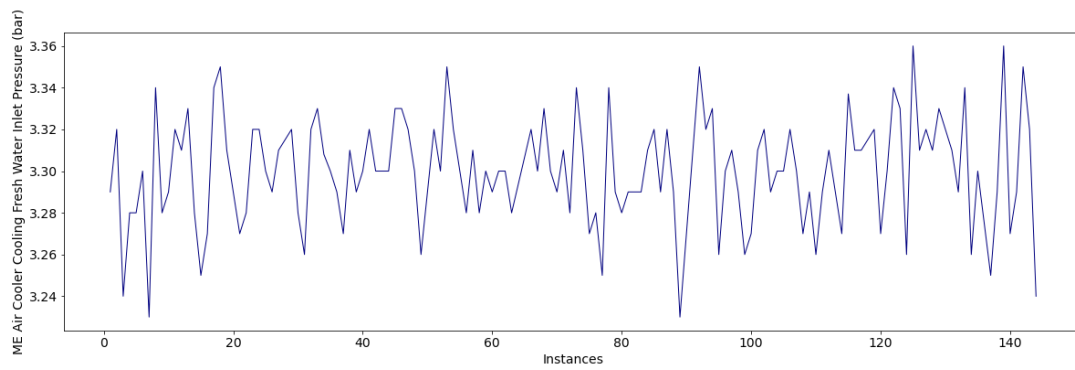
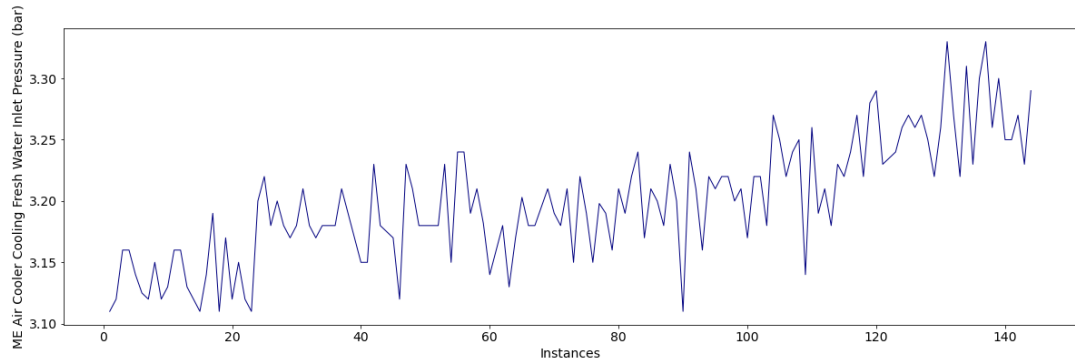


Fig. D.1. 2. Identification of the operational states for the monitored parameter.

Subsequently, due to the lack of fault data, both collective anomalies and degradation patterns are simulated. Some examples are presented in Figs. E.1.4 – E.1.5. Examples of normal sequences are also introduced in Fig. D.1.3.



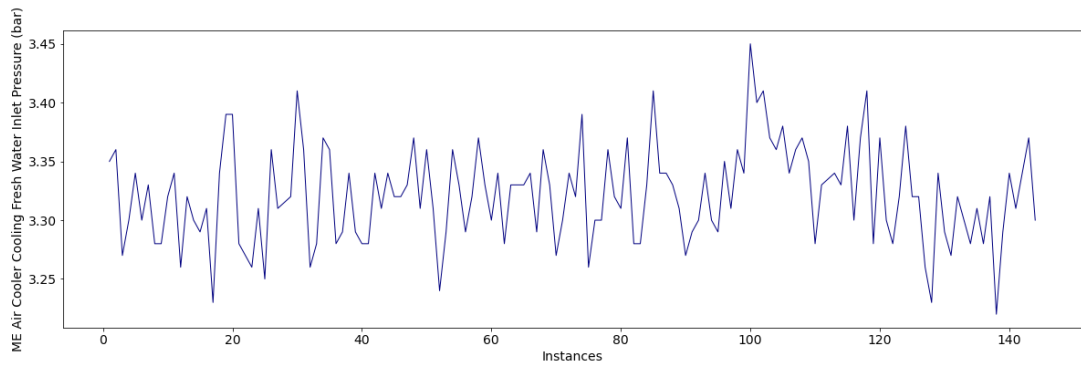
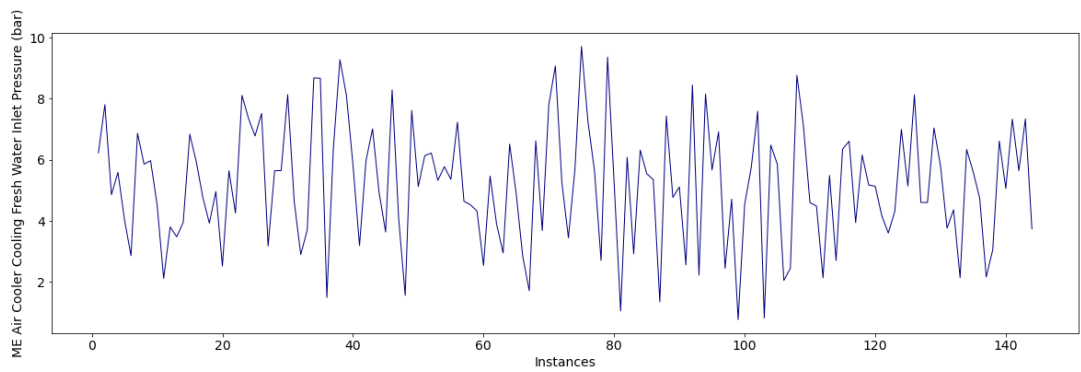
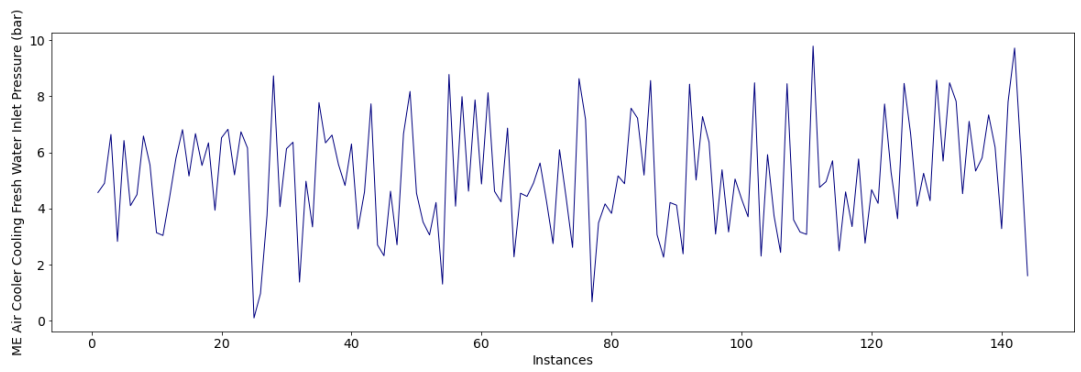


Fig. D.1.3. Example of normal sequences.



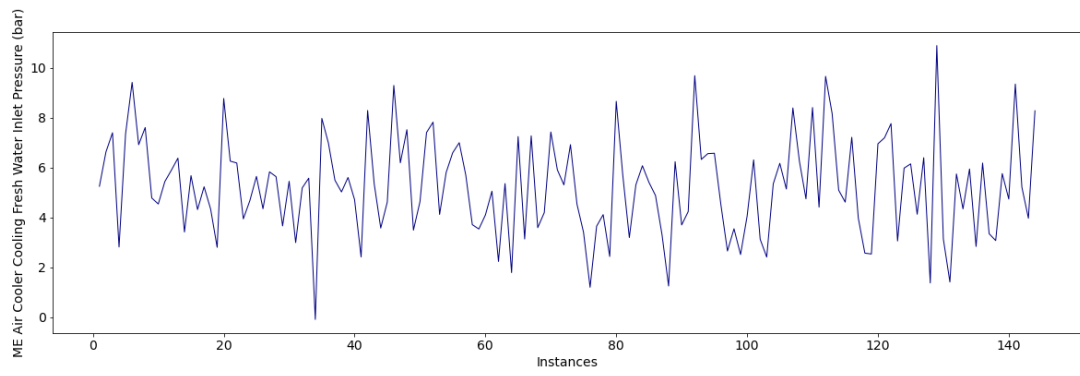
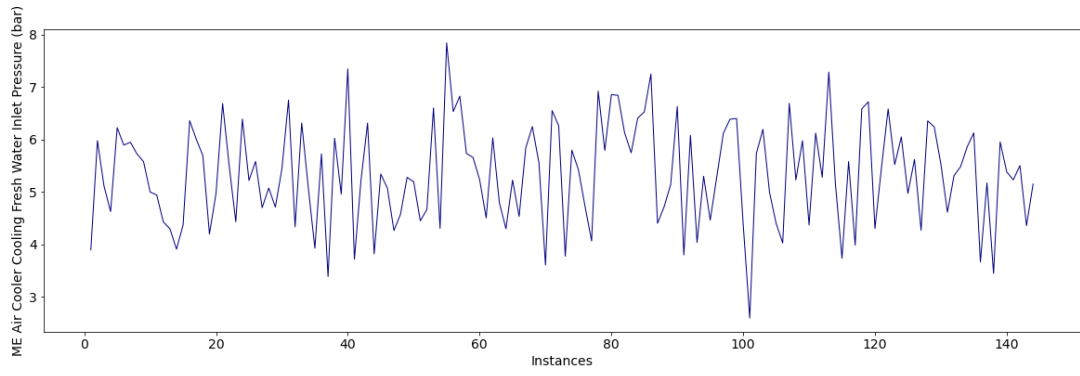
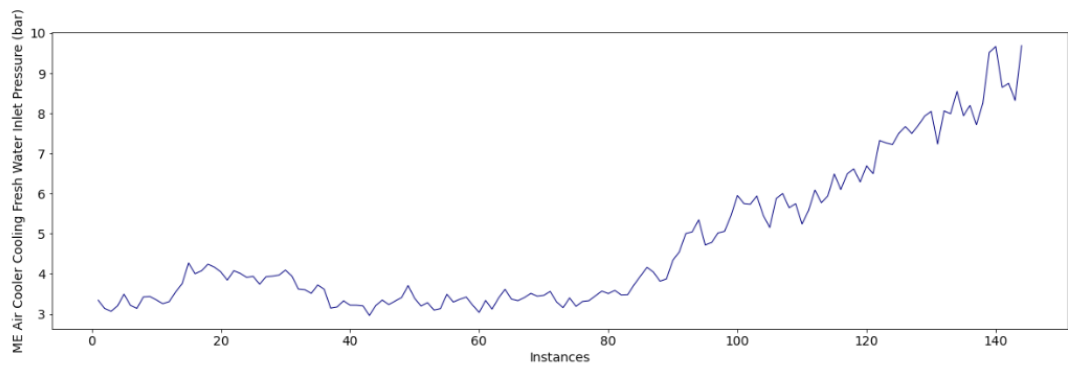


Fig. D.1.4. Example of sequences with collective anomalies.



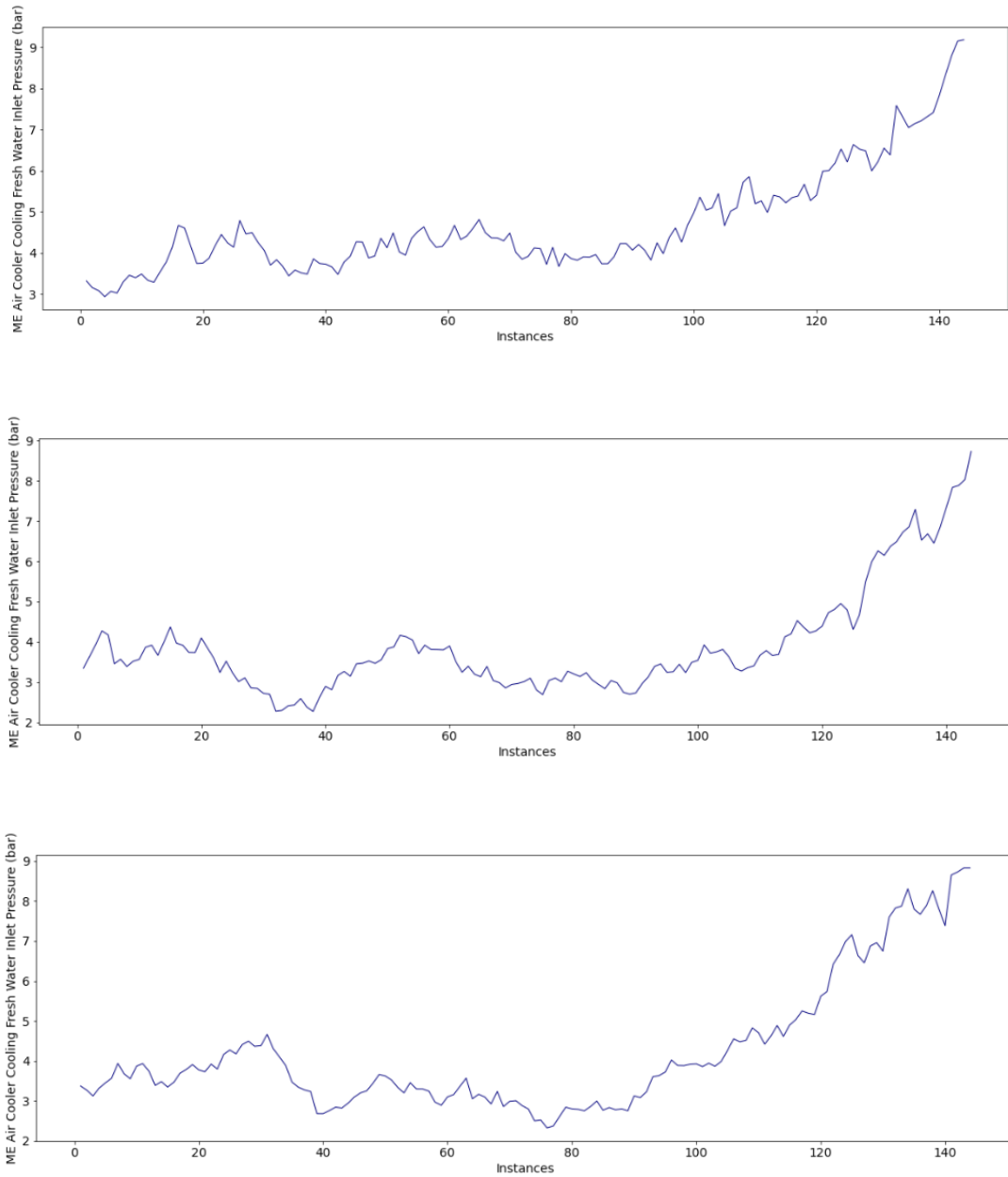


Fig. D.1.5. Example of sequences with degradation patterns.

As part of the MA framework, the subsequent module to be applied is the diagnostic analytics module. Accordingly, the fault detection step is implemented as stated in section 5.3.1. *Fault Detection*. As perceived in the histograms (Figs. E.1.6 – E.1.8), a

simple threshold is adequate in this case study to distinguish the normal sequences from the abnormal sequences.

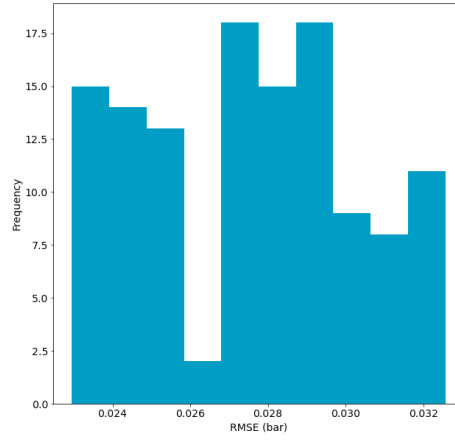


Fig. D.1.6. Histogram of the reconstructed errors of the normal sequences (test set).

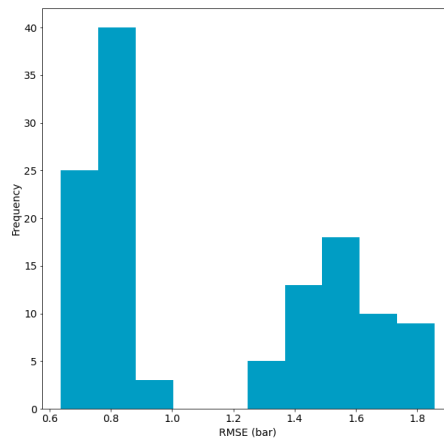


Fig. D.1.7. Histogram of the reconstructed errors of the sequences with collective anomalies.

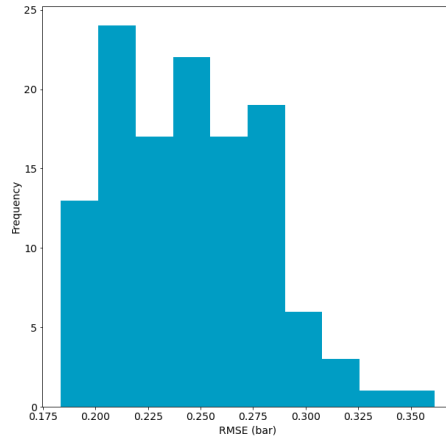
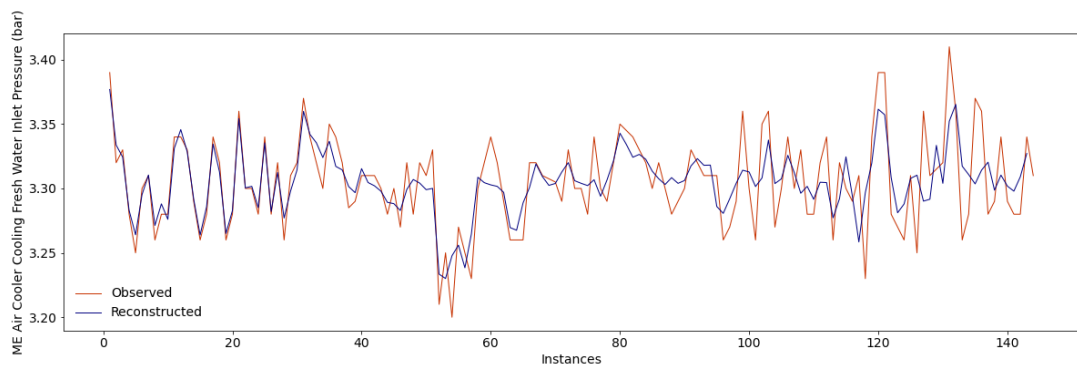
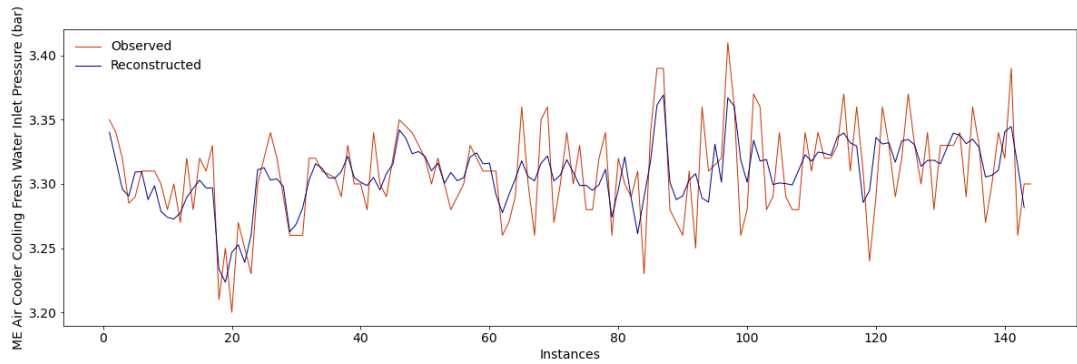


Fig. D.1. 8. Histogram of the reconstructed errors of the sequences with degradation patterns.

Examples of reconstructed sequences are also introduced in Figs. E.1.9 – E.1.11.



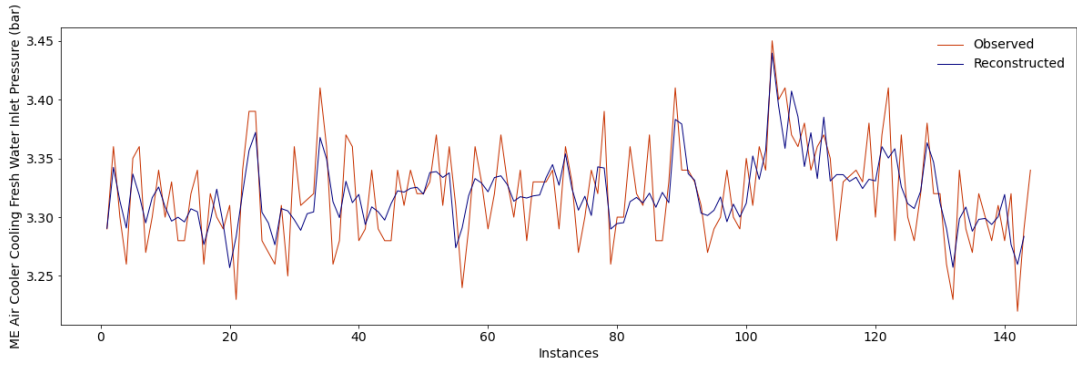
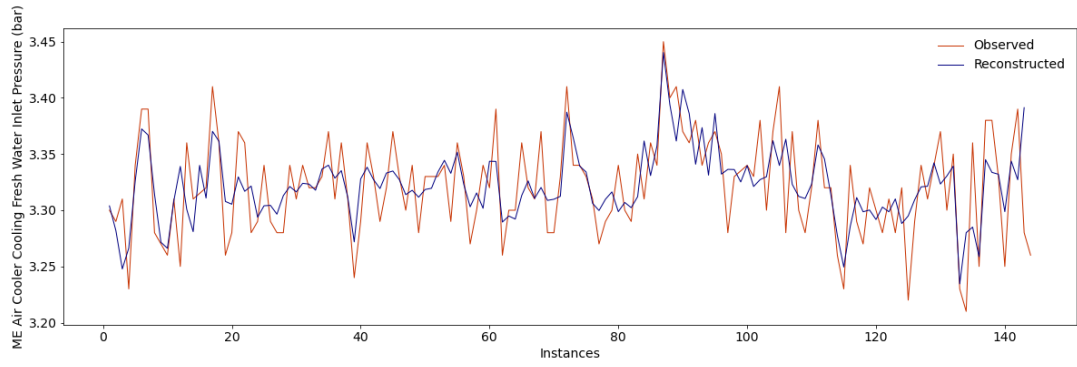
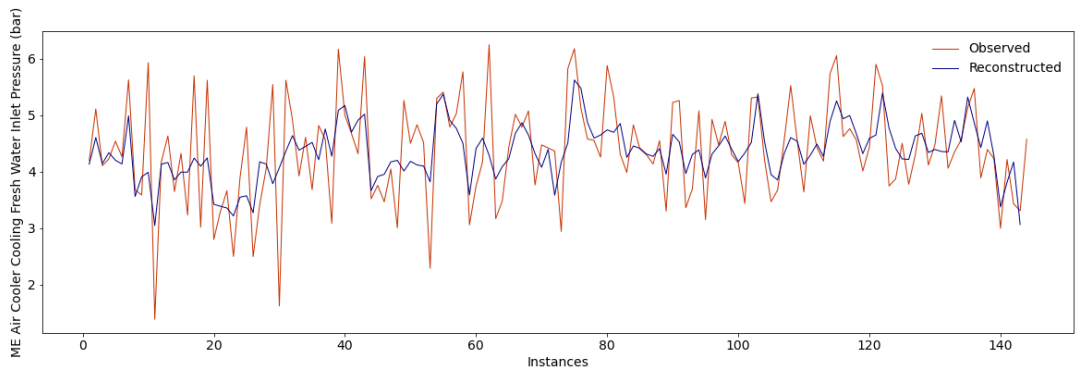


Fig. D.1.9. Example of normal reconstructed sequences (test set).



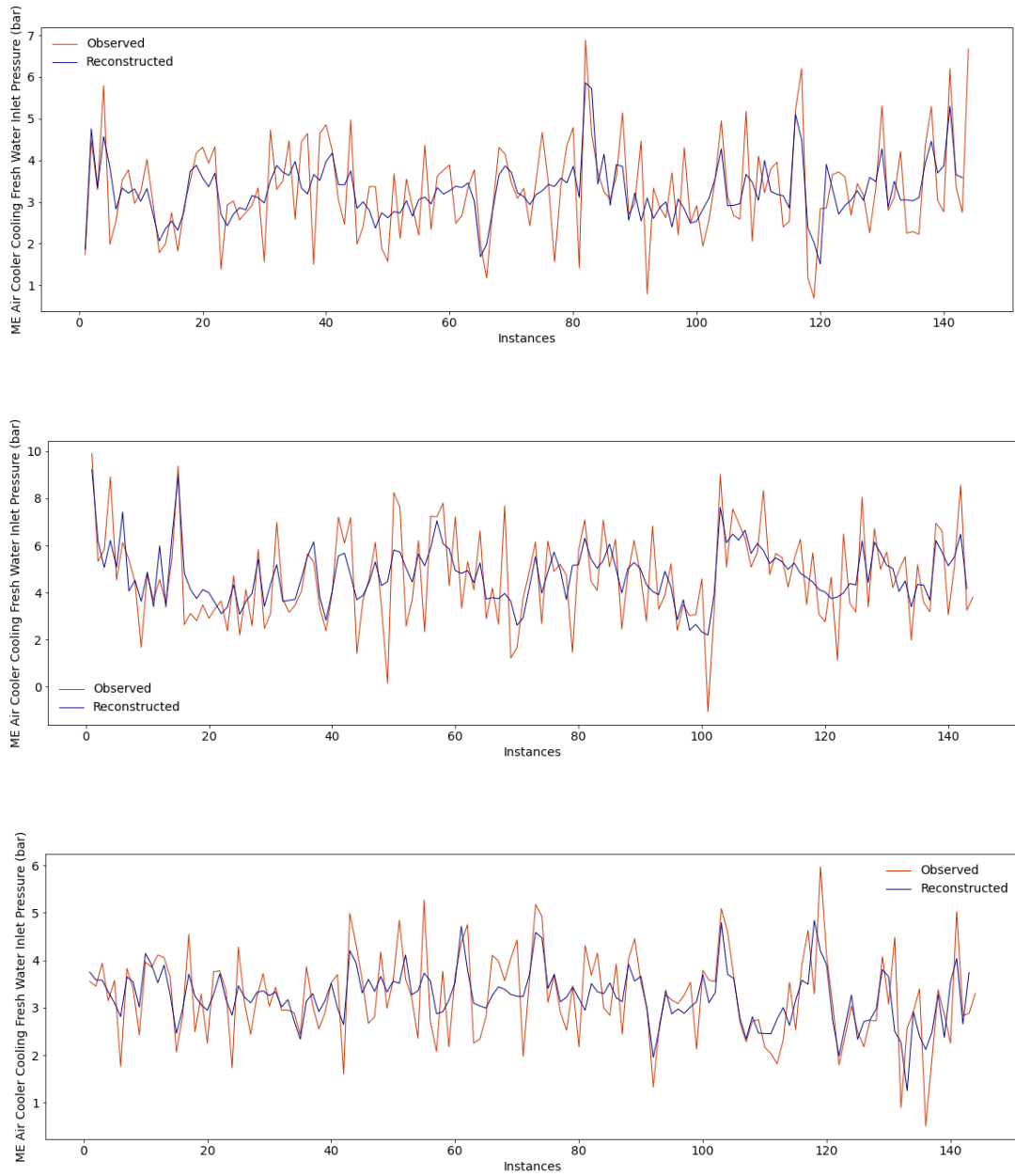
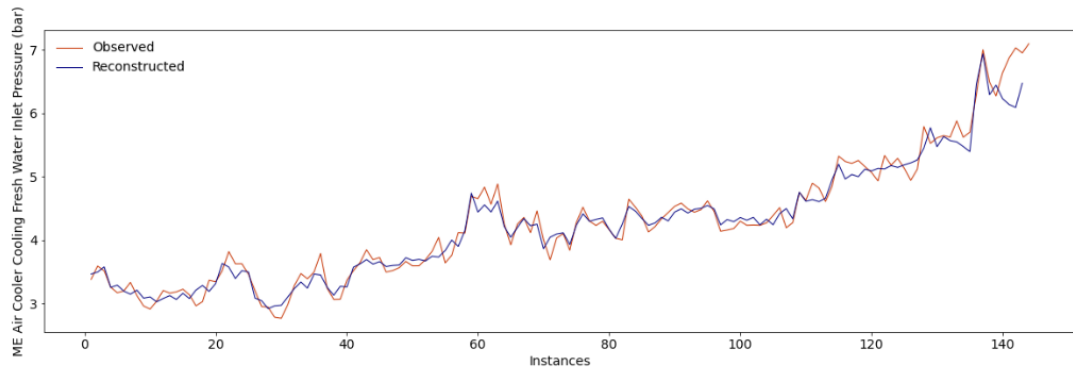
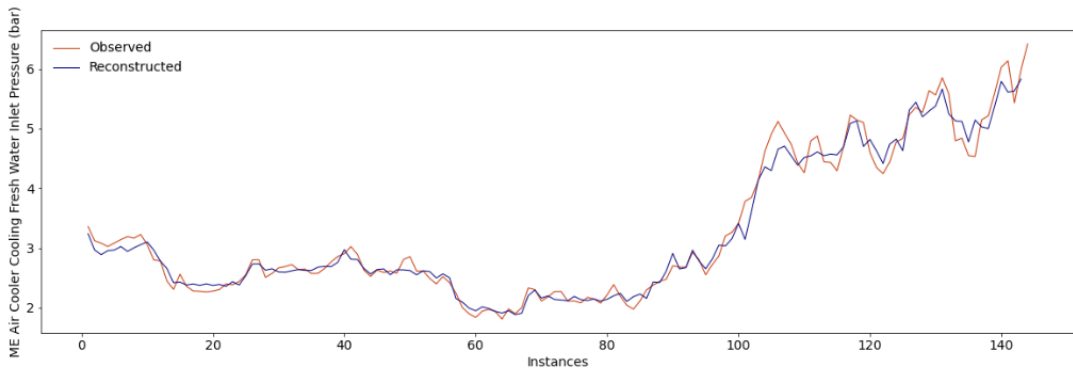
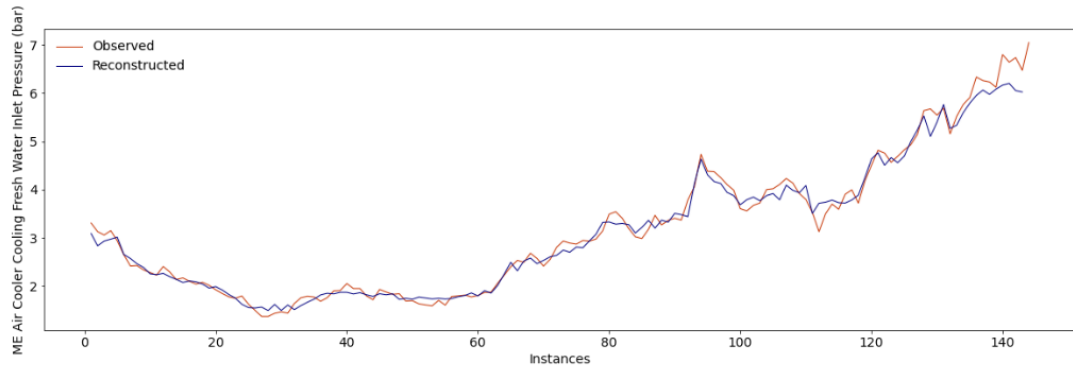


Fig. D.1.10. Example of reconstructed sequences with collective anomalies.



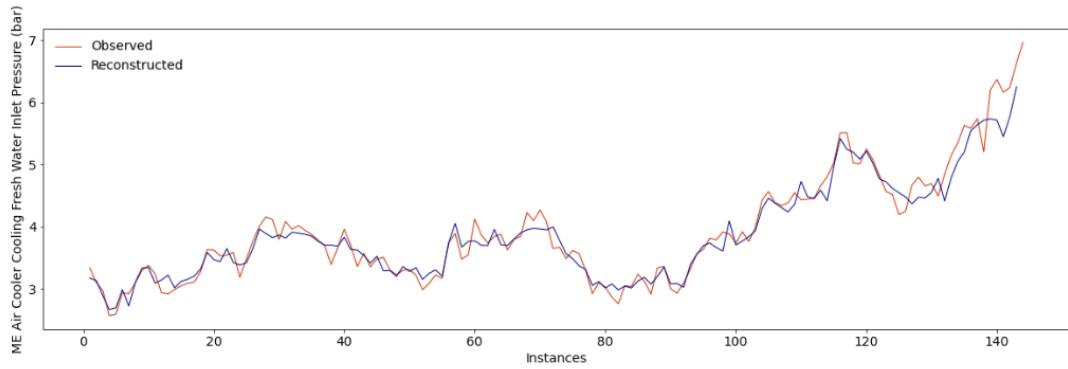
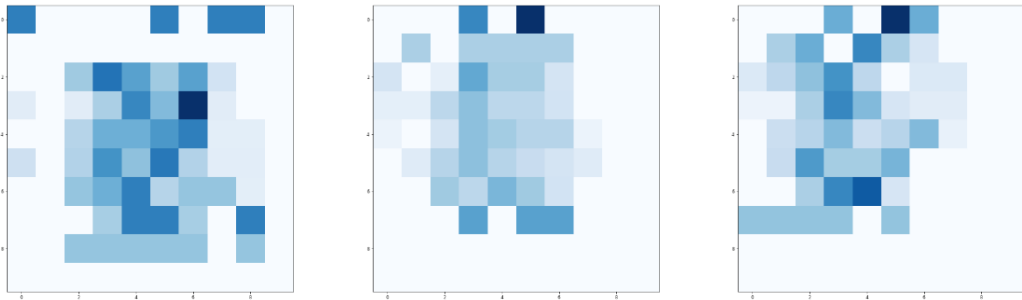


Fig. D.1.11. Example of reconstructed sequences with degradation patterns.

Those sequences detected as anomalous are then classified into two categories: sequences with collective anomalies, and sequences with degradation patterns. Accordingly, as the implemented approach refers to a time series imaging approach (see section 5.3.2. *Fault Identification*), the anomalous sequences detected are transformed into images. Examples of these can be perceived in Figs. E.1.12 – E.1.13.



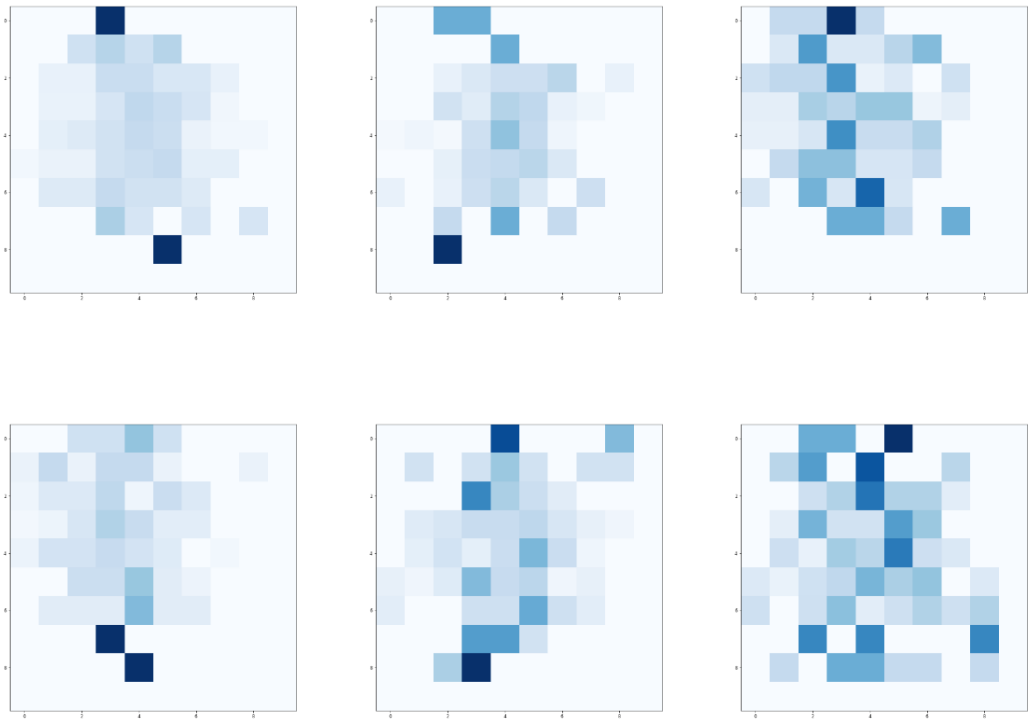
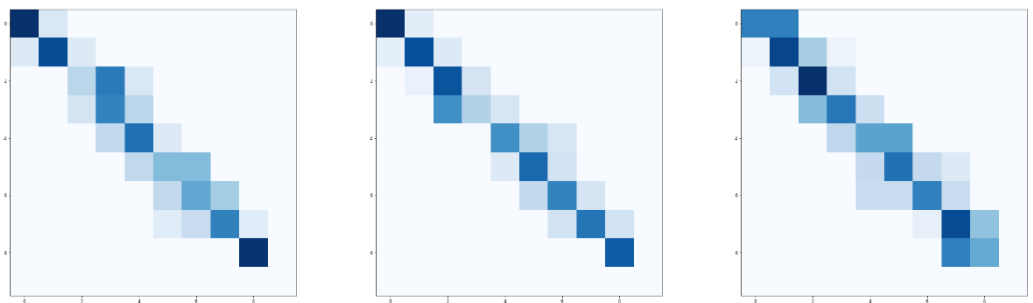


Fig. D.1.12. Images with collective anomalies.



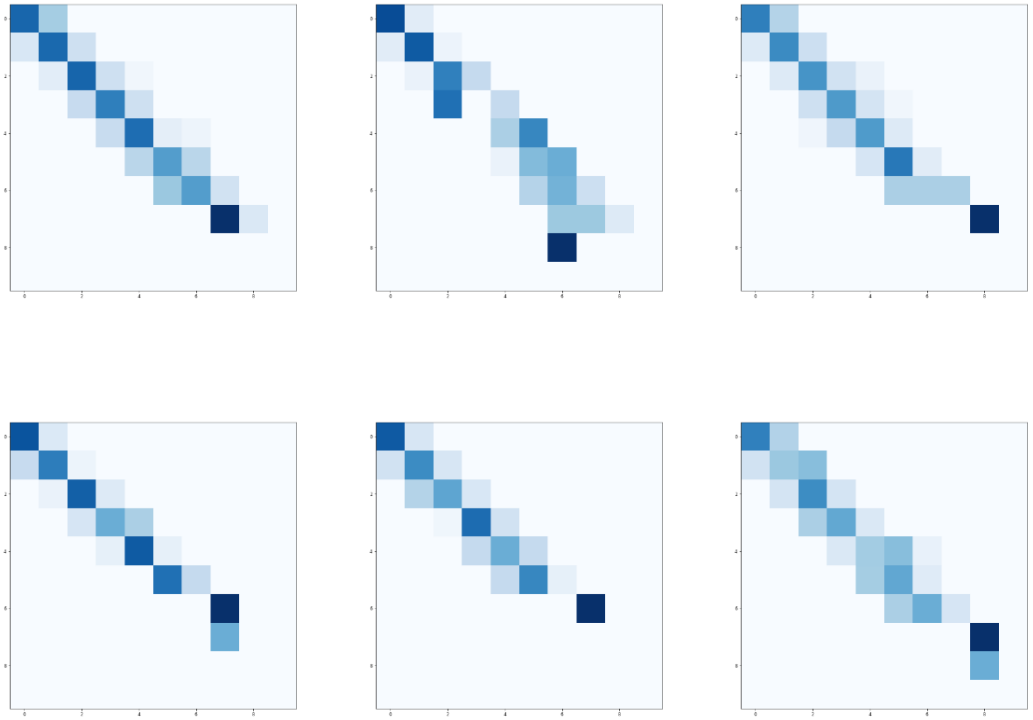


Fig. D.1.13. Images with degradation patterns.

As it can be perceived, the two categories can be easily distinguished. This aspect facilitated the achievement of the maximum accuracy score in this process, thus classifying all the images adequately.

By adequately classifying such images, the sequences with degradation patterns are selected so that the RUL can be predicted. Examples of such a prediction are presented in Fig. D.1.14. The RMSE and Maintenance Score of the first 100 sequences are also presented in Table E.1.2.

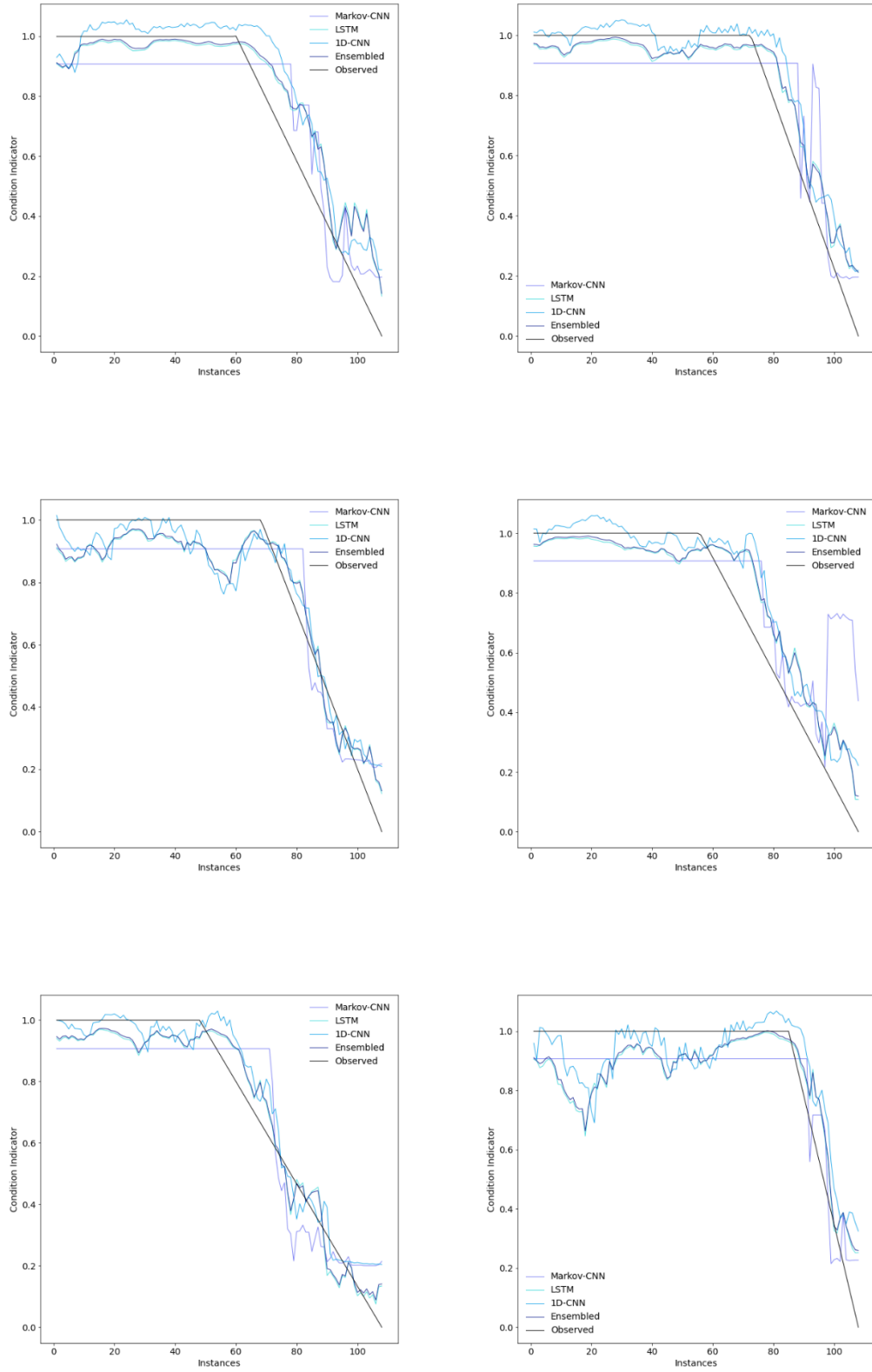


Fig. D.1.14. Examples of condition indicator.

Table D.1.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.

Sequences	RMSE (Minutes)				Maintenance Score			
	Markov-CNN	LSTM	1D-CNN	Ensembled	Markov-CNN	LSTM	1D-CNN	Ensembled
Sequence 1	34.03	37.74	36.39	37.37	1.54	1.89	1.86	1.87
Sequence 2	23.77	16.96	19.37	17.01	1.22	0.81	1.00	0.81
Sequence 3	24.31	22.02	25.11	22.04	1.28	1.13	1.22	1.12
Sequence 4	39.95	29.48	32.27	29.54	2.45	1.83	1.86	1.82
Sequence 5	21.90	19.25	18.95	19.01	1.18	0.93	0.92	0.92
Sequence 6	55.15	32.82	32.82	32.49	3.83	2.10	1.75	2.05
Sequence 7	45.28	36.81	38.03	36.81	2.09	1.54	1.58	1.51
Sequence 8	53.36	34.79	37.39	34.91	3.27	1.68	1.78	1.67
Sequence 9	66.19	69.35	69.25	69.24	2.97	3.57	3.49	3.55
Sequence 10	17.94	14.90	17.50	14.89	1.10	0.88	0.76	0.85
Sequence 11	40.20	44.57	45.94	44.59	2.01	2.00	2.07	1.99
Sequence 12	29.44	23.54	27.93	23.70	1.90	1.35	1.42	1.33
Sequence 13	14.62	18.38	17.92	18.02	1.01	1.06	0.91	1.03
Sequence 14	25.89	21.17	21.35	20.92	1.37	1.02	1.00	0.99
Sequence 15	40.93	34.91	34.81	34.70	2.44	1.99	2.05	1.97
Sequence 16	21.17	19.38	18.37	19.04	1.40	1.19	1.08	1.16
Sequence 17	30.97	35.80	38.92	35.94	1.54	1.57	1.75	1.56
Sequence 18	43.39	43.28	45.16	43.31	2.27	2.25	2.39	2.26
Sequence 19	44.04	28.65	32.21	28.88	2.03	1.25	1.42	1.25
Sequence 20	42.87	56.09	65.98	56.95	2.06	2.82	3.16	2.85
Sequence 21	29.68	28.23	32.87	28.46	1.40	1.41	1.71	1.43
Sequence 22	44.67	41.65	45.80	41.95	2.22	1.89	2.19	1.90
Sequence 23	20.03	15.46	17.31	15.42	1.17	0.80	0.72	0.76

Sequence 24	14.65	13.05	13.37	12.92	0.97	0.83	0.72	0.81
Sequence 25	59.68	53.74	55.16	53.76	3.37	3.15	3.28	3.16
Sequence 26	29.24	26.47	26.86	26.37	1.41	1.23	1.32	1.23
Sequence 27	49.44	52.71	54.17	52.76	2.32	2.61	2.66	2.61
Sequence 28	32.46	30.65	32.22	30.57	1.55	1.66	1.84	1.66
Sequence 29	67.46	52.70	51.88	52.27	4.10	3.27	2.95	3.24
Sequence 30	52.08	36.67	33.68	36.00	3.24	2.28	1.86	2.23
Sequence 31	16.10	14.01	17.13	14.10	1.06	0.82	0.78	0.80
Sequence 32	22.29	23.70	25.28	23.71	1.41	1.45	1.42	1.44
Sequence 33	16.87	14.54	18.73	14.67	1.09	0.91	1.11	0.92
Sequence 34	34.72	29.26	32.36	29.42	1.73	1.56	1.75	1.58
Sequence 35	16.80	16.57	18.91	16.47	1.08	0.88	0.73	0.83
Sequence 36	34.93	34.24	35.12	34.08	1.88	1.91	2.00	1.92
Sequence 37	46.53	48.43	48.50	48.33	2.29	2.30	2.39	2.30
Sequence 38	27.86	25.74	25.29	25.56	1.44	1.14	0.98	1.11
Sequence 39	37.98	44.41	45.64	44.42	1.91	1.94	2.04	1.94
Sequence 40	30.88	24.02	25.56	24.04	1.49	1.22	1.29	1.23
Sequence 41	22.62	23.95	27.22	24.01	1.49	1.36	1.38	1.34
Sequence 42	13.30	20.74	24.67	20.98	0.94	1.22	1.22	1.21
Sequence 43	18.70	18.77	20.96	18.78	1.06	0.83	0.86	0.79
Sequence 44	24.50	18.70	22.05	18.80	1.31	0.98	0.95	0.96
Sequence 45	23.51	17.28	16.81	16.92	1.19	0.90	0.90	0.88
Sequence 46	21.97	21.21	25.02	21.44	1.24	0.95	1.00	0.93
Sequence 47	26.46	18.65	21.01	18.70	1.33	0.89	0.79	0.86
Sequence 48	81.96	74.22	76.54	74.37	4.59	3.78	4.04	3.80
Sequence 49	59.19	47.03	48.65	46.97	4.05	3.30	3.28	3.29

Sequence 50	55.49	60.68	56.26	60.18	3.33	3.59	3.37	3.56
Sequence 51	18.82	25.98	25.58	25.64	1.03	1.18	1.19	1.16
Sequence 52	67.88	57.01	56.58	56.86	3.86	3.23	3.21	3.23
Sequence 53	42.65	29.45	36.45	30.00	1.93	1.25	1.44	1.25
Sequence 54	23.45	31.34	31.64	31.07	1.52	2.08	1.82	2.05
Sequence 55	20.96	19.93	20.72	19.74	1.19	0.98	0.92	0.96
Sequence 56	30.30	28.12	30.85	28.22	1.81	1.65	1.63	1.65
Sequence 57	23.09	22.06	22.59	21.80	1.22	1.19	1.24	1.18
Sequence 58	30.01	25.03	26.55	24.89	1.95	1.53	1.45	1.49
Sequence 59	25.84	25.84	29.43	25.95	1.36	1.14	1.34	1.13
Sequence 60	24.50	21.79	25.27	21.94	1.19	0.98	1.09	0.96
Sequence 61	17.02	15.11	18.37	15.20	1.06	1.00	1.10	1.00
Sequence 62	51.30	39.00	36.29	38.53	3.29	2.44	2.16	2.40
Sequence 63	32.63	25.78	27.35	25.63	1.95	1.28	1.38	1.25
Sequence 64	45.99	25.06	23.84	24.74	2.21	1.17	1.07	1.16
Sequence 65	71.72	54.51	55.93	54.53	4.36	3.23	3.21	3.21
Sequence 66	22.78	22.45	22.80	22.36	1.26	0.92	0.94	0.90
Sequence 67	20.97	30.23	30.94	30.03	1.05	1.69	1.72	1.69
Sequence 68	30.03	19.99	21.45	20.03	1.50	0.89	0.97	0.89
Sequence 69	50.16	44.95	45.17	44.88	2.53	2.20	2.32	2.21
Sequence 70	32.60	19.94	23.38	20.12	1.52	0.89	0.85	0.87
Sequence 71	23.66	17.64	19.82	17.72	1.28	0.83	0.78	0.80
Sequence 72	17.43	16.54	17.61	16.38	1.05	0.76	0.81	0.73
Sequence 73	27.00	20.29	23.19	20.35	1.73	1.03	0.93	0.99
Sequence 74	19.95	15.42	17.38	15.36	1.02	0.77	0.78	0.75
Sequence 75	38.38	30.76	32.12	30.79	1.86	1.35	1.40	1.34

Sequence 76	24.03	16.86	17.02	16.54	1.57	1.01	0.85	0.97
Sequence 77	30.63	31.35	32.06	31.24	1.98	2.02	1.85	2.00
Sequence 78	34.75	32.05	33.87	32.07	1.59	1.56	1.71	1.56
Sequence 79	36.37	30.52	35.16	30.78	2.03	1.84	2.18	1.87
Sequence 80	37.40	40.19	41.73	40.15	2.22	2.82	2.78	2.82
Sequence 81	23.98	26.68	28.11	26.60	1.28	1.36	1.34	1.33
Sequence 82	39.22	43.54	45.12	43.51	1.96	2.50	2.59	2.51
Sequence 83	53.29	59.96	52.57	59.13	3.23	3.59	3.01	3.52
Sequence 84	43.58	41.00	40.91	40.75	3.01	2.68	2.48	2.65
Sequence 85	25.49	21.13	20.47	20.81	1.49	1.43	1.16	1.39
Sequence 86	22.23	18.68	20.16	18.65	1.19	0.80	0.83	0.77
Sequence 87	54.67	47.32	46.68	47.13	3.65	3.01	2.86	2.99
Sequence 88	13.44	15.57	16.80	15.46	0.95	1.01	0.91	0.99
Sequence 89	30.25	25.86	27.21	25.82	1.52	1.09	1.11	1.06
Sequence 90	36.72	25.70	25.62	25.60	1.84	1.30	1.35	1.31
Sequence 91	62.26	57.30	58.05	57.18	3.34	3.11	3.13	3.12
Sequence 92	18.83	15.18	15.55	14.97	1.07	0.81	0.70	0.78
Sequence 93	19.58	12.54	14.51	12.46	1.09	0.66	0.59	0.63
Sequence 94	18.34	14.39	19.62	14.52	1.02	0.72	0.95	0.71
Sequence 95	18.79	15.73	20.17	15.91	1.13	0.75	0.78	0.72
Sequence 96	51.92	34.59	36.54	34.60	2.99	1.91	1.95	1.91
Sequence 97	46.95	35.49	35.93	35.36	2.61	2.04	2.19	2.06
Sequence 98	35.75	30.53	30.63	30.37	1.86	1.54	1.63	1.54
Sequence 99	32.65	29.22	31.22	29.28	1.58	1.16	1.12	1.13
Sequence 100	50.06	33.35	40.64	33.80	3.24	2.11	2.46	2.13
Median	29.74	25.86	27.93	25.82	1.54	1.30	1.38	1.31

D.2. Main Engine Cylinder 1

The parameter analysed refers to the exhaust gas outlet temperature. A graphical representation of such a parameter is expressed in Fig. D.2.1. The descriptive statistics is also introduced in Table E.2.1.

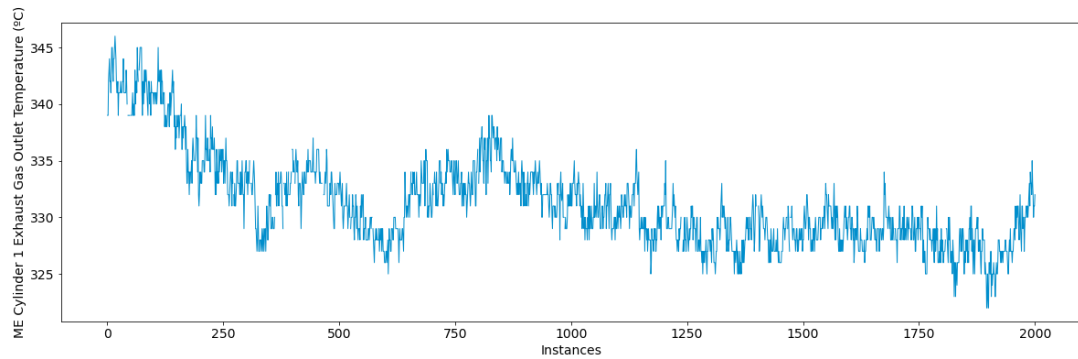


Fig. D.2.1. Graphical representation of the exhaust gas outlet temperature parameter.

Table D.2.1. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
Cyl. 1 Exh. Gas Out. Temp	331.31	4.06	322	328	331	333	346

As part of the data pre-processing phase, the identification of operational states step has been implemented (see section 4.5. *A Novel Framework for the Identification of Steady States* for a comprehensive explanation of such a step). In total, only one operational state has been identified, as perceived in Fig. D.2.2.

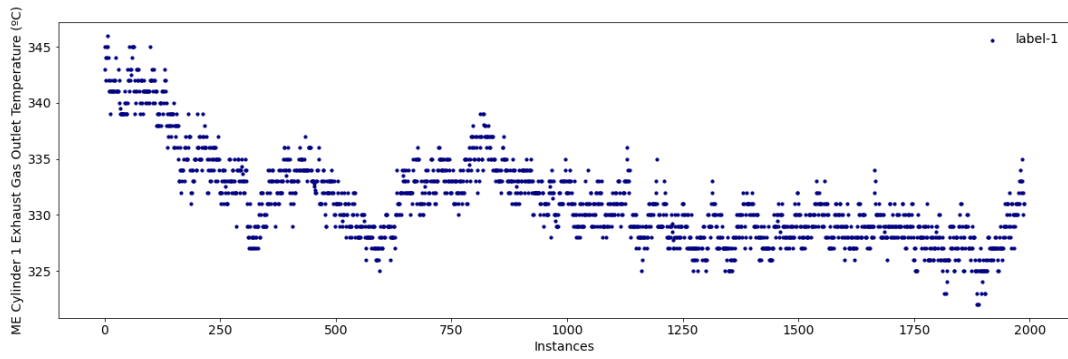
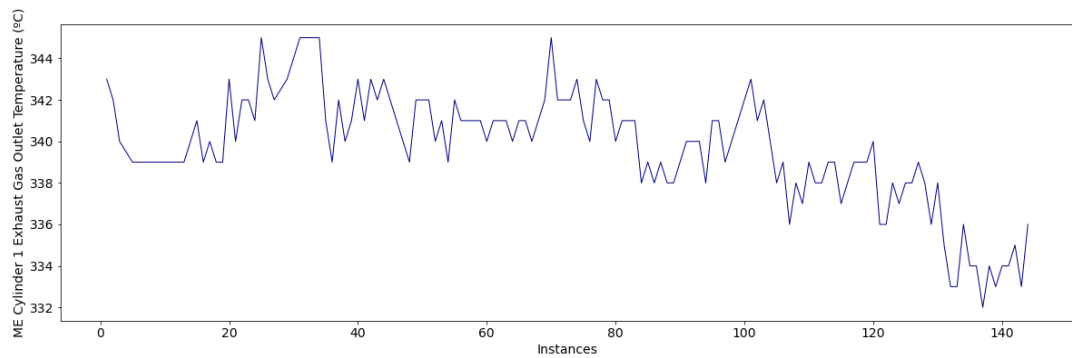


Fig. D.2.2. Identification of the operational states for the monitored parameter.

Subsequently, due to the lack of fault data, both collective anomalies and degradation patterns are simulated. Some examples are presented in Figs. E.2.4 – E.2.5. Examples of normal sequences are also introduced in Fig. D.2.3.



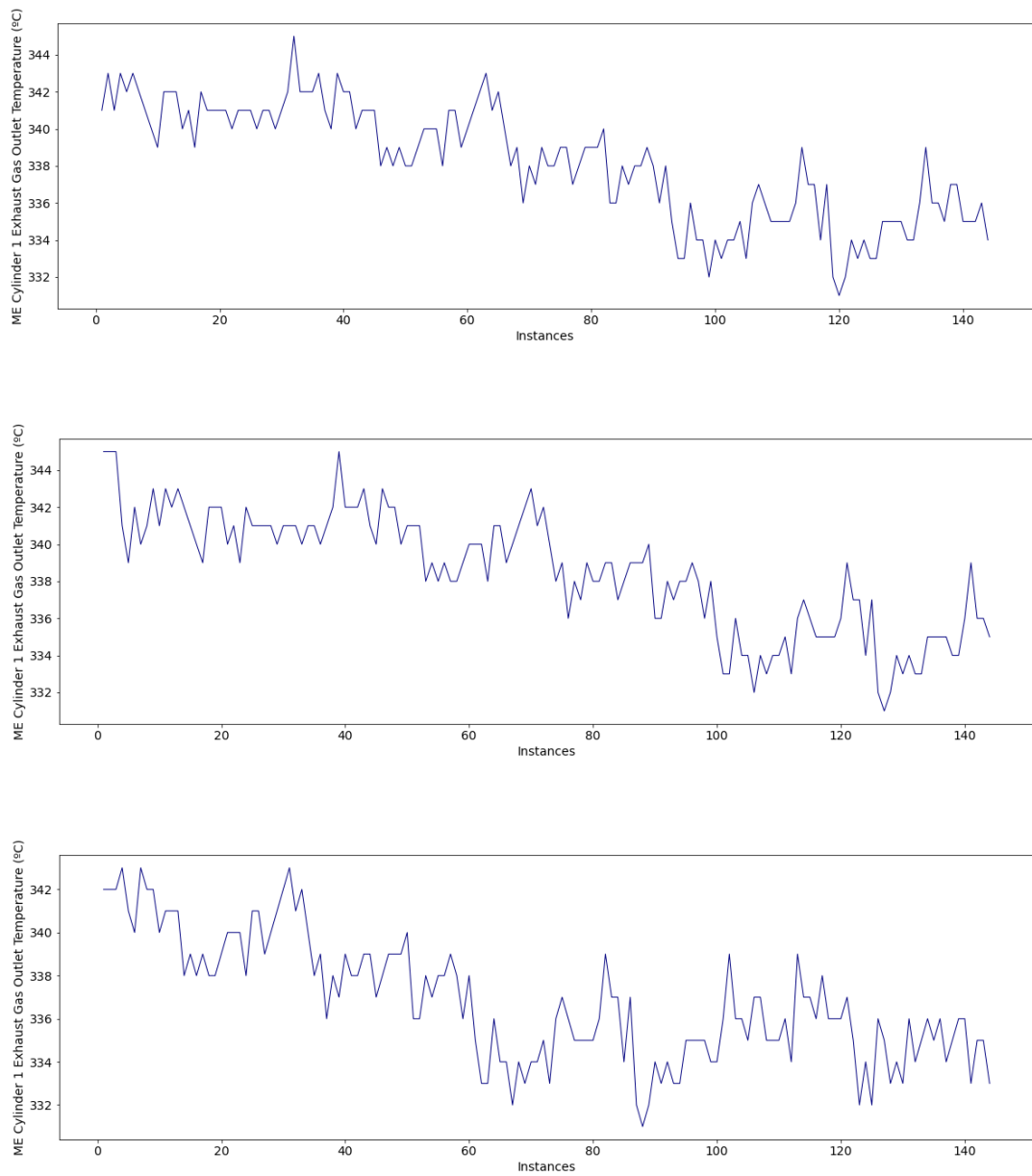


Fig. D.2.3. Example of normal sequences.

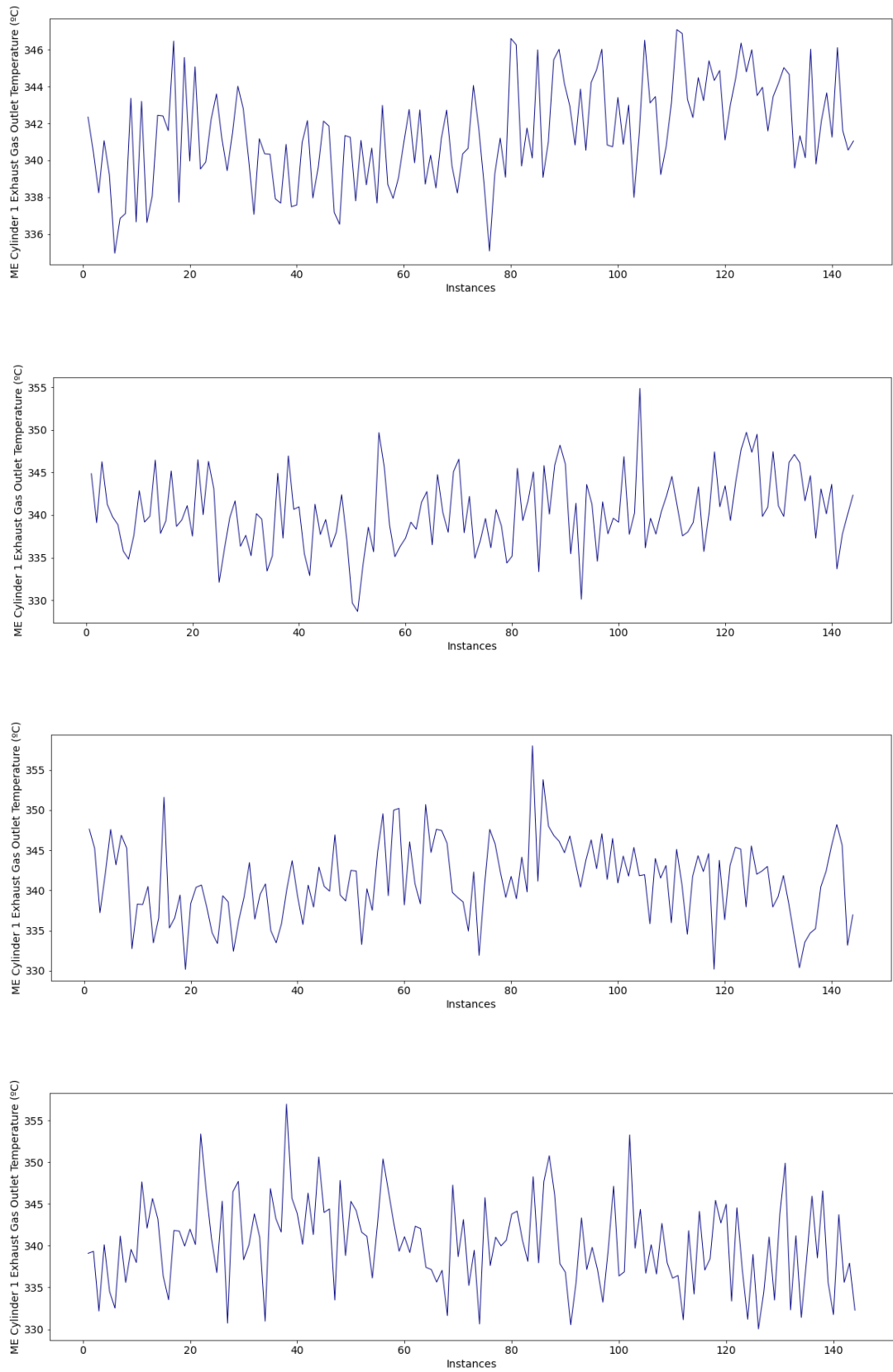
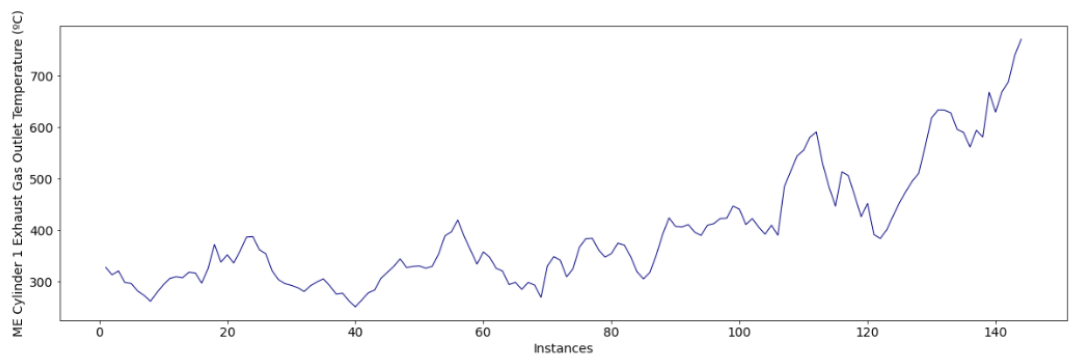
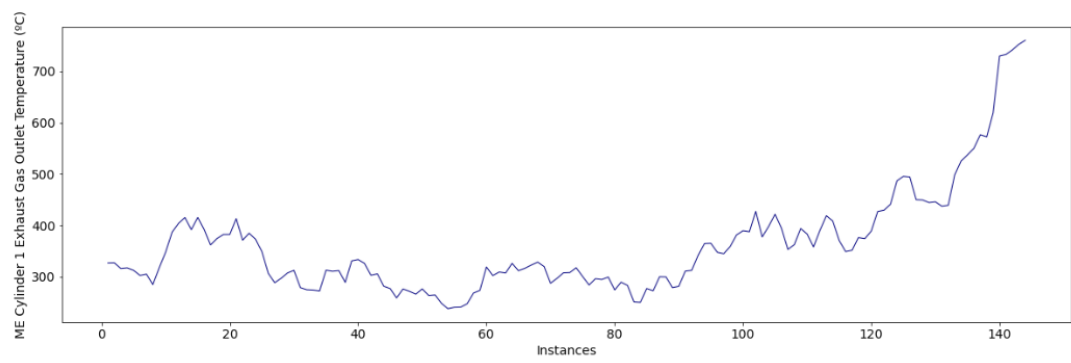
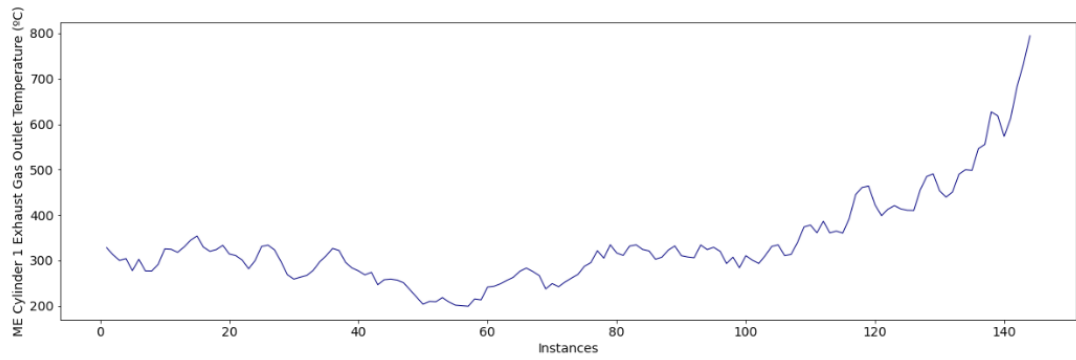


Fig. D.2.4. Example of sequences with collective anomalies.



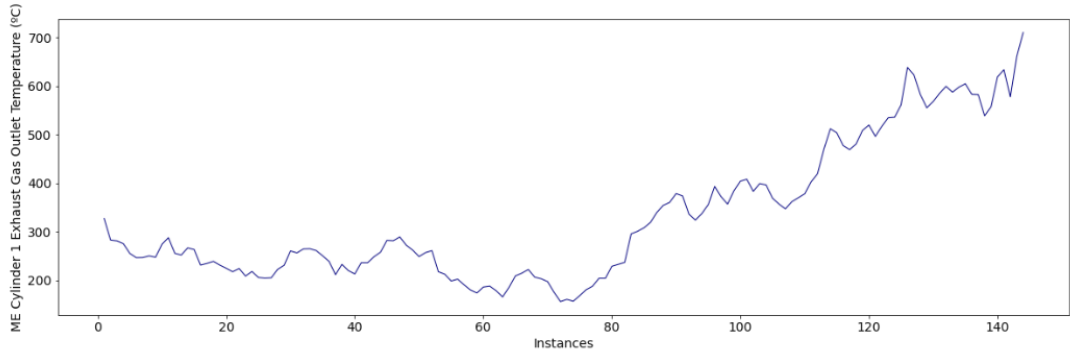


Fig. D.2.5. Example of sequences with degradation patterns.

As part of the MA framework, the subsequent module to be applied is the diagnostic analytics module. Accordingly, the fault detection step is implemented as stated in section 5.3.1. *Fault Detection*. As perceived in the histograms (Figs. E.2.6 – E.2.8), a simple threshold is adequate in this case study to distinguish the normal sequences from the abnormal sequences.

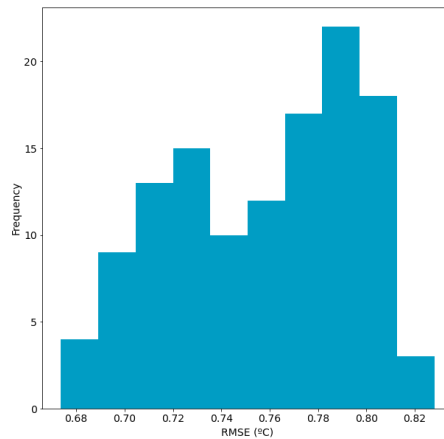


Fig. D.2.6. Histogram of the reconstructed errors of the normal sequences (test set).

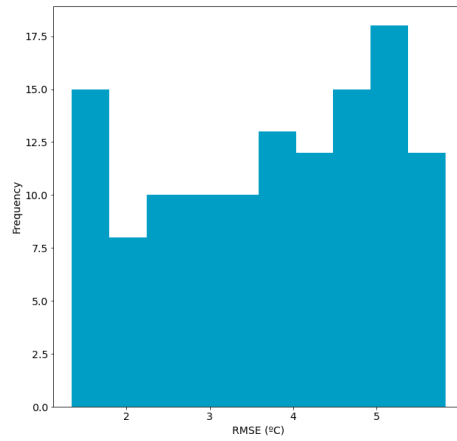


Fig. D.2.7. Histogram of the reconstructed errors of the sequences with collective anomalies.

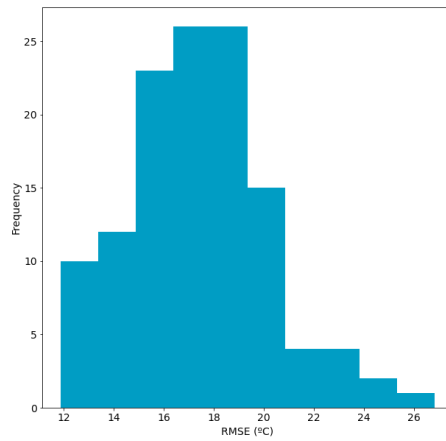


Fig. D.2.8. Histogram of the reconstructed errors of the sequences with degradation patterns.

Examples of reconstructed sequences are also introduced in Figs. E.2.9 – E.2.11.

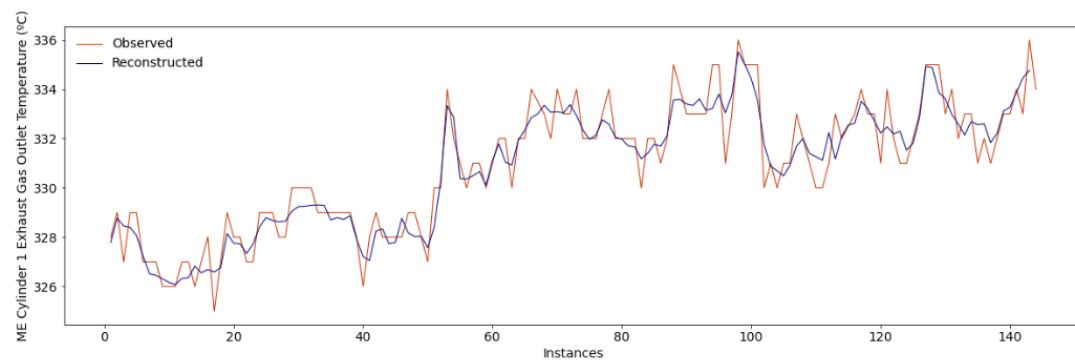
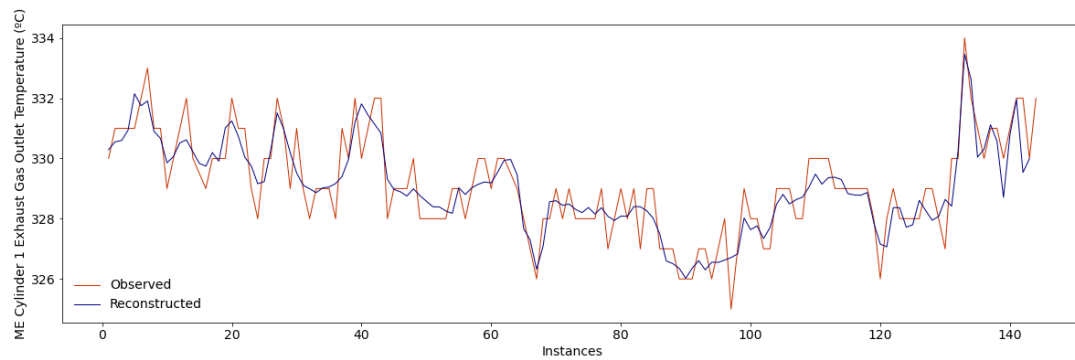
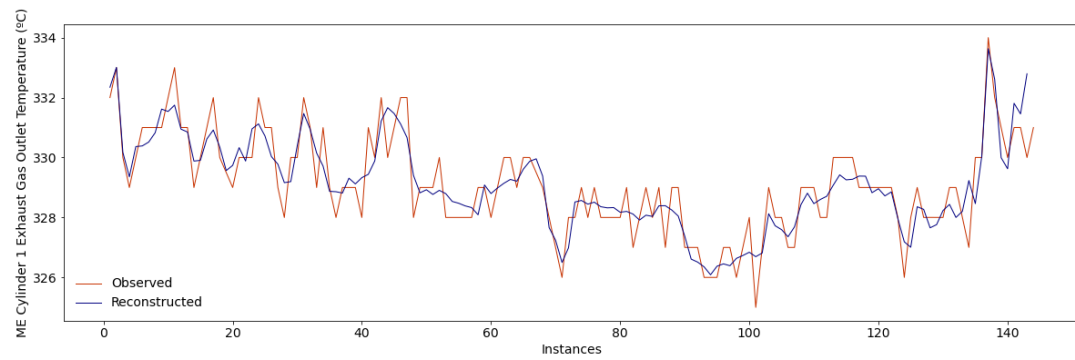
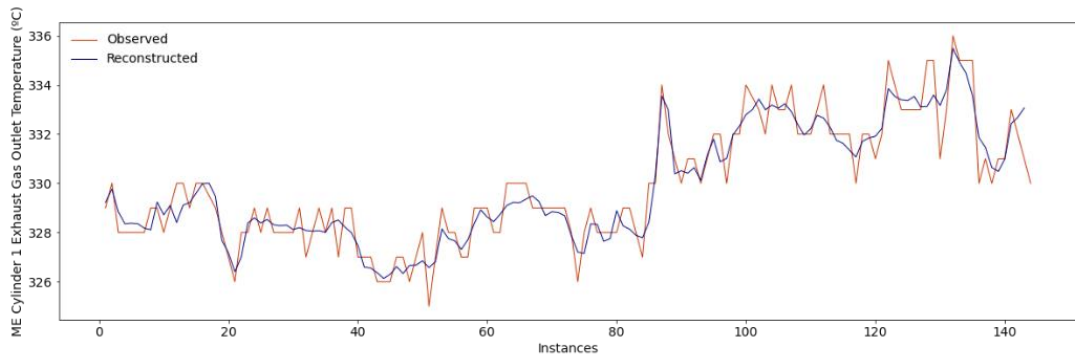
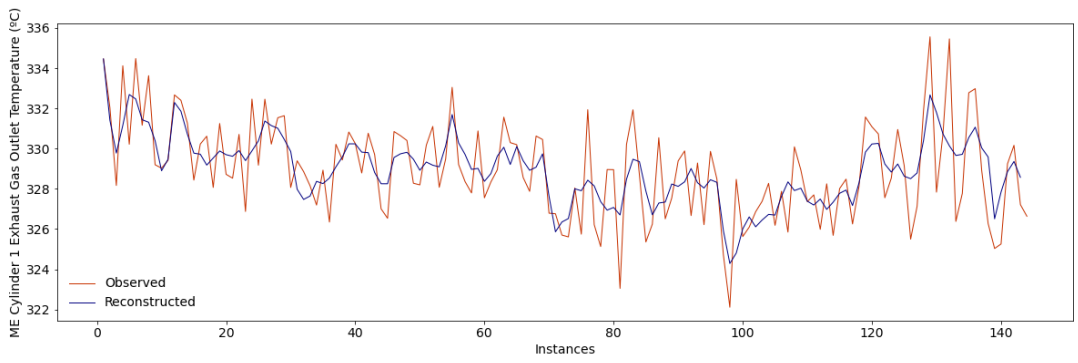
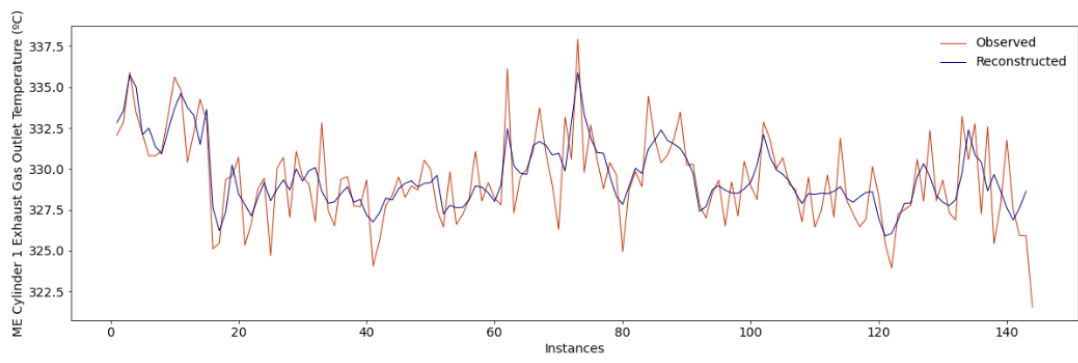
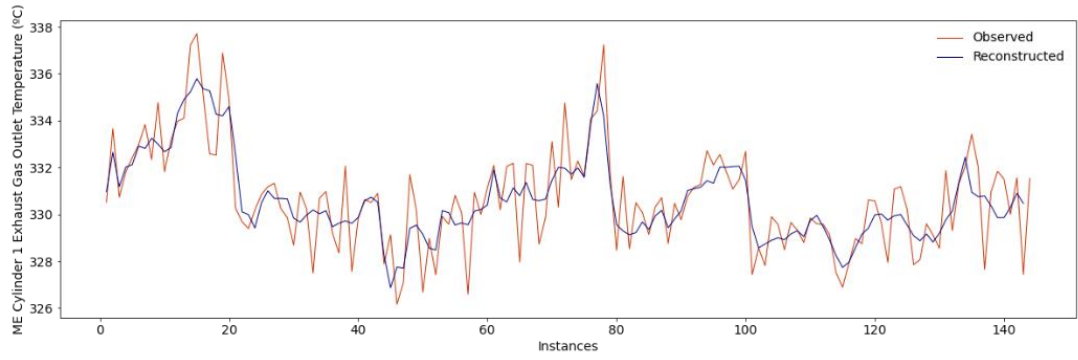


Fig. D.2.9. Example of normal reconstructed sequences (test set).



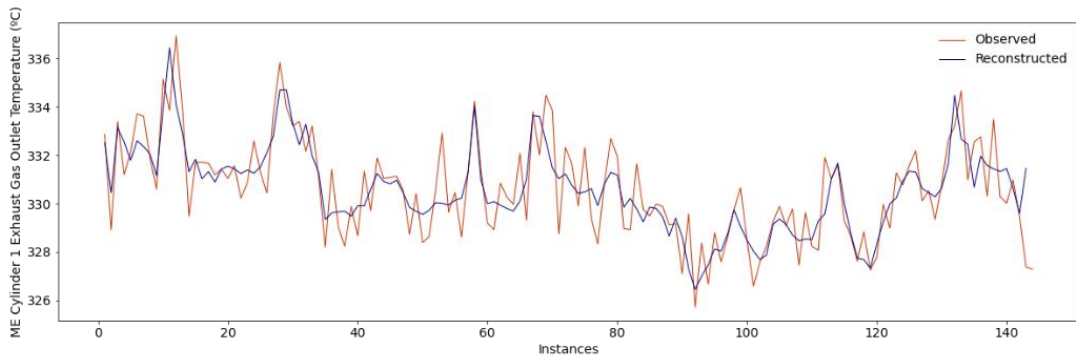
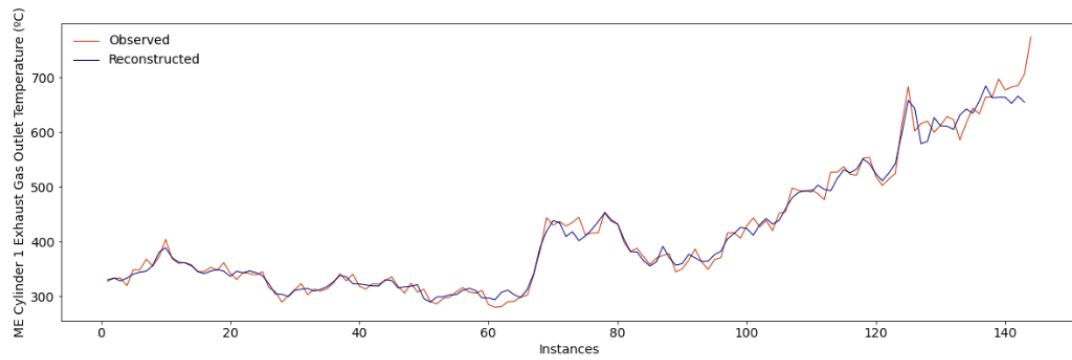
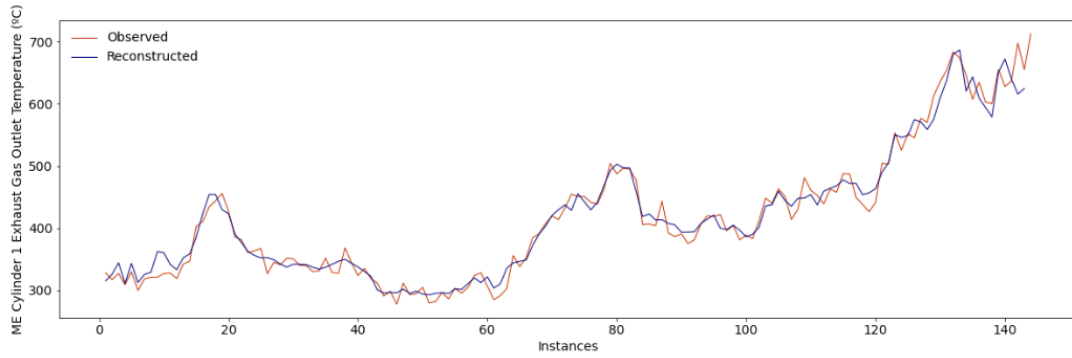


Fig. D.2.10. Example of reconstructed sequences with collective anomalies.



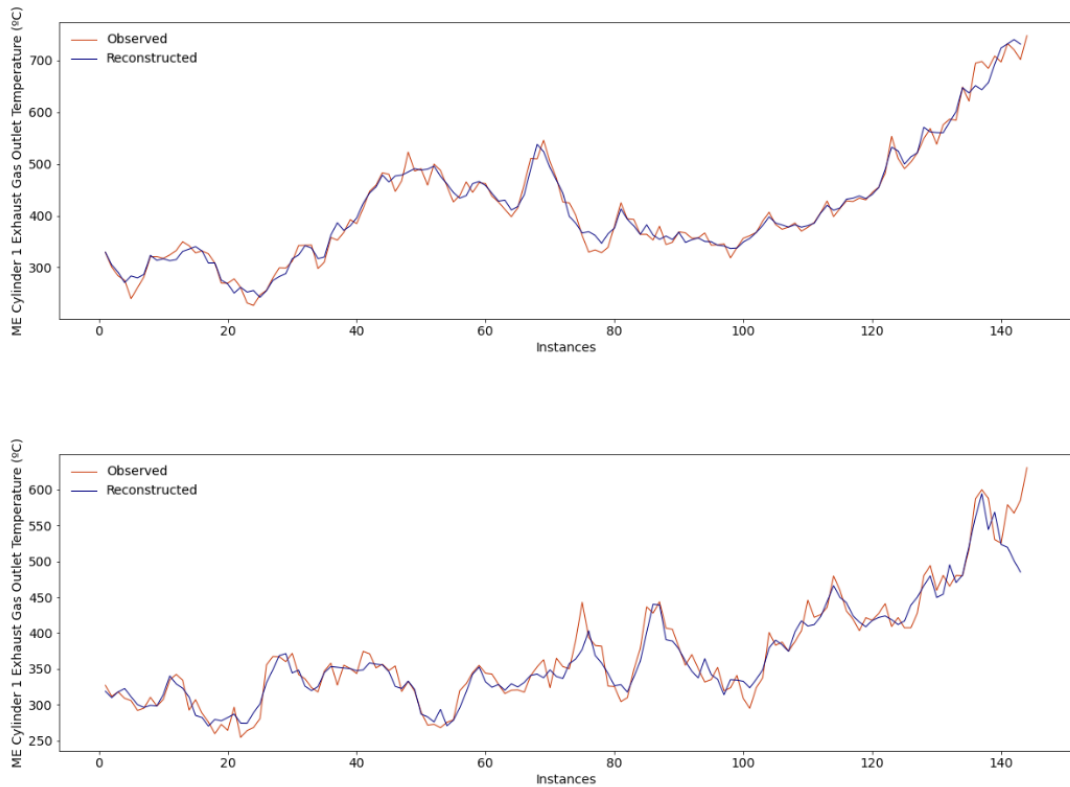


Fig. D.2.11. Example of reconstructed sequences with degradation patterns.

Those sequences detected as anomalous are then classified into two categories: sequences with collective anomalies, and sequences with degradation patterns. Accordingly, as the implemented approach refers to a time series imaging approach (see section 5.3.2. *Fault Identification*), the anomalous sequences detected are transformed into images. Examples of these can be perceived in Figs. E.2.12 – E.2.13.



Fig. D.2.12. Images with collective anomalies.

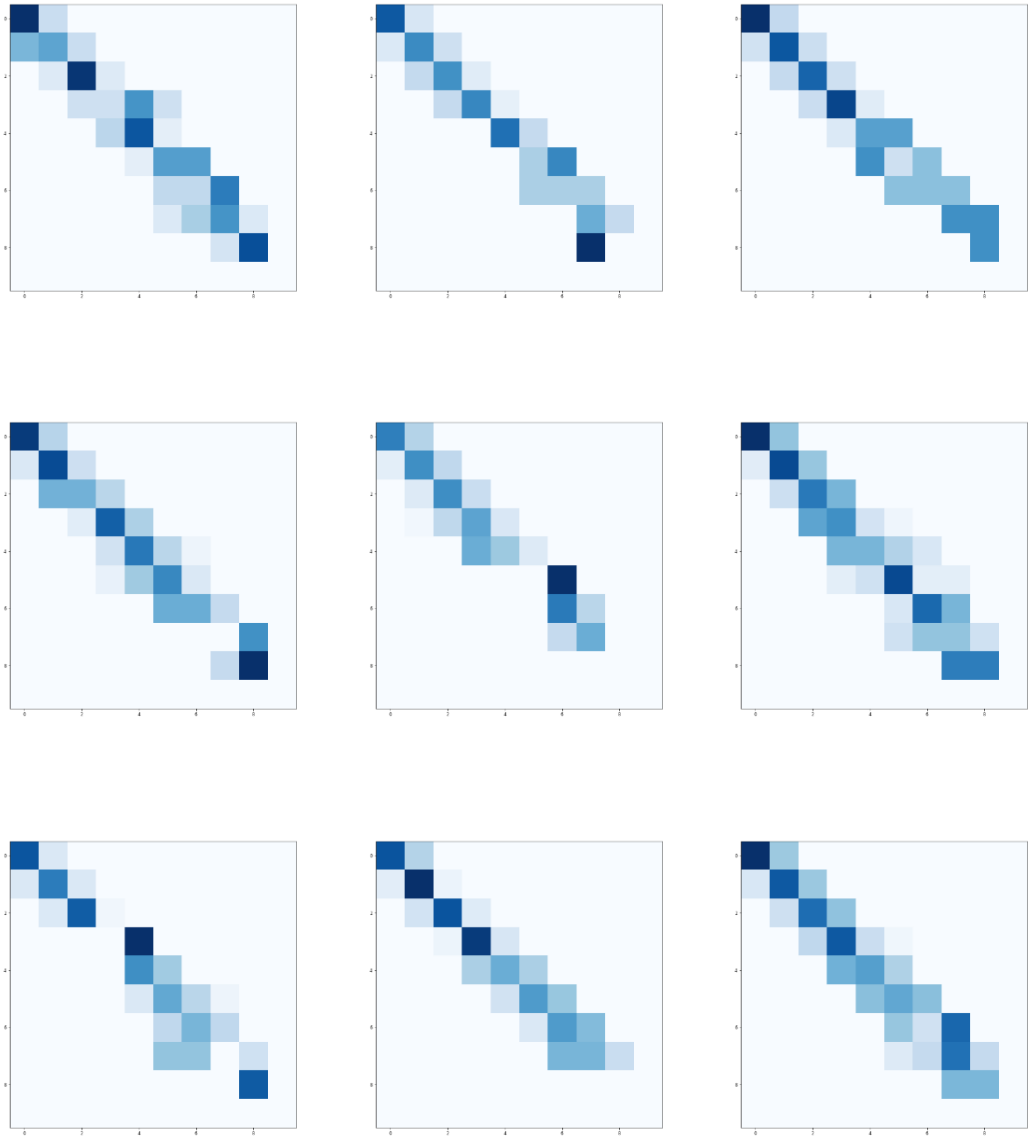
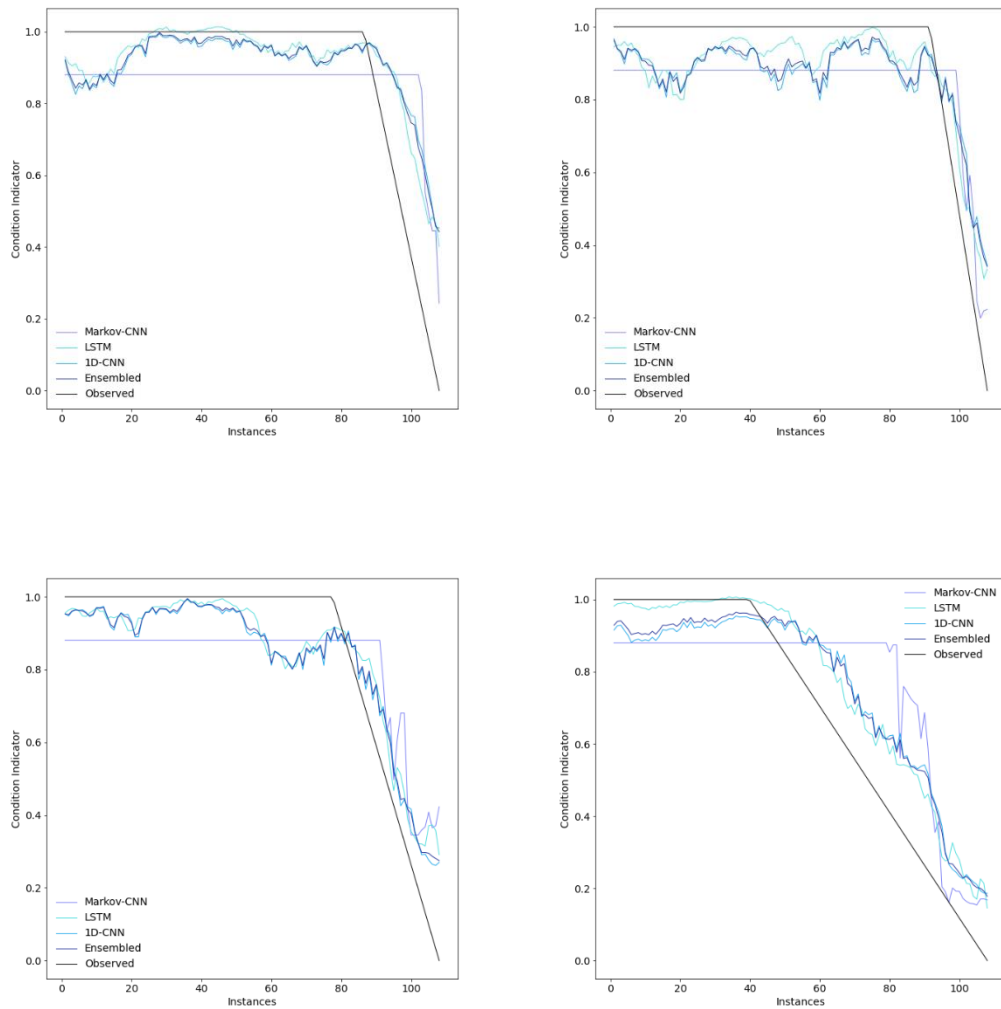


Fig. D.2.13. Images with degradation patterns.

As it can be perceived, the two categories can be easily distinguished. This aspect facilitated the achievement of the maximum accuracy score in this process, thus classifying all the images adequately.

By adequately classifying such images, the sequences with degradation patterns are selected so that the RUL can be predicted. Examples of such a prediction are presented in Fig. D.2.14. The RMSE and Maintenance Score of the first 100 sequences are also presented in Table E.2.2.



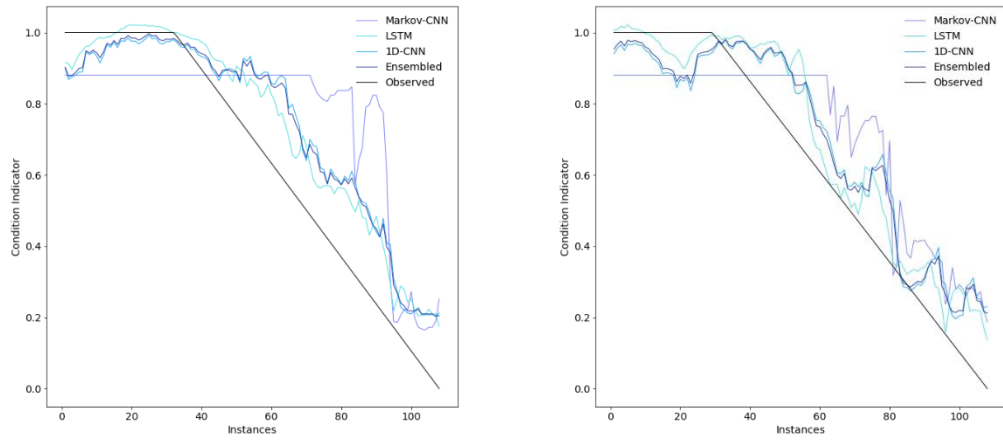


Fig. D.2.14. Examples of condition indicator.

Table D.2.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.

Sequences	RMSE (Minutes)				Maintenance Score			
	Markov-CNN	LSTM	1D-CNN	Ensembled	Markov-CNN	LSTM	1D-CNN	Ensembled
Sequence 1	20.58	15.51	17.65	16.94	1.37	0.75	0.98	0.93
Sequence 2	26.30	19.36	22.73	21.89	1.50	0.82	1.20	1.13
Sequence 3	53.17	50.04	53.55	52.69	2.77	2.68	2.73	2.72
Sequence 4	29.95	24.38	28.50	27.45	1.74	1.00	1.24	1.19
Sequence 5	35.40	40.85	40.15	39.79	1.73	2.14	2.13	2.12
Sequence 6	33.19	22.71	27.10	26.06	1.80	0.94	1.38	1.29
Sequence 7	56.05	50.54	52.24	51.60	3.64	2.77	3.04	2.98
Sequence 8	53.03	38.72	40.64	40.16	2.55	1.82	1.96	1.92
Sequence 9	50.55	34.32	33.59	33.43	3.06	1.94	1.90	1.90
Sequence 10	41.66	19.94	24.75	23.60	2.01	0.89	1.13	1.08
Sequence 11	50.23	59.25	56.78	57.14	3.31	4.11	3.90	3.94
Sequence 12	32.89	23.46	26.68	25.83	1.74	1.01	1.20	1.16
Sequence 13	26.61	12.61	13.81	13.33	1.38	0.59	0.67	0.65
Sequence 14	22.70	14.23	15.40	14.95	1.15	0.65	0.71	0.69
Sequence 15	16.55	20.04	24.54	23.30	1.20	1.16	1.60	1.51
Sequence 16	23.99	32.49	34.62	33.81	1.48	1.66	1.86	1.78
Sequence 17	48.50	30.00	31.35	30.84	2.24	1.44	1.50	1.47
Sequence 18	55.45	49.83	49.57	49.50	2.70	2.31	2.27	2.27
Sequence 19	19.54	15.77	18.46	17.68	1.31	0.82	0.91	0.89
Sequence 20	54.29	35.55	40.65	39.43	3.21	1.80	2.26	2.16
Sequence 21	70.71	53.93	52.49	52.39	4.92	3.47	3.38	3.40
Sequence 22	72.13	52.35	52.91	52.62	4.44	3.14	3.15	3.15
Sequence 23	23.57	14.40	17.62	16.75	1.22	0.65	0.83	0.79

Sequence 24	19.02	19.93	20.09	19.67	1.23	1.18	1.19	1.17
Sequence 25	40.93	55.93	53.46	53.86	2.57	3.51	3.36	3.39
Sequence 26	89.34	86.70	90.90	89.95	5.09	4.55	4.78	4.72
Sequence 27	19.57	14.37	18.79	17.49	1.21	0.71	0.97	0.91
Sequence 28	35.07	24.77	25.70	25.24	1.86	1.22	1.33	1.31
Sequence 29	33.89	20.59	22.73	22.10	1.67	0.78	0.92	0.89
Sequence 30	37.26	24.23	23.59	23.56	1.91	1.16	1.21	1.19
Sequence 31	46.62	33.71	35.08	34.61	2.22	1.38	1.65	1.59
Sequence 32	22.40	24.23	29.09	27.82	0.96	1.24	1.43	1.38
Sequence 33	25.37	15.29	19.59	18.57	1.49	0.63	0.76	0.72
Sequence 34	20.27	20.98	23.30	22.46	1.43	1.27	1.44	1.40
Sequence 35	25.08	17.17	24.30	22.80	1.49	0.67	0.95	0.89
Sequence 36	35.58	24.29	27.81	26.89	1.78	1.20	1.42	1.38
Sequence 37	28.09	18.91	22.66	21.66	1.42	0.95	1.15	1.10
Sequence 38	39.09	46.93	48.39	47.86	2.37	2.95	2.96	2.95
Sequence 39	50.81	34.23	36.04	35.33	3.08	1.84	2.06	2.01
Sequence 40	56.00	31.25	31.81	31.46	3.03	1.48	1.59	1.57
Sequence 41	31.03	17.30	20.95	20.01	1.59	0.80	1.16	1.08
Sequence 42	33.89	44.76	44.15	44.16	1.63	2.28	2.24	2.25
Sequence 43	34.94	21.49	21.93	21.55	1.77	0.95	1.09	1.06
Sequence 44	58.05	62.71	62.66	62.46	2.98	2.85	2.83	2.82
Sequence 45	34.16	32.93	36.34	35.40	2.27	1.86	2.31	2.22
Sequence 46	41.27	30.47	27.72	28.03	1.90	1.46	1.39	1.40
Sequence 47	33.50	32.45	34.62	34.02	2.06	1.67	1.89	1.84
Sequence 48	36.49	18.13	22.34	21.36	1.81	0.90	1.12	1.06
Sequence 49	51.94	37.50	40.28	39.45	2.21	1.87	2.05	1.99

Sequence 50	19.96	17.91	17.22	16.98	1.31	0.95	0.91	0.92
Sequence 51	30.63	17.68	19.92	19.30	1.64	0.77	1.04	0.99
Sequence 52	28.13	20.05	21.00	20.55	1.56	0.96	1.04	1.02
Sequence 53	24.70	18.85	20.84	20.04	1.62	0.93	1.17	1.12
Sequence 54	21.98	20.30	22.69	22.04	1.17	0.83	0.94	0.91
Sequence 55	45.50	57.09	57.20	57.03	2.19	2.72	2.76	2.74
Sequence 56	73.08	37.15	37.57	36.96	4.33	1.83	1.98	1.94
Sequence 57	62.51	32.65	34.73	34.10	3.68	1.66	1.82	1.78
Sequence 58	33.76	26.24	24.84	24.99	1.59	1.32	1.23	1.25
Sequence 59	70.83	65.65	64.20	64.35	4.40	4.22	4.16	4.18
Sequence 60	22.12	15.73	15.12	15.11	1.38	0.82	0.87	0.86
Sequence 61	34.00	17.96	19.36	18.95	1.77	0.89	1.06	1.02
Sequence 62	22.56	15.04	16.77	16.24	1.39	0.70	0.81	0.79
Sequence 63	24.13	16.54	19.15	18.33	1.26	0.76	0.91	0.86
Sequence 64	66.90	32.97	40.41	38.68	3.77	1.66	2.15	2.05
Sequence 65	24.50	56.74	52.33	52.88	1.67	4.10	3.66	3.75
Sequence 66	34.08	25.38	29.45	28.54	1.80	0.99	1.35	1.27
Sequence 67	46.20	47.31	45.09	45.24	3.07	3.11	2.97	2.99
Sequence 68	19.29	15.01	18.95	17.98	1.32	0.87	1.16	1.10
Sequence 69	44.08	34.15	39.29	37.98	3.02	2.19	2.56	2.48
Sequence 70	39.51	42.59	43.15	42.90	2.50	2.67	2.68	2.67
Sequence 71	75.65	72.34	71.78	71.82	4.25	3.89	3.75	3.78
Sequence 72	29.65	17.92	21.55	20.74	1.63	0.76	1.03	0.98
Sequence 73	21.19	14.65	17.82	16.99	1.13	0.68	0.85	0.80
Sequence 74	52.22	56.46	56.39	56.20	2.86	2.89	2.84	2.84
Sequence 75	43.35	25.03	29.78	28.52	2.44	1.35	1.66	1.59

Sequence 76	29.00	20.22	23.53	22.65	1.59	0.95	1.18	1.14
Sequence 77	19.25	19.22	19.44	19.14	1.35	1.05	0.98	0.99
Sequence 78	50.75	42.94	39.55	40.07	3.13	2.36	2.18	2.21
Sequence 79	33.58	28.33	29.22	28.60	1.76	1.42	1.39	1.40
Sequence 80	36.84	32.40	34.95	34.23	1.91	1.57	1.68	1.65
Sequence 81	40.31	30.48	34.13	33.16	2.27	1.48	1.65	1.60
Sequence 82	90.87	74.72	74.81	74.65	5.30	3.97	3.93	3.93
Sequence 83	36.61	21.62	25.60	24.68	1.82	0.80	1.19	1.11
Sequence 84	21.76	14.72	19.63	18.26	1.31	0.77	1.08	1.01
Sequence 85	42.71	40.69	39.39	39.25	2.71	2.01	1.96	1.94
Sequence 86	39.91	32.40	31.47	31.44	1.95	1.54	1.53	1.53
Sequence 87	23.56	25.87	24.64	24.56	1.68	1.33	1.30	1.30
Sequence 88	39.58	23.14	24.44	23.73	1.67	1.30	1.35	1.31
Sequence 89	39.83	16.75	21.92	20.69	2.37	0.82	1.21	1.13
Sequence 90	27.56	19.65	23.32	22.53	1.56	0.82	1.05	1.00
Sequence 91	69.32	63.74	61.59	61.84	4.65	4.07	3.94	3.96
Sequence 92	68.87	63.90	62.43	62.54	3.17	3.27	3.10	3.13
Sequence 93	29.46	24.21	27.43	26.47	1.47	1.17	1.32	1.28
Sequence 94	46.71	45.88	44.55	44.35	2.50	2.34	2.31	2.28
Sequence 95	19.16	17.83	20.91	19.98	1.15	0.83	1.04	0.99
Sequence 96	17.58	15.26	18.16	17.22	1.25	0.66	0.67	0.66
Sequence 97	44.28	28.29	28.20	28.06	2.30	1.54	1.54	1.54
Sequence 98	42.47	29.94	31.27	30.77	2.17	1.58	1.62	1.60
Sequence 99	51.55	25.49	29.44	28.39	2.56	1.34	1.56	1.51
Sequence 100	25.57	16.02	18.79	17.87	1.42	0.74	1.00	0.94
Median	34.08	25.03	27.81	27.45	1.80	1.27	1.39	1.38

D.3. Main Engine Cylinder 2

The parameter analysed refers to the exhaust gas outlet temperature. A graphical representation of such a parameter is expressed in Fig. D.3.1. The descriptive statistics is also introduced in Table E.3.1.

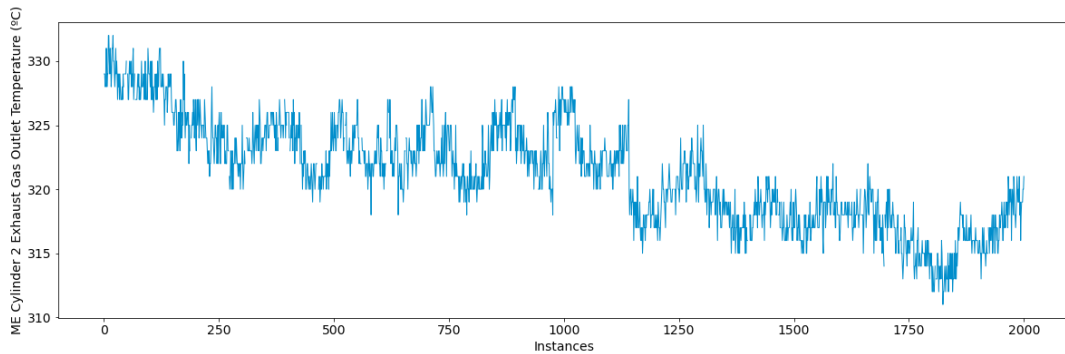


Fig. D.3.1. Graphical representation of the exhaust gas outlet temperature parameter.

Table D.3.1. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
Cyl. 2 Exh. Gas Out. Temp	321.23	4.01	311	318	321	324	332

As part of the data pre-processing phase, the identification of operational states step has been implemented (see section 4.5. *A Novel Framework for the Identification of Steady States* for a comprehensive explanation of such a step). In total, only one operational state has been identified, as perceived in Fig. D.3.2.

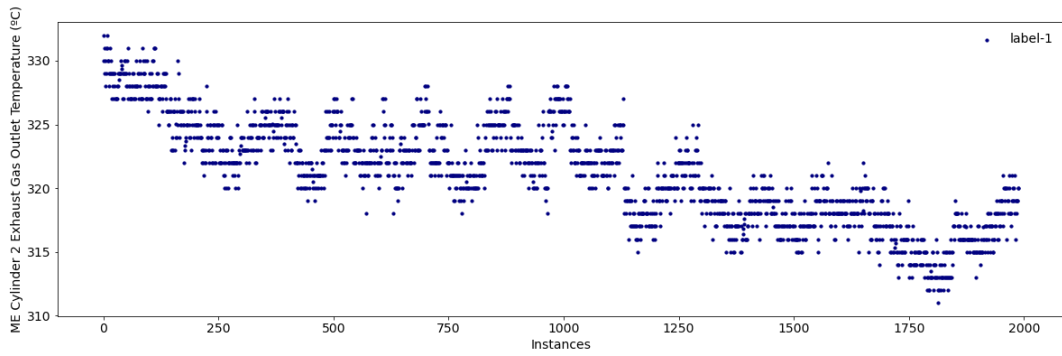
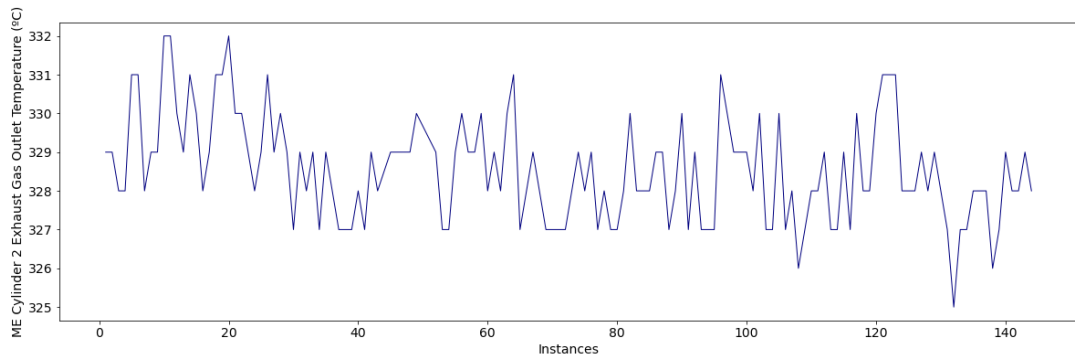


Fig. D.3.2. Identification of the operational states for the monitored parameter.

Subsequently, due to the lack of fault data, both collective anomalies and degradation patterns are simulated. Some examples are presented in Figs. E.3.4 – E.3.5. Examples of normal sequences are also introduced in Fig. D.3.3.



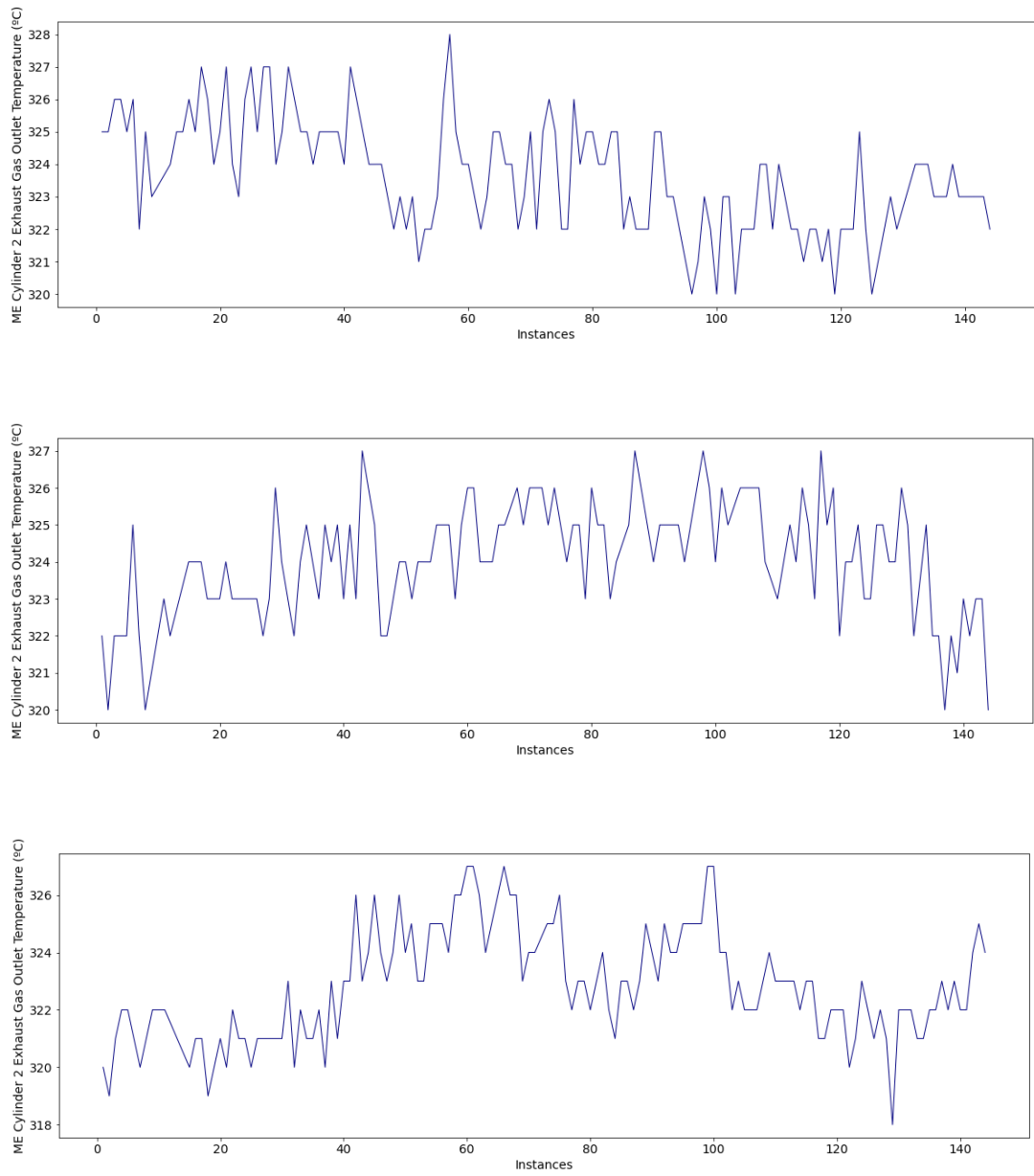


Fig. D.3.3. Example of normal sequences.

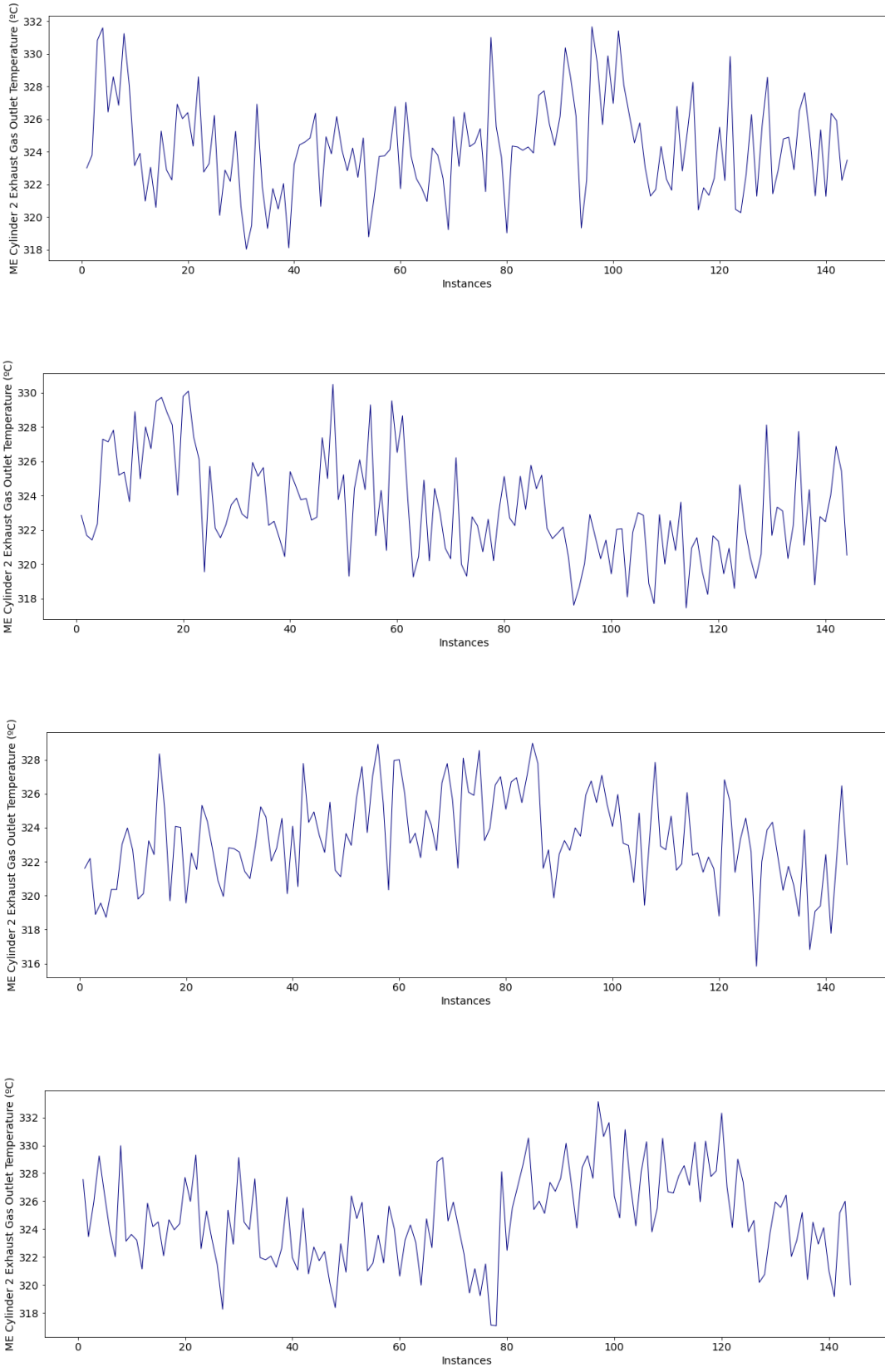
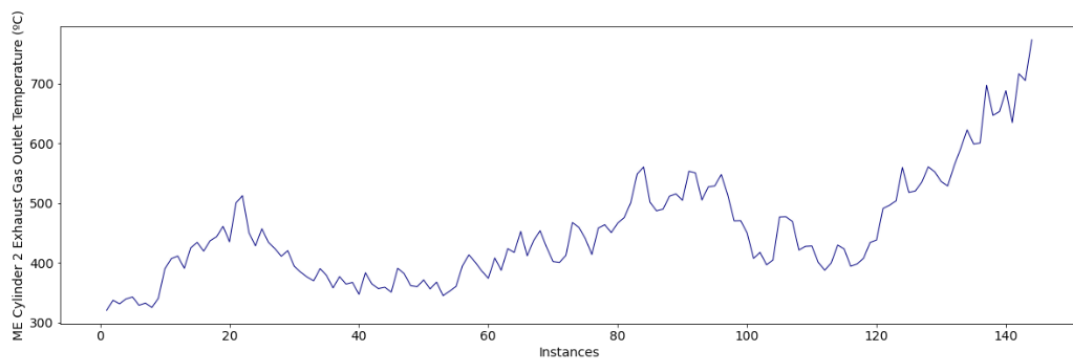
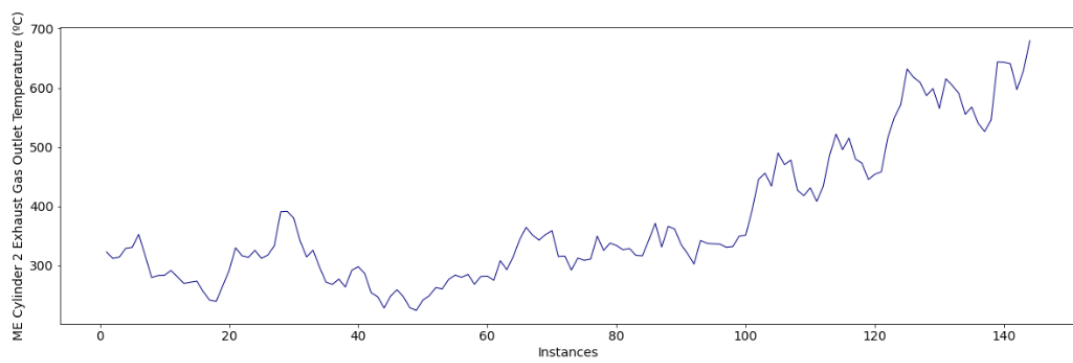
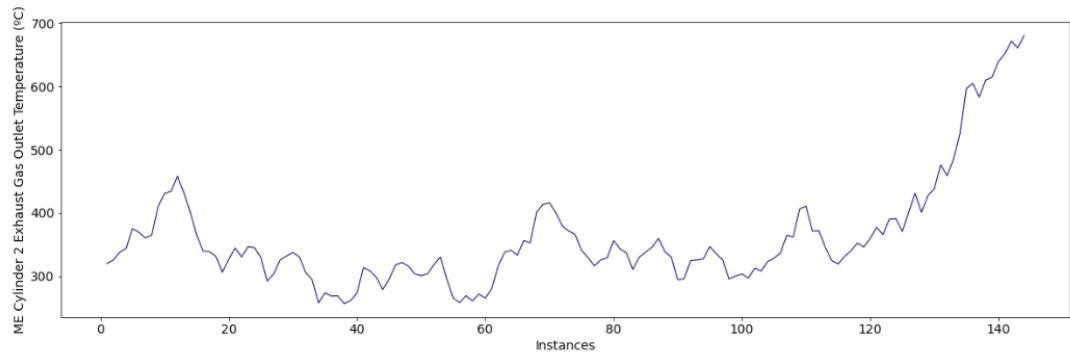


Fig. D.3.4. Example of sequences with collective anomalies.



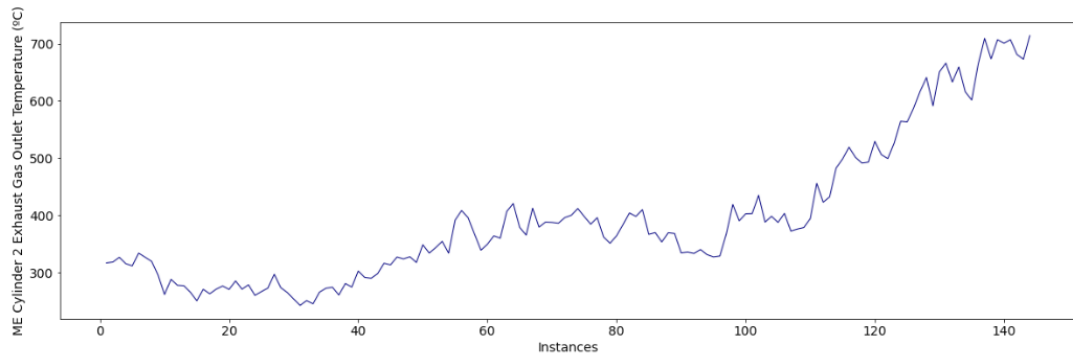


Fig. D.3.5. Example of sequences with degradation patterns.

As part of the MA framework, the subsequent module to be applied is the diagnostic analytics module. Accordingly, the fault detection step is implemented as stated in section 5.3.1. *Fault Detection*. As perceived in the histograms (Figs. E.3.6 – E.3.8), a simple threshold is adequate in this case study to distinguish the normal sequences from the abnormal sequences.

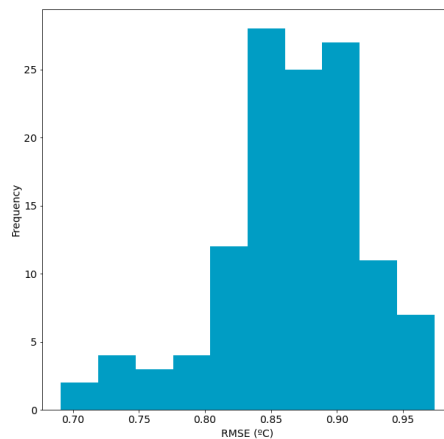


Fig. D.3.6. Histogram of the reconstructed errors of the normal sequences (test set).

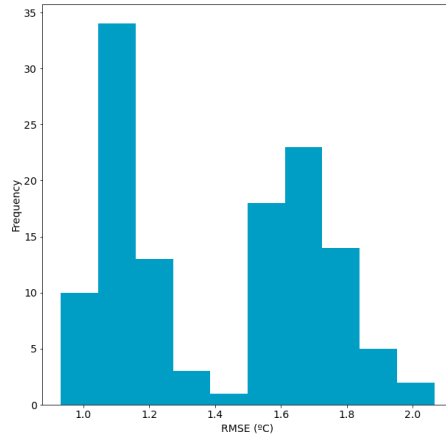


Fig. D.3.7. Histogram of the reconstructed errors of the sequences with collective anomalies.

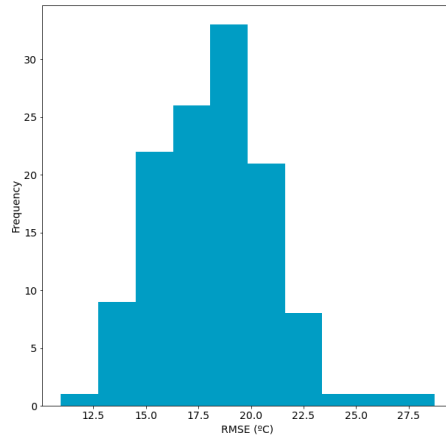


Fig. D.3.8. Histogram of the reconstructed errors of the sequences with degradation patterns.

Examples of reconstructed sequences are also introduced in Figs. E.3.9 – E.3.11.

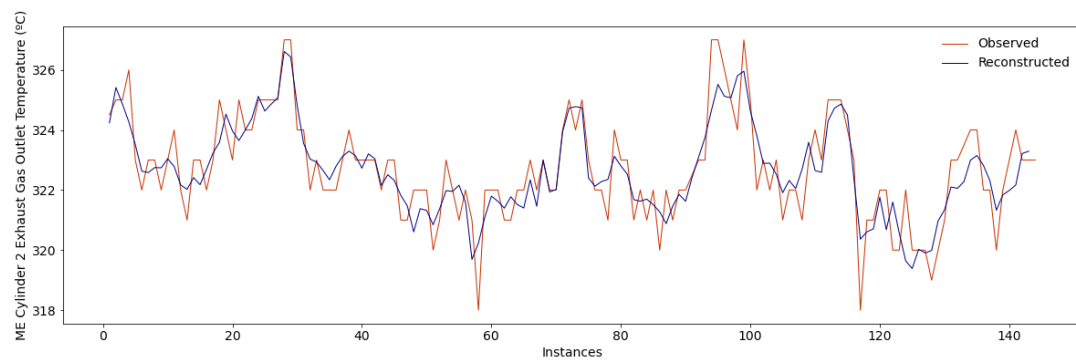
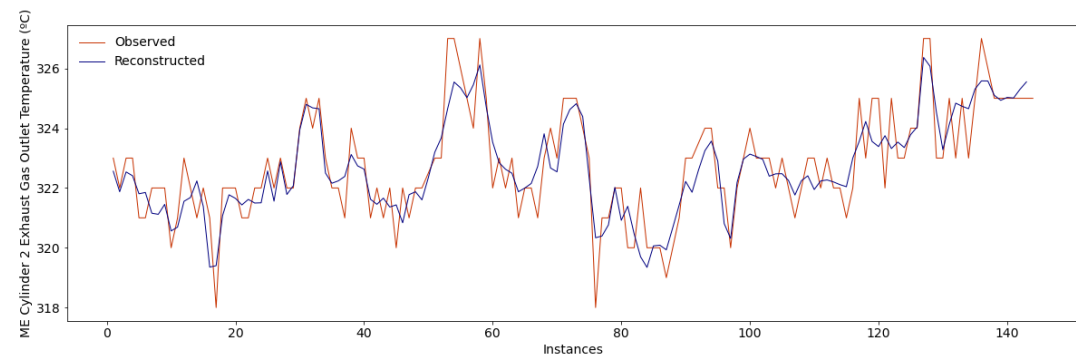
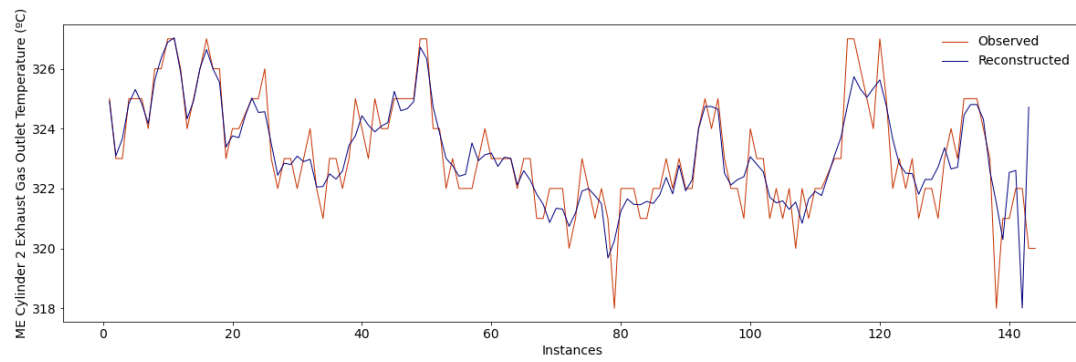
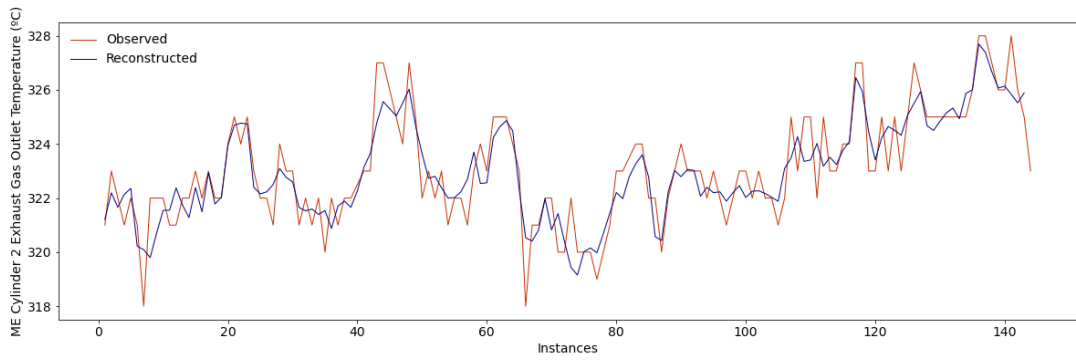
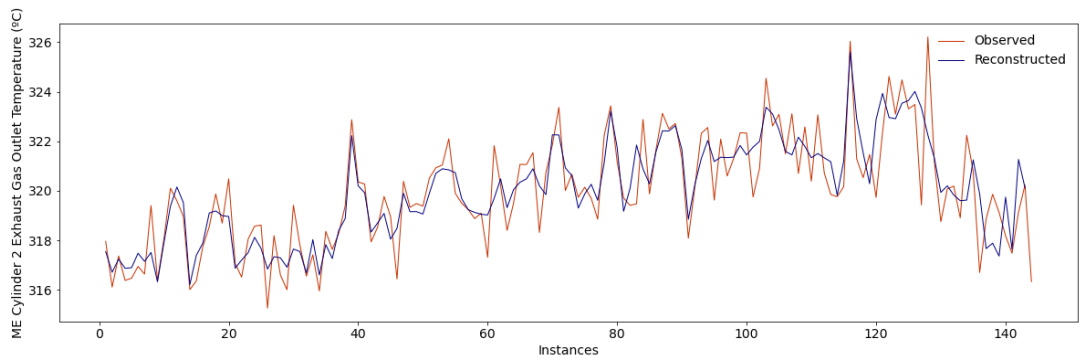
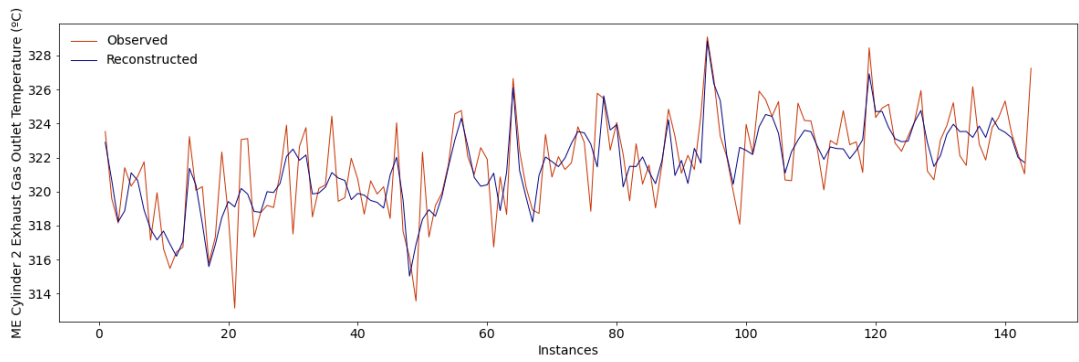
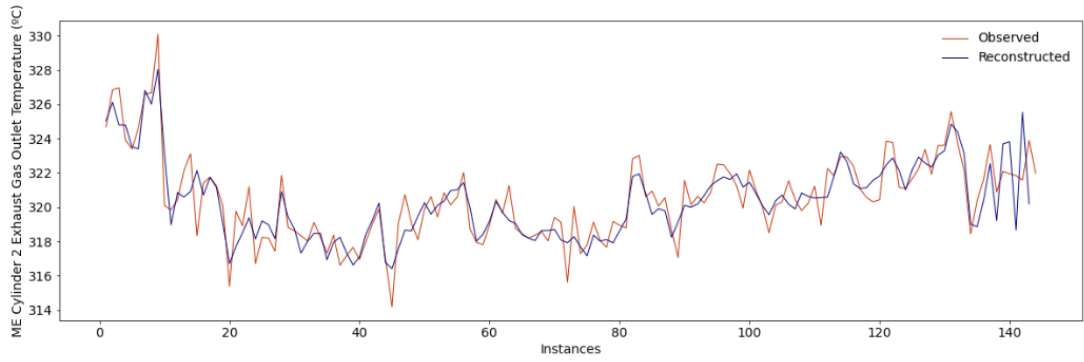


Fig. D.3.9. Example of normal reconstructed sequences (test set).



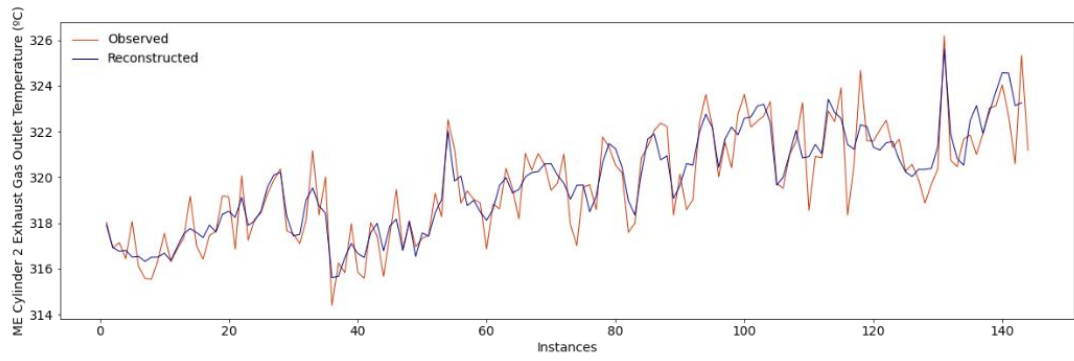
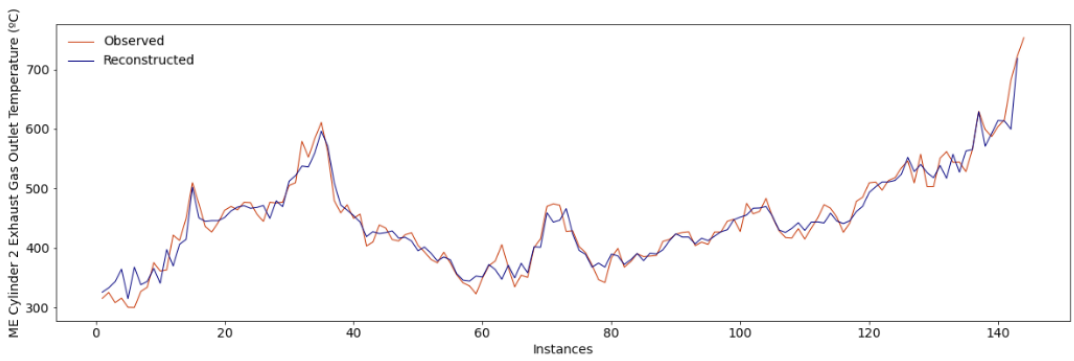
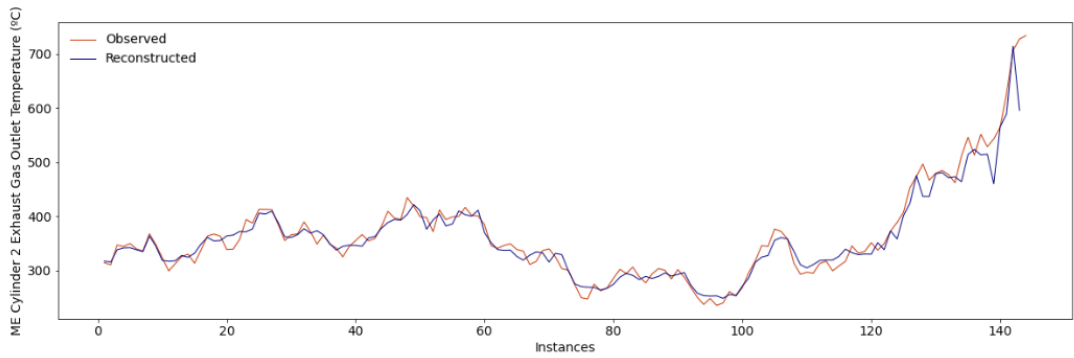


Fig. D.3.10. Example of reconstructed sequences with collective anomalies.



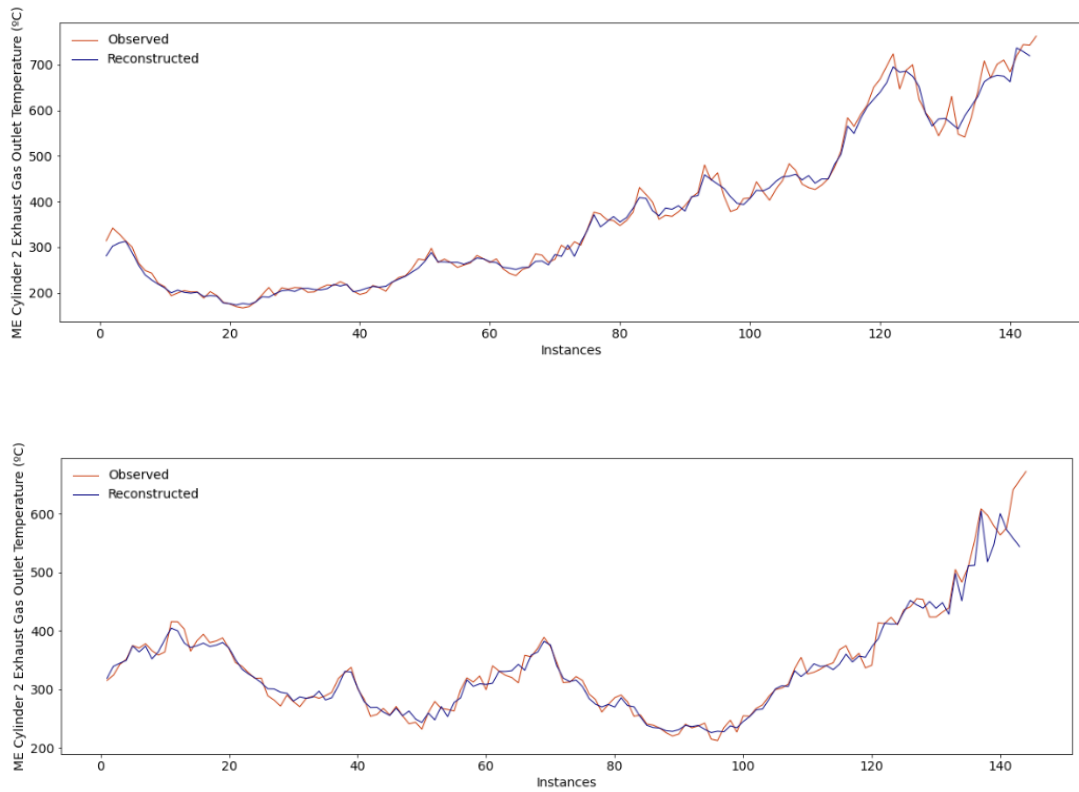


Fig. D.3.11. Example of reconstructed sequences with degradation patterns.

Those sequences detected as anomalous are then classified into two categories: sequences with collective anomalies, and sequences with degradation patterns. Accordingly, as the implemented approach refers to a time series imaging approach (see section 5.3.2. *Fault Identification*), the anomalous sequences detected are transformed into images. Examples of these can be perceived in Figs. E.3.12 – E.3.13.

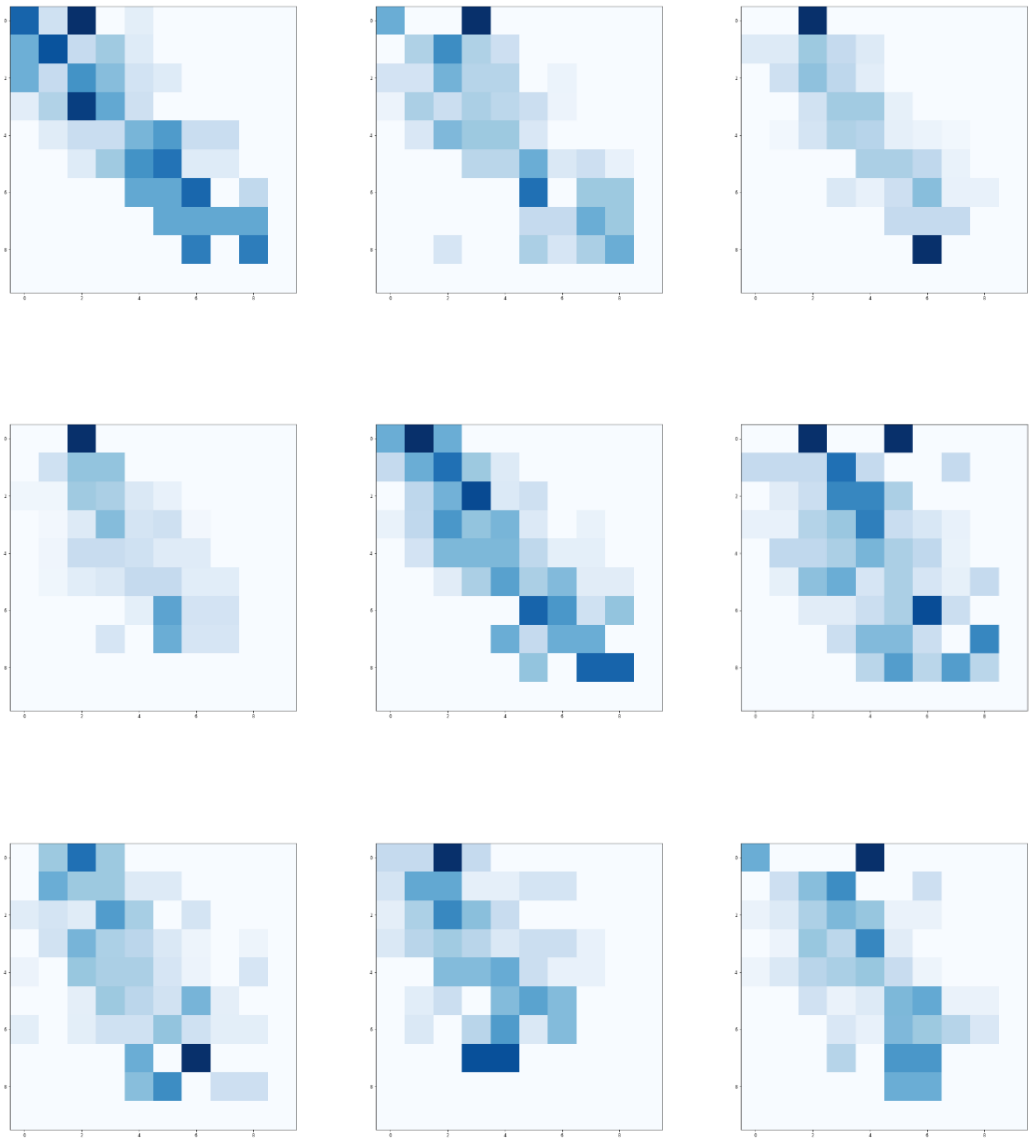


Fig. D.3.12. Images with collective anomalies.

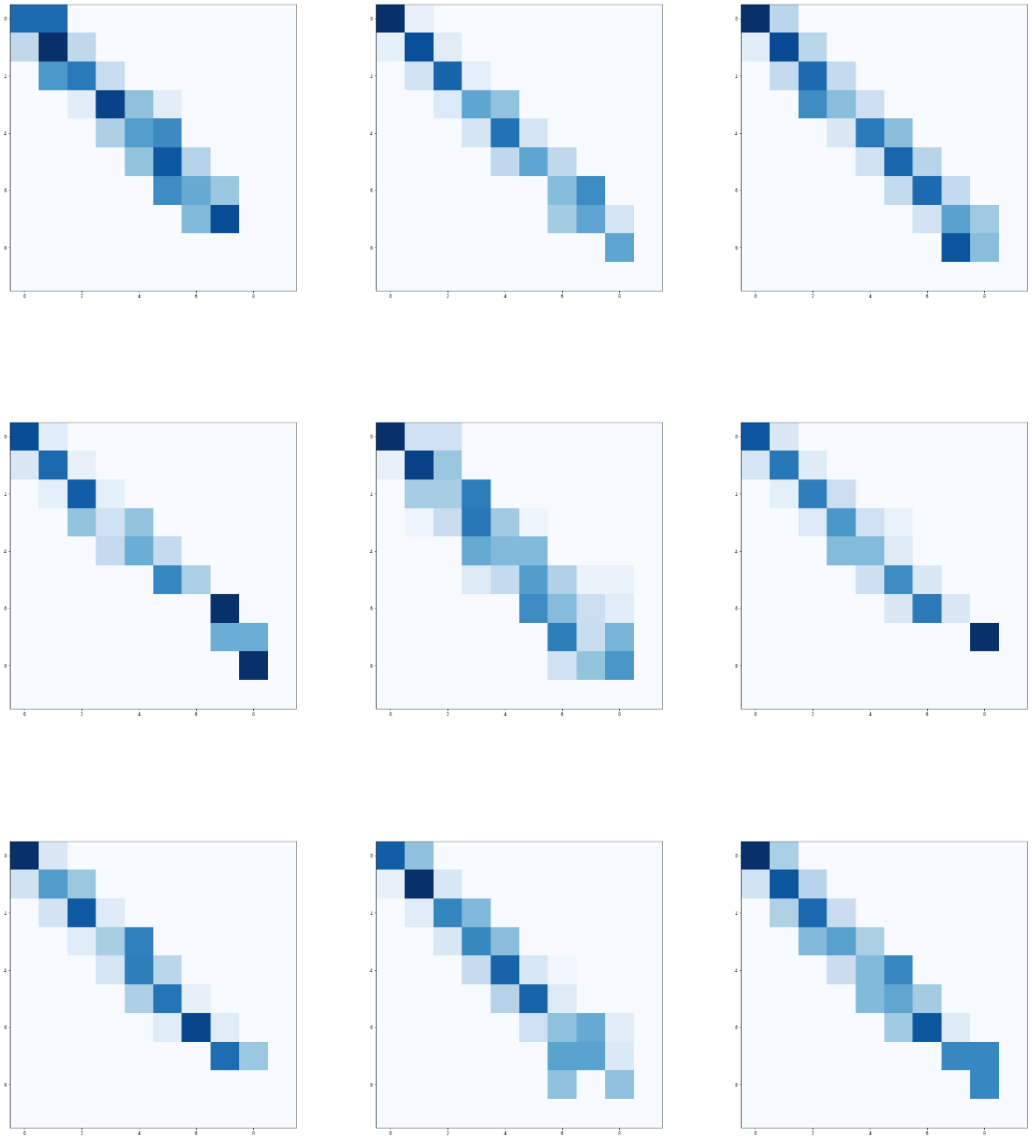
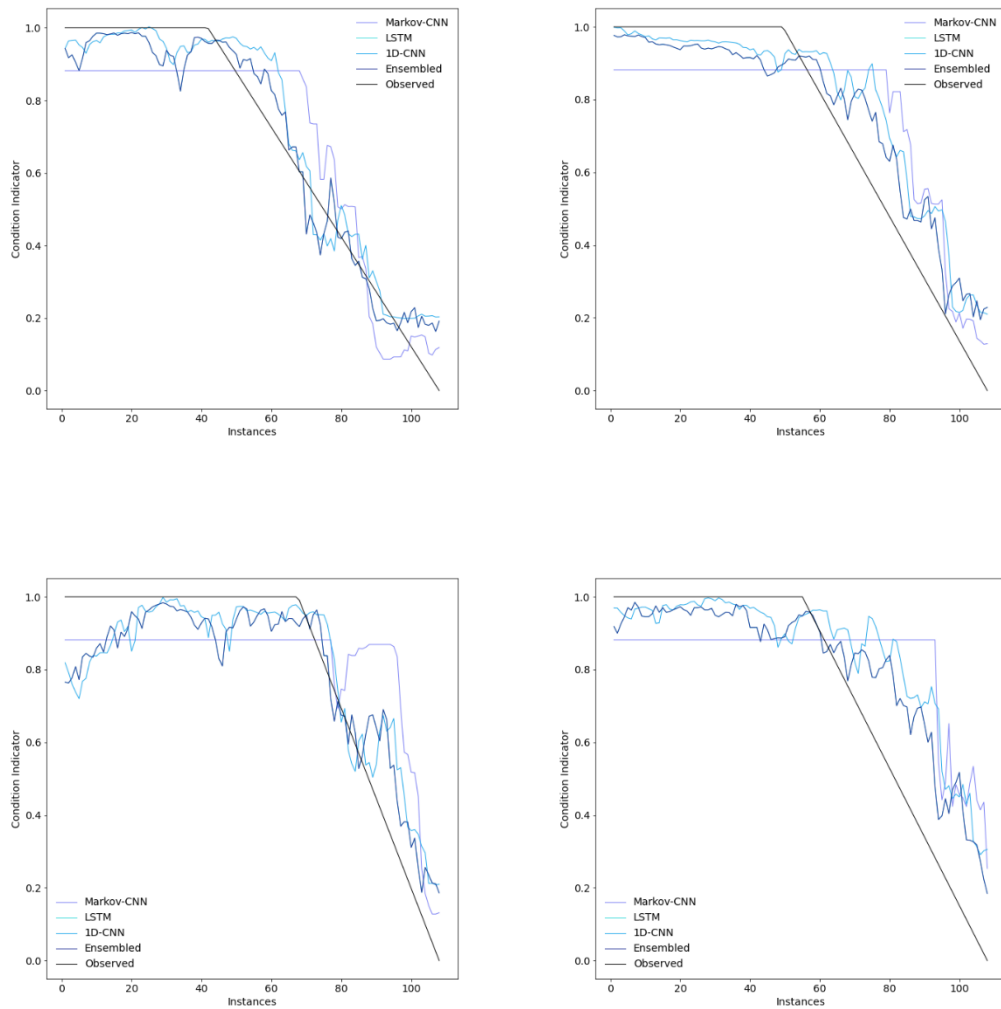


Fig. D.3.13. Images with degradation patterns.

As it can be perceived, the two categories can be easily distinguished. This aspect facilitated the achievement of the maximum accuracy score in this process, thus classifying all the images adequately.

By adequately classifying such images, the sequences with degradation patterns are selected so that the RUL can be predicted. Examples of such a prediction are presented in Fig. D.3.14. The RMSE and Maintenance Score of the first 100 sequences are also presented in Table E.3.2.



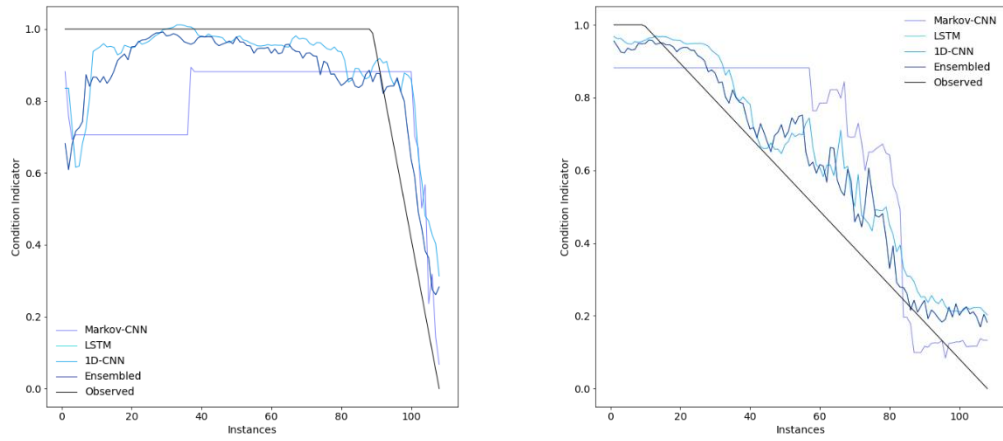


Fig. D.3.14. Examples of condition indicator.

Table D.3.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models

Sequences	RMSE (Minutes)				Maintenance Score			
	Markov-CNN	LSTM	1D-CNN	Ensembled	Markov-CNN	LSTM	1D-CNN	Ensembled
Sequence 1	46.86	43.99	48.29	43.99	2.85	2.70	2.97	2.70
Sequence 2	45.28	20.27	20.62	20.27	2.21	1.04	1.00	1.04
Sequence 3	20.84	10.61	11.16	10.61	1.16	0.62	0.60	0.62
Sequence 4	34.67	21.18	23.72	21.18	1.77	0.93	0.98	0.93
Sequence 5	53.06	51.98	56.36	51.98	3.10	2.78	2.96	2.78
Sequence 6	16.78	17.99	17.53	17.99	1.14	1.00	0.88	1.00
Sequence 7	31.21	42.63	37.17	42.63	2.07	2.82	2.39	2.82
Sequence 8	68.70	64.18	62.26	64.18	4.47	4.21	4.12	4.21
Sequence 9	77.72	68.20	66.17	68.20	3.77	3.85	3.34	3.85
Sequence 10	34.94	50.34	46.74	50.34	2.29	2.91	2.55	2.91
Sequence 11	37.33	22.65	26.66	22.65	2.04	1.03	1.18	1.03
Sequence 12	23.24	19.13	20.52	19.13	1.24	0.92	0.92	0.92
Sequence 13	22.65	20.94	23.04	20.94	1.28	0.98	0.93	0.98
Sequence 14	23.43	19.23	20.76	19.23	1.44	1.11	1.11	1.11
Sequence 15	42.77	48.20	48.23	48.20	2.67	2.91	2.95	2.91
Sequence 16	24.72	24.43	23.99	24.43	1.45	1.55	1.46	1.55
Sequence 17	24.26	27.89	28.07	27.89	1.50	1.53	1.36	1.53
Sequence 18	25.36	16.35	19.63	16.35	1.41	0.88	0.95	0.88
Sequence 19	38.18	24.27	28.28	24.27	2.37	1.28	1.37	1.28
Sequence 20	27.83	17.10	20.33	17.10	1.50	0.91	1.07	0.91
Sequence 21	32.50	46.19	43.80	46.19	1.35	2.31	2.20	2.31
Sequence 22	33.74	46.00	39.91	46.00	2.20	2.91	2.49	2.91
Sequence 23	25.60	19.01	21.01	19.01	1.49	1.12	1.12	1.12

Sequence 24	25.56	19.09	19.08	19.09	1.41	1.25	1.19	1.25
Sequence 25	63.92	63.31	66.65	63.31	2.78	3.00	3.11	3.00
Sequence 26	32.66	17.98	19.38	17.98	1.69	0.93	0.95	0.93
Sequence 27	57.80	42.65	40.15	42.65	3.51	2.78	2.41	2.78
Sequence 28	26.18	18.62	20.02	18.62	1.37	0.87	0.97	0.87
Sequence 29	54.97	42.77	45.86	42.77	2.80	2.02	2.10	2.02
Sequence 30	38.15	22.00	27.07	22.00	1.93	1.08	1.27	1.08
Sequence 31	26.64	25.39	27.03	25.39	1.78	1.52	1.53	1.52
Sequence 32	73.02	61.94	65.35	61.94	4.32	3.65	3.89	3.65
Sequence 33	32.08	18.90	21.98	18.90	1.92	1.01	1.26	1.01
Sequence 34	21.62	11.69	13.61	11.69	1.17	0.58	0.65	0.58
Sequence 35	26.69	15.84	18.37	15.84	1.53	0.92	0.98	0.92
Sequence 36	37.62	24.29	27.49	24.29	1.88	1.18	1.27	1.18
Sequence 37	34.14	36.87	46.72	36.87	1.77	1.88	2.46	1.88
Sequence 38	21.76	19.01	18.10	19.01	1.32	1.12	1.05	1.12
Sequence 39	51.88	38.43	40.78	38.43	2.61	1.94	2.08	1.94
Sequence 40	18.64	17.39	19.16	17.39	1.16	1.02	1.07	1.02
Sequence 41	19.96	32.41	31.22	32.41	1.34	1.98	1.78	1.98
Sequence 42	22.76	25.70	24.79	25.70	1.53	1.53	1.38	1.53
Sequence 43	36.14	22.68	25.93	22.68	1.82	0.99	1.13	0.99
Sequence 44	27.84	18.51	20.59	18.51	1.57	1.03	1.02	1.03
Sequence 45	35.36	21.66	22.75	21.66	1.83	1.12	1.14	1.12
Sequence 46	19.69	26.05	23.17	26.05	1.34	1.54	1.31	1.54
Sequence 47	17.03	20.33	19.26	20.33	1.06	1.25	1.18	1.25
Sequence 48	27.55	19.45	21.75	19.45	1.57	0.91	0.92	0.91
Sequence 49	33.77	27.66	26.39	27.66	2.20	1.65	1.48	1.65

Sequence 50	28.59	26.84	30.10	26.84	1.24	1.44	1.59	1.44
Sequence 51	22.30	15.77	15.80	15.77	1.41	0.89	0.95	0.89
Sequence 52	27.01	18.48	17.82	18.48	1.33	0.89	0.84	0.89
Sequence 53	58.55	45.10	46.04	45.10	2.84	2.18	2.24	2.18
Sequence 54	41.48	31.28	32.01	31.28	2.16	1.46	1.52	1.46
Sequence 55	29.27	25.20	23.96	25.20	1.81	1.33	1.23	1.33
Sequence 56	22.67	21.36	22.58	21.36	1.41	1.26	1.29	1.26
Sequence 57	57.46	48.74	52.08	48.74	2.74	2.11	2.30	2.11
Sequence 58	48.85	38.88	41.10	38.88	3.09	2.48	2.49	2.48
Sequence 59	28.72	22.59	21.49	22.59	1.73	1.31	1.17	1.31
Sequence 60	57.21	50.45	57.75	50.45	2.76	2.58	2.92	2.58
Sequence 61	48.39	53.55	54.29	53.55	2.89	2.93	3.05	2.93
Sequence 62	35.38	25.47	28.03	25.47	1.94	1.23	1.47	1.23
Sequence 63	22.67	34.24	30.94	34.24	1.57	2.12	1.83	2.12
Sequence 64	19.60	18.49	17.50	18.49	1.29	1.10	0.94	1.10
Sequence 65	22.43	19.89	22.02	19.89	1.42	1.17	1.19	1.17
Sequence 66	28.81	33.82	37.56	33.82	1.90	1.94	2.01	1.94
Sequence 67	45.29	14.45	17.30	14.45	2.08	0.76	0.90	0.76
Sequence 68	22.14	18.07	18.42	18.07	1.36	1.05	1.01	1.05
Sequence 69	48.37	62.20	62.59	62.20	2.06	3.24	3.24	3.24
Sequence 70	43.78	32.02	39.04	32.02	2.09	1.43	1.64	1.43
Sequence 71	18.49	10.62	12.09	10.62	1.12	0.57	0.59	0.57
Sequence 72	20.39	15.77	16.43	15.77	1.31	0.91	0.93	0.91
Sequence 73	44.31	42.57	42.05	42.57	2.27	2.02	2.05	2.02
Sequence 74	18.70	24.55	23.67	24.55	1.31	1.65	1.52	1.65
Sequence 75	25.73	18.15	21.47	18.15	1.50	0.87	0.89	0.87

Sequence 76	16.30	24.18	22.44	24.18	1.11	1.33	1.15	1.33
Sequence 77	31.77	19.12	20.34	19.12	1.67	1.10	1.09	1.10
Sequence 78	31.05	42.58	39.78	42.58	2.06	2.68	2.34	2.68
Sequence 79	23.57	18.87	18.86	18.87	1.28	0.88	0.89	0.88
Sequence 80	23.19	16.95	18.01	16.95	1.40	1.07	1.08	1.07
Sequence 81	21.04	21.96	22.26	21.96	1.16	1.00	1.01	1.00
Sequence 82	33.98	23.58	28.24	23.58	1.71	1.21	1.46	1.21
Sequence 83	28.17	25.78	24.14	25.78	1.85	1.76	1.59	1.76
Sequence 84	29.53	33.76	33.18	33.76	2.09	2.21	2.09	2.21
Sequence 85	25.22	18.03	21.17	18.03	1.48	0.97	1.09	0.97
Sequence 86	29.32	23.74	24.09	23.74	1.52	1.26	1.23	1.26
Sequence 87	39.63	25.27	26.90	25.27	2.01	1.16	1.20	1.16
Sequence 88	35.47	20.47	24.44	20.47	1.79	0.88	1.05	0.88
Sequence 89	47.59	31.05	35.70	31.05	2.32	1.40	1.63	1.40
Sequence 90	53.46	20.27	24.74	20.27	2.57	1.05	1.26	1.05
Sequence 91	31.06	19.03	21.68	19.03	2.01	1.04	0.96	1.04
Sequence 92	48.61	24.19	31.31	24.19	2.49	1.28	1.66	1.28
Sequence 93	31.92	20.15	20.36	20.15	1.49	0.94	0.95	0.94
Sequence 94	19.61	24.59	23.99	24.59	1.34	1.57	1.42	1.57
Sequence 95	22.17	13.29	13.75	13.29	1.32	0.73	0.72	0.73
Sequence 96	17.93	20.93	22.25	20.93	1.23	1.23	1.30	1.23
Sequence 97	48.39	51.66	57.08	51.66	2.27	2.71	3.01	2.71
Sequence 98	41.71	34.02	32.54	34.02	2.60	2.10	1.85	2.10
Sequence 99	23.54	17.57	18.06	17.57	1.43	0.95	0.92	0.95
Sequence 100	76.18	46.07	46.23	46.07	4.41	2.59	2.42	2.59
Median	31.03	23.74	24.45	23.74	1.73	1.26	1.28	1.26

D.4. Main Engine Cylinder 3

The parameter analysed refers to the exhaust gas outlet temperature. A graphical representation of such a parameter is expressed in Fig. D.4.1. The descriptive statistics is also introduced in Table E.4.1.

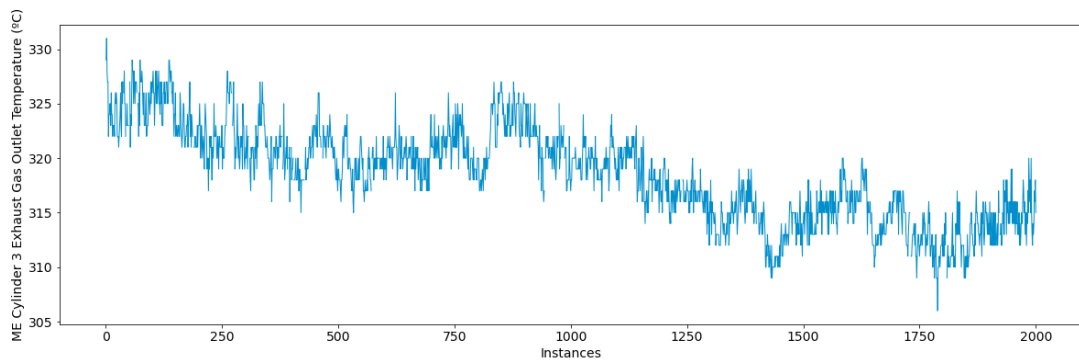


Fig. D.4.1. Graphical representation of the exhaust gas outlet temperature parameter.

Table D.4.1. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
Cyl. 3 Exh. Gas Out. Temp	318.5	4.27	306	315	319	322	331

As part of the data pre-processing phase, the identification of operational states step has been implemented (see section 4.5. *A Novel Framework for the Identification of Steady States* for a comprehensive explanation of such a step). In total, only one operational state has been identified, as perceived in Fig. D.4.2.

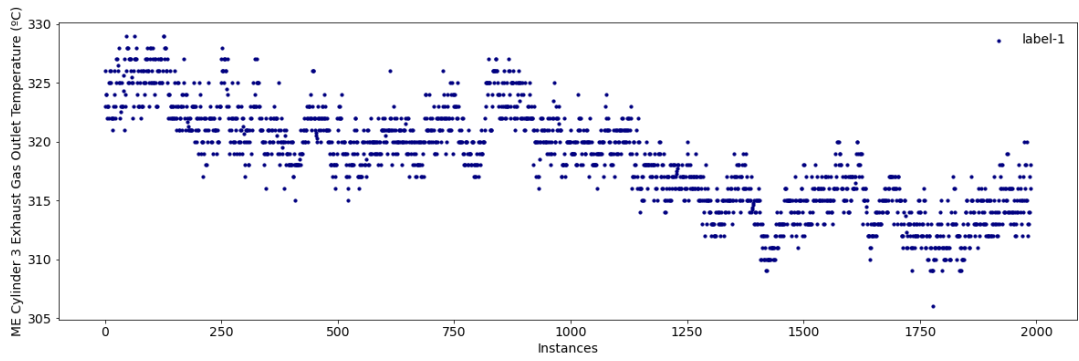
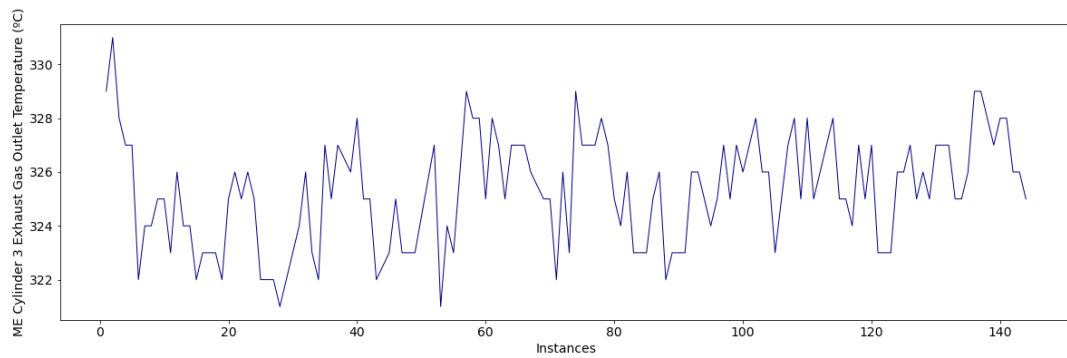


Fig. D.4.2. Identification of the operational states for the monitored parameter.

Subsequently, due to the lack of fault data, both collective anomalies and degradation patterns are simulated. Some examples are presented in Figs. E.4.4 – E.4.5. Examples of normal sequences are also introduced in Fig. D.4.3.



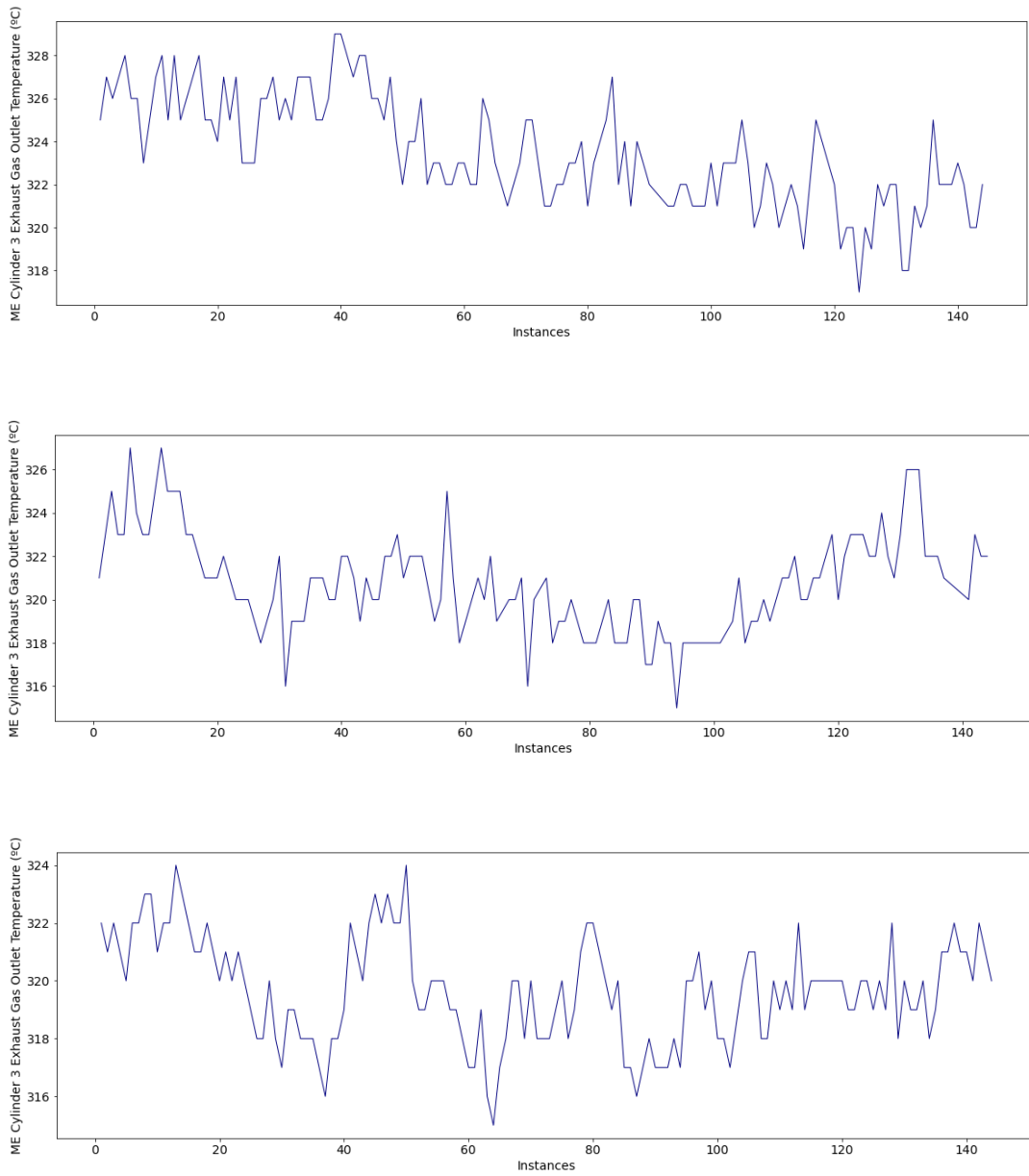


Fig. D.4.3. Example of normal sequences.

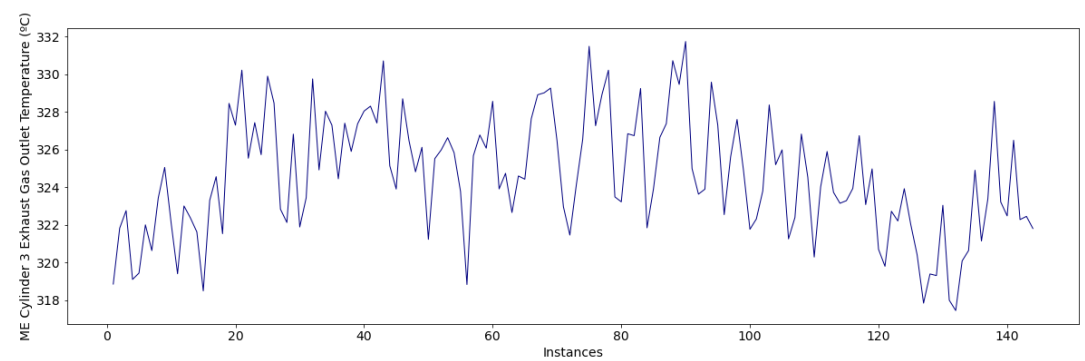
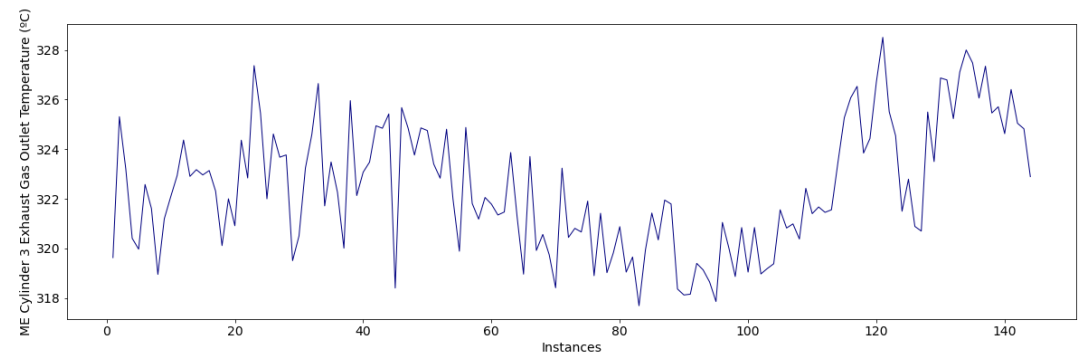
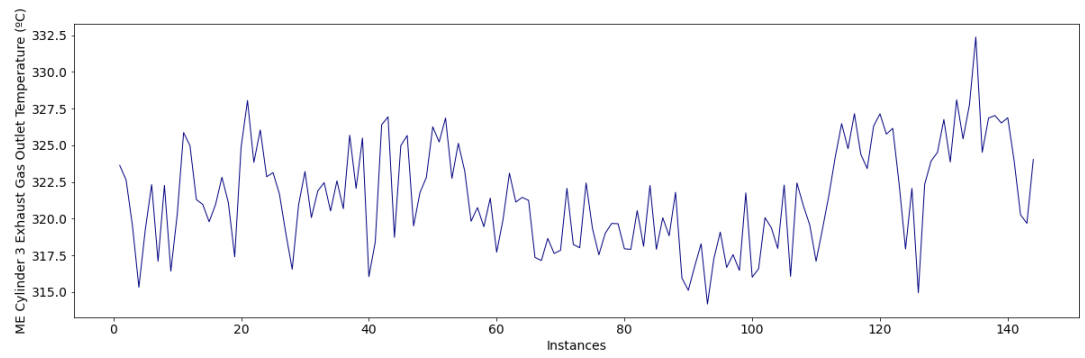
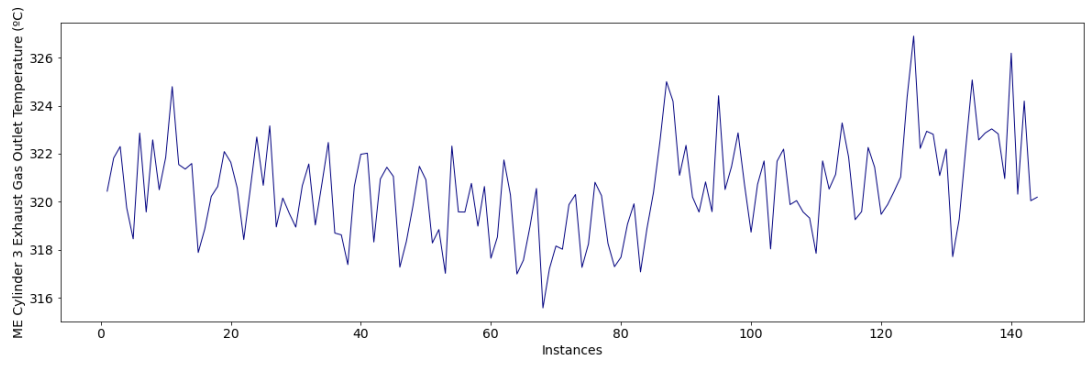


Fig. D.4.4. Example of sequences with collective anomalies.

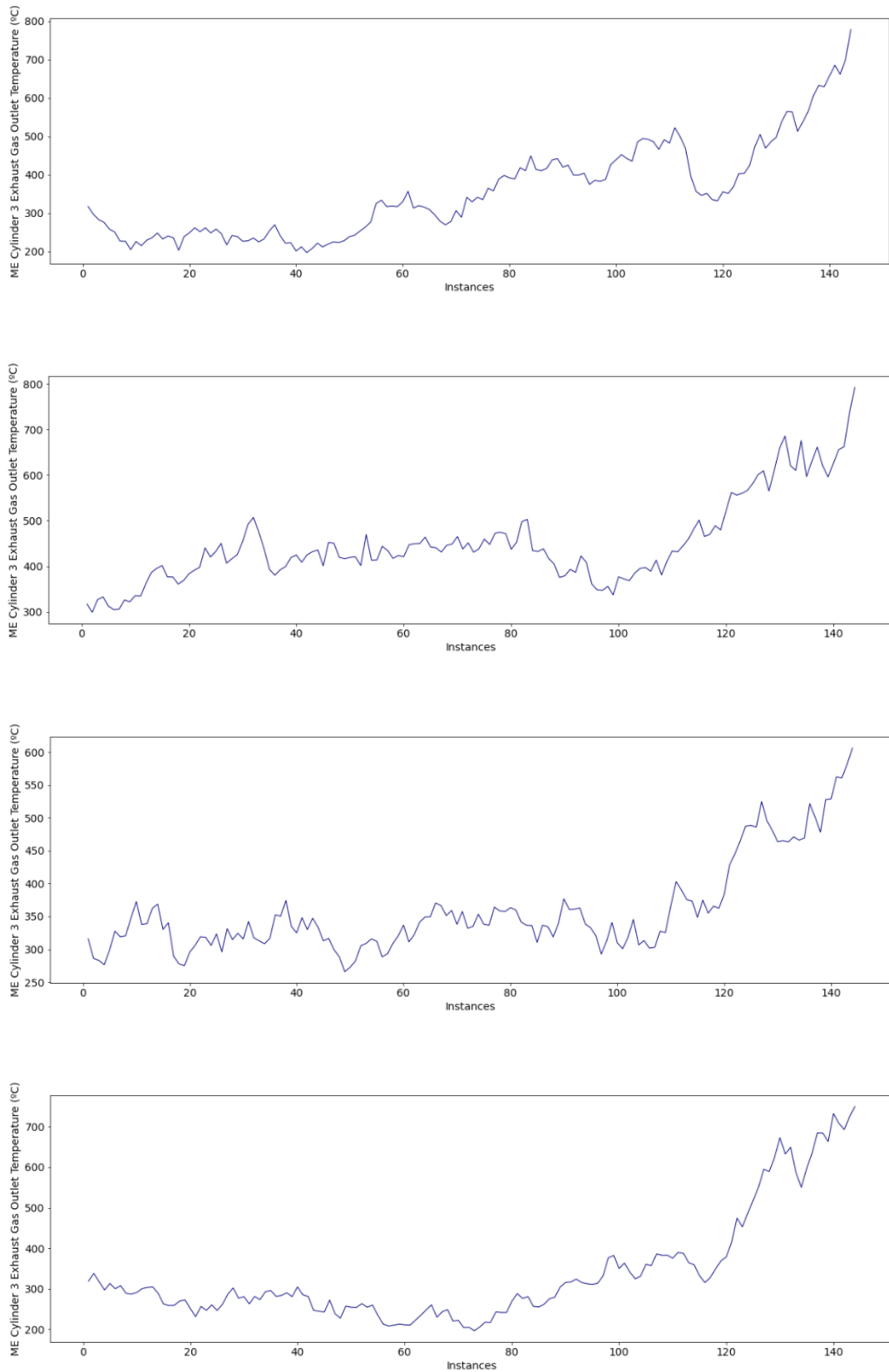


Fig. D.4.5. Example of sequences with degradation patterns.

As part of the MA framework, the subsequent module to be applied is the diagnostic analytics module. Accordingly, the fault detection step is implemented as stated in section 5.3.1. *Fault Detection*. As perceived in the histograms (Figs. E.4.6 – E.4.8), a simple threshold is adequate in this case study to distinguish the normal sequences from the abnormal sequences.

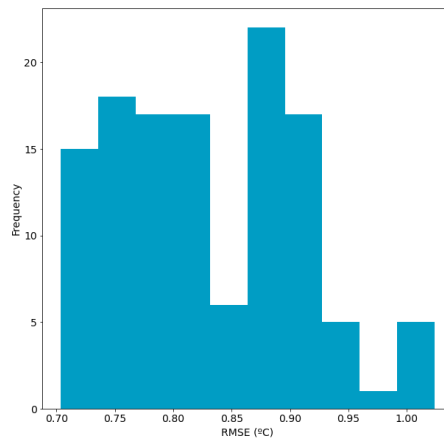


Fig. D.4.6. Histogram of the reconstructed errors of the normal sequences (test set).

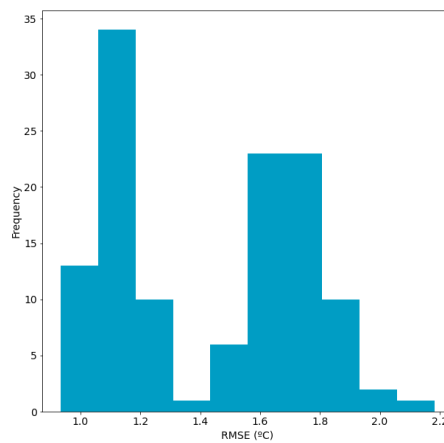


Fig. D.4.7 Histogram of the reconstructed errors of the sequences with collective anomalies.

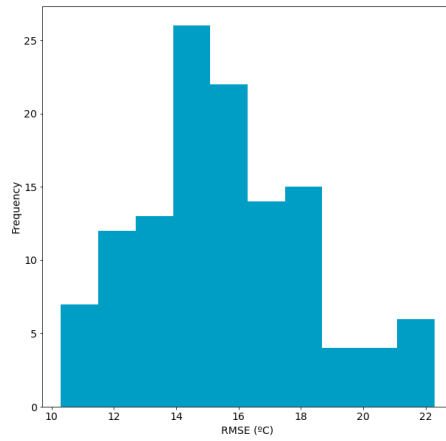
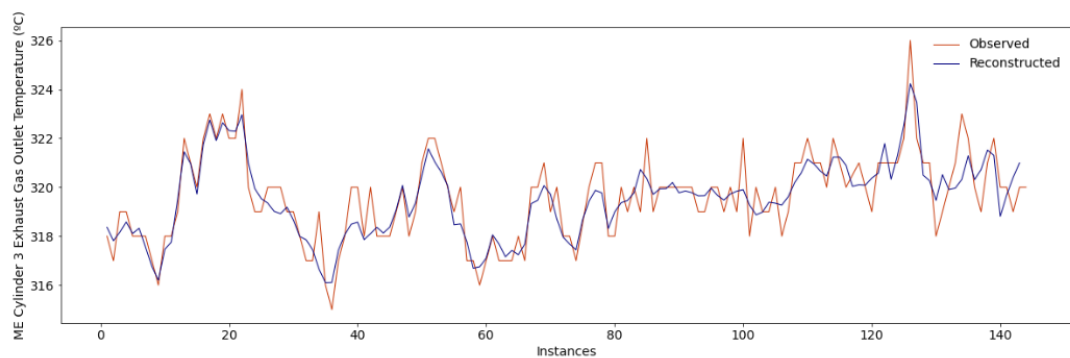
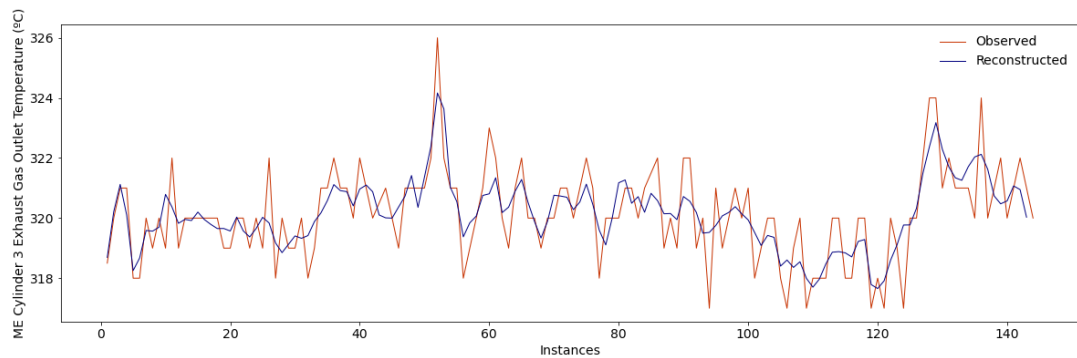


Fig. D.4.8. Histogram of the reconstructed errors of the sequences with degradation patterns.

Examples of reconstructed sequences are also introduced in Figs. E.4.9 – E.4.11.



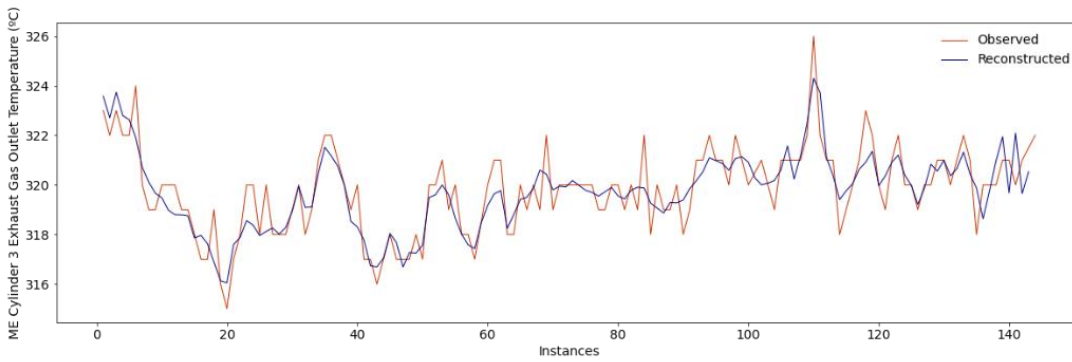
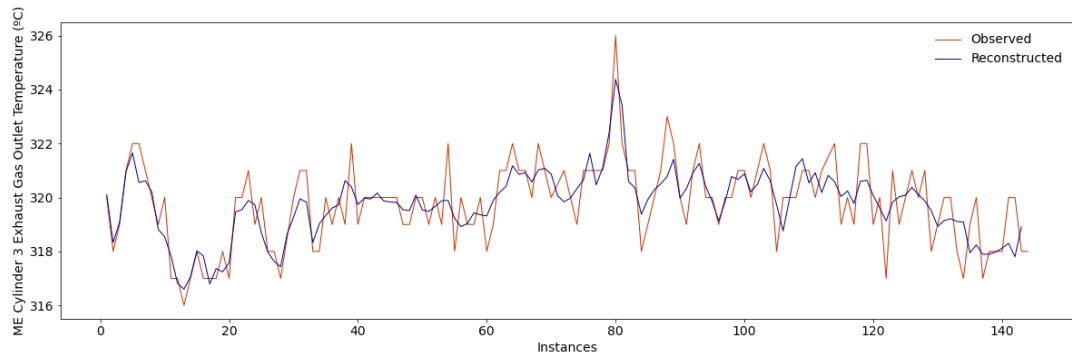
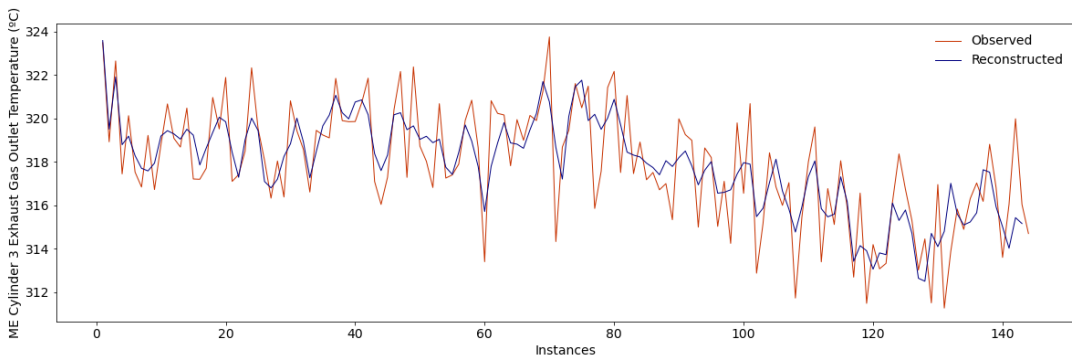


Fig. D.4.9. Example of normal reconstructed sequences (test set).



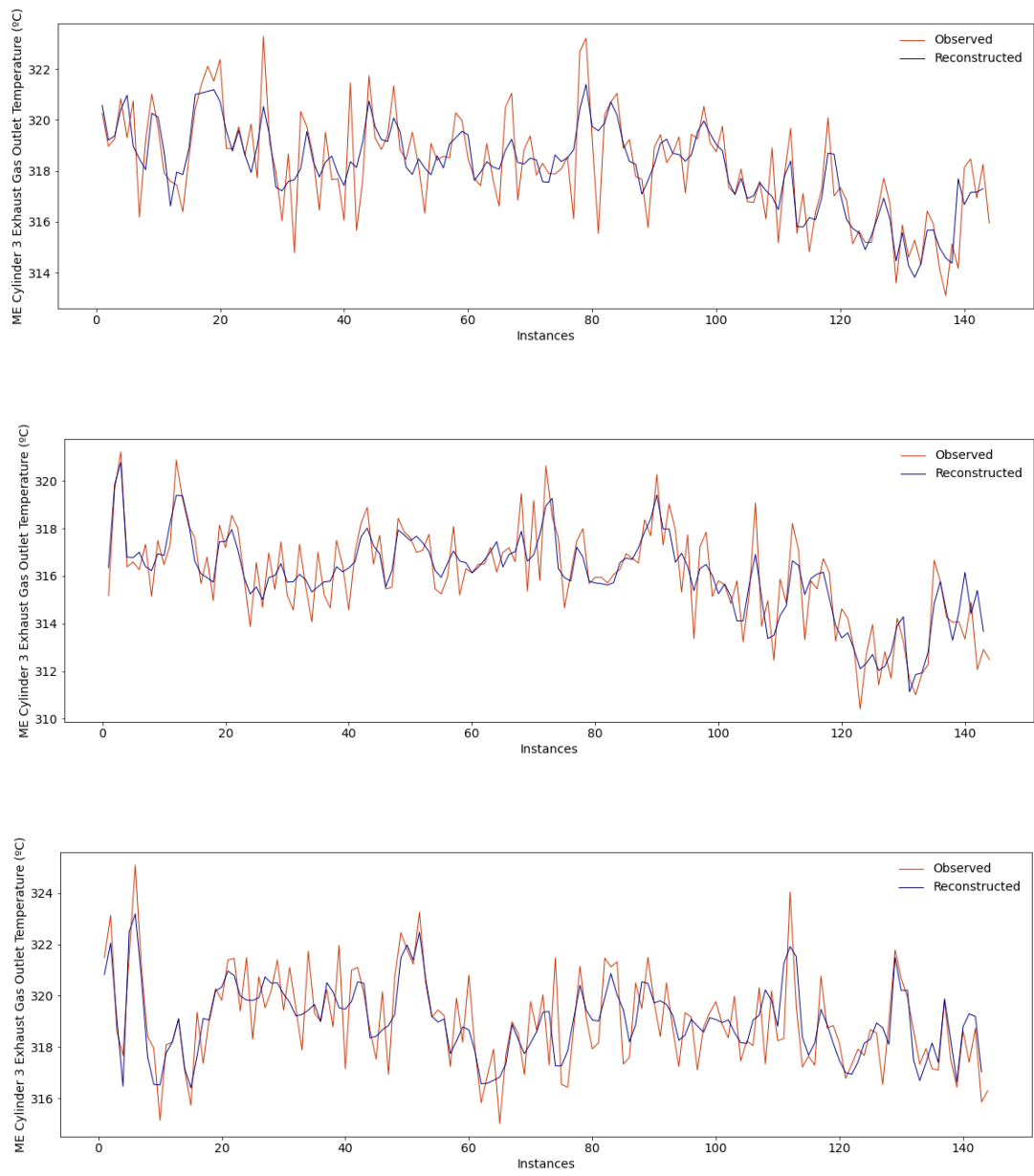


Fig. D.4.10. Example of reconstructed sequences with collective anomalies.

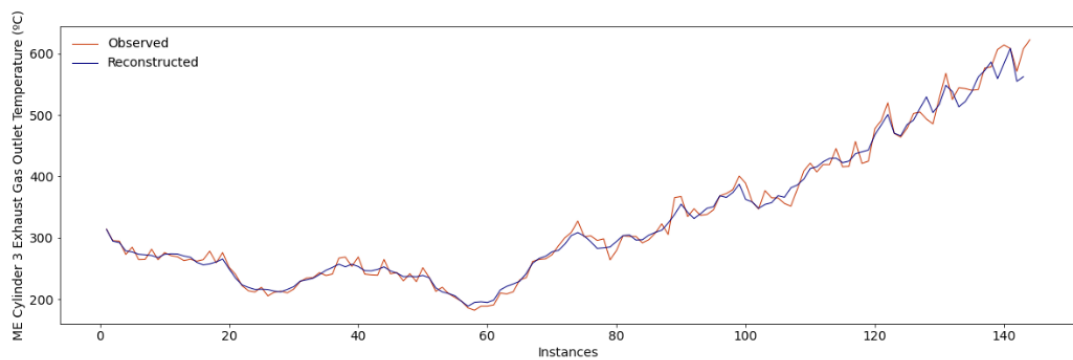
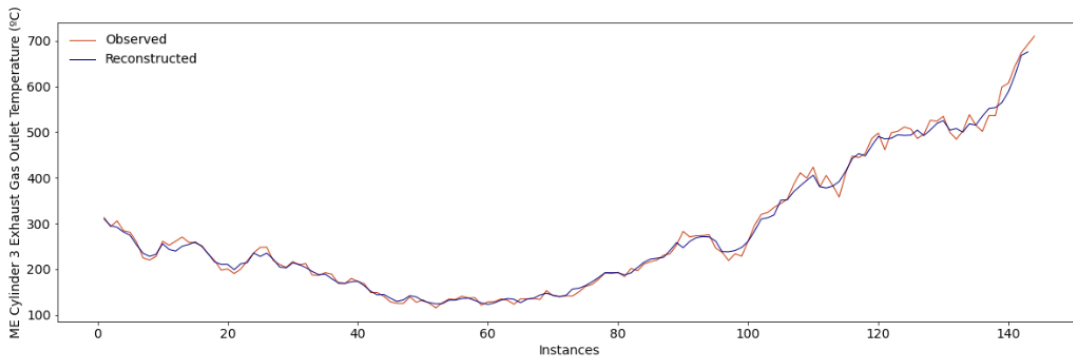
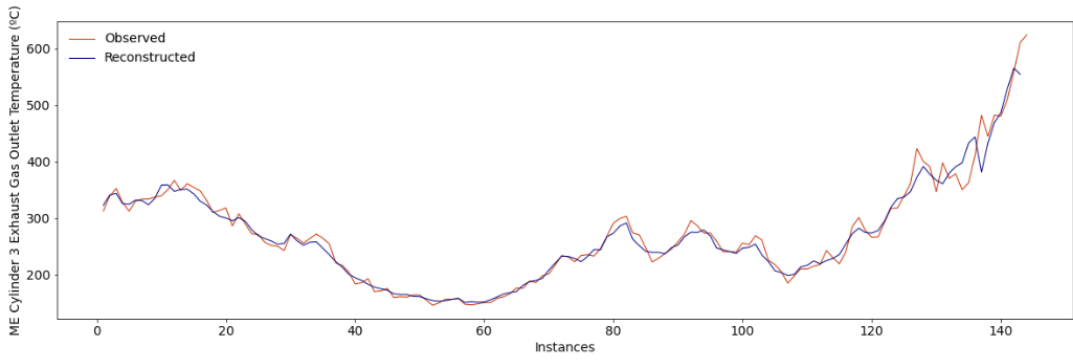
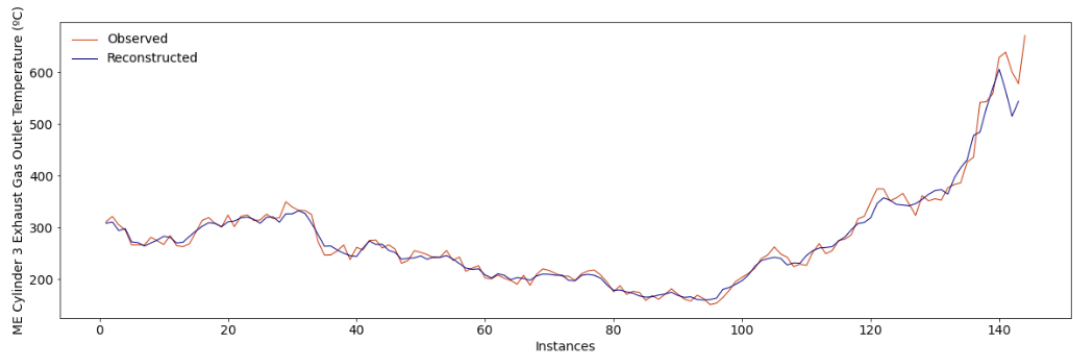


Fig. D.4.11. Example of reconstructed sequences with degradation patterns.

Those sequences detected as anomalous are then classified into two categories: sequences with collective anomalies, and sequences with degradation patterns. Accordingly, as the implemented approach refers to a time series imaging approach (see section 5.3.2. *Fault Identification*), the anomalous sequences detected are transformed into images. Examples of these can be perceived in Figs. E.4.12 – E.4.13.

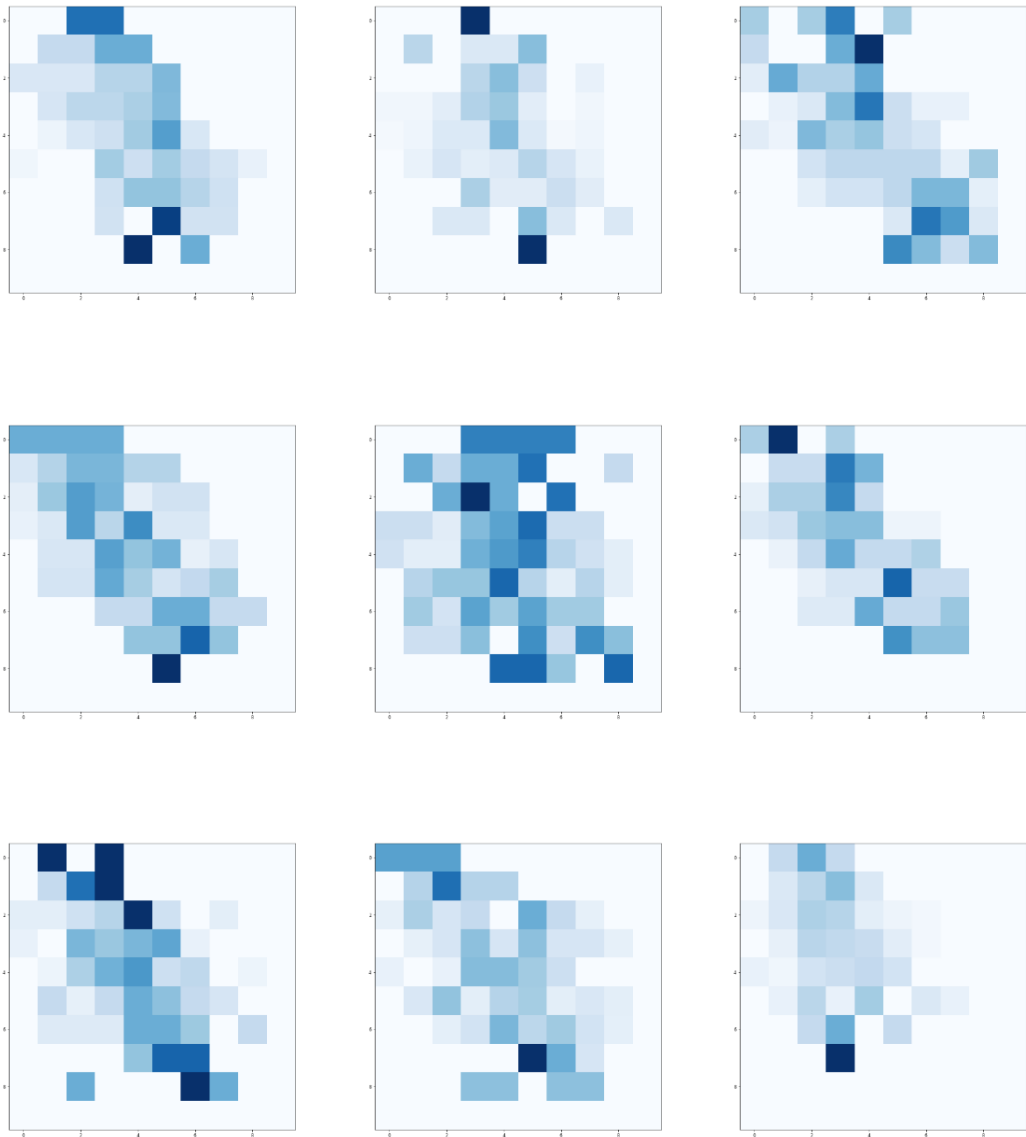


Fig. D.4.12. Images with collective anomalies.

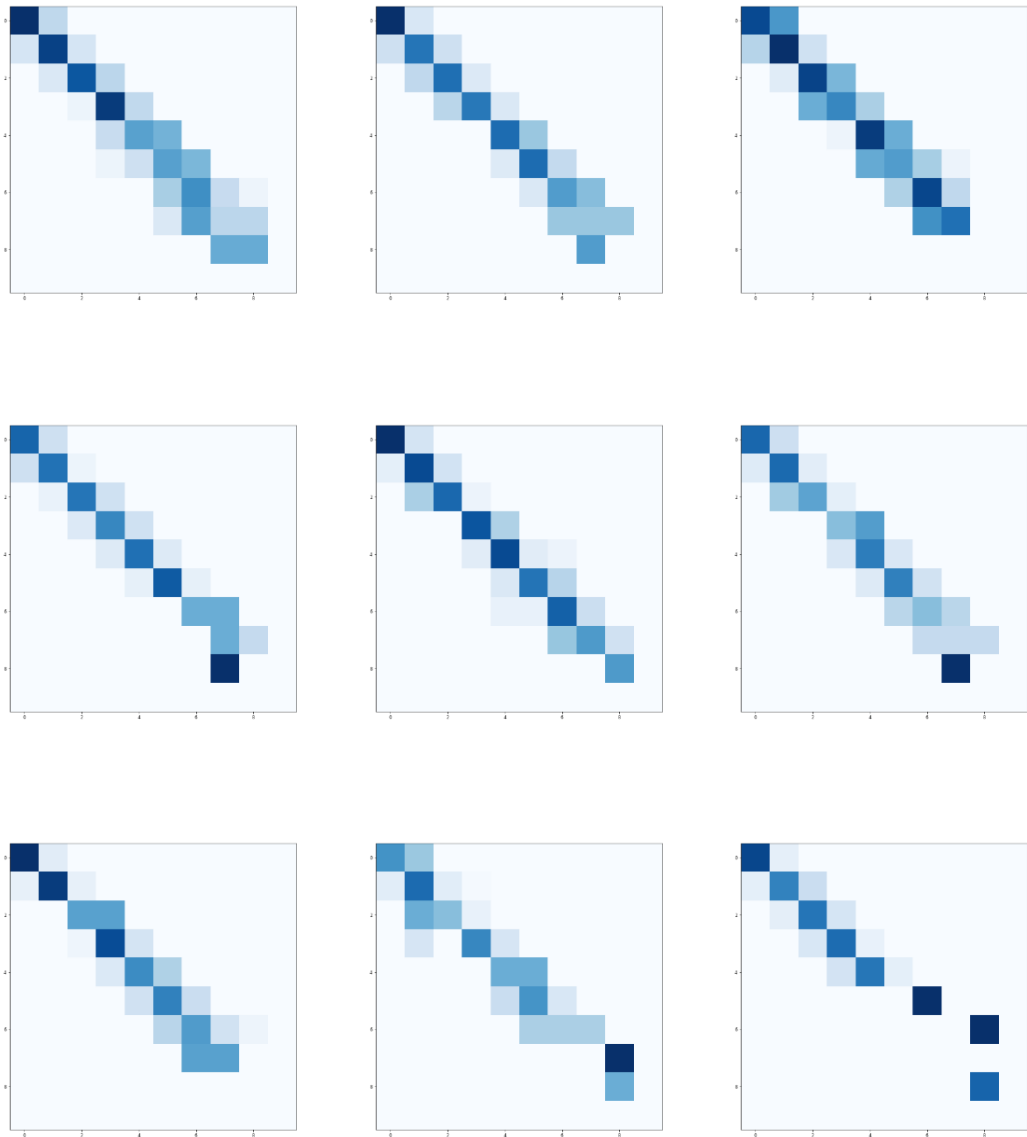
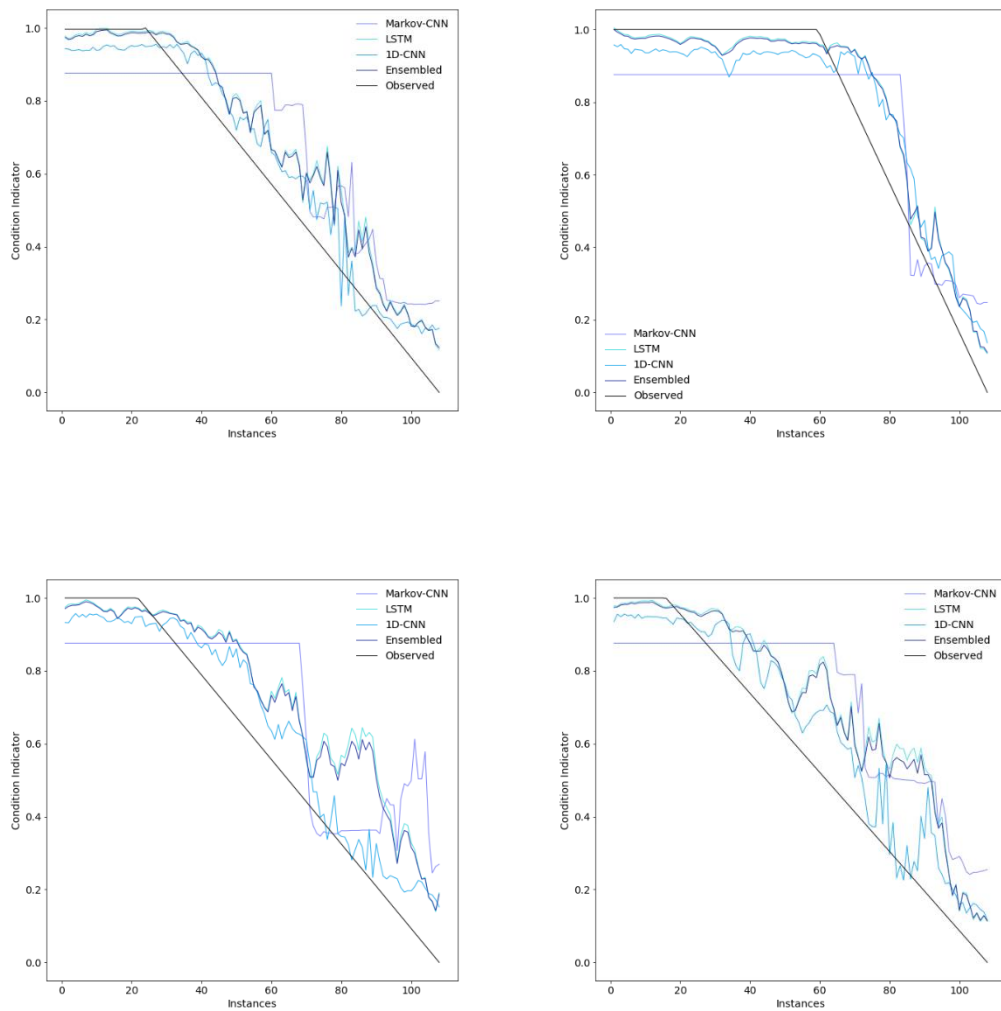


Fig. D.4.13. Images with degradation patterns.

As it can be perceived, the two categories can be easily distinguished. This aspect facilitated the achievement of the maximum accuracy score in this process, thus classifying all the images adequately.

By adequately classifying such images, the sequences with degradation patterns are selected so that the RUL can be predicted. Examples of such a prediction are presented in Fig. D.4.14. The RMSE and Maintenance Score of the first 100 sequences are also presented in Table E.4.2.



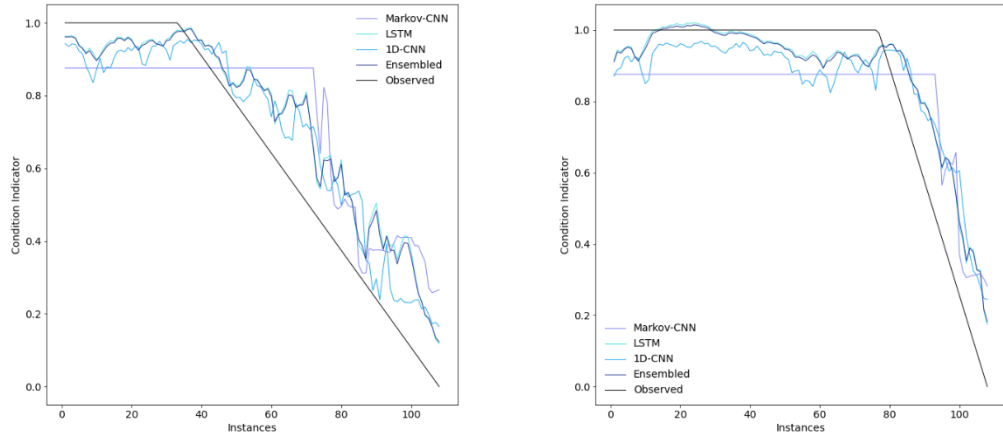


Fig. D.4.14. Examples of condition indicator.

Table D.4.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models

Sequences	RMSE (Minutes)				Maintenance Score			
	Markov-CNN	LSTM	1D-CNN	Ensembled	Markov-CNN	LSTM	1D-CNN	Ensembled
Sequence 1	69.35	61.03	60.24	60.82	3.84	3.36	3.44	3.37
Sequence 2	41.59	36.60	34.07	36.23	2.10	1.68	1.61	1.67
Sequence 3	62.55	50.06	51.25	50.00	3.82	2.92	3.11	2.94
Sequence 4	43.94	34.76	30.26	34.21	2.17	1.58	1.50	1.57
Sequence 5	23.87	16.47	11.93	15.73	1.31	0.79	0.65	0.77
Sequence 6	30.51	26.14	13.89	24.61	1.51	1.27	0.73	1.21
Sequence 7	40.89	28.43	30.70	28.02	2.42	1.44	1.80	1.41
Sequence 8	18.70	19.14	22.10	19.34	1.23	0.86	1.12	0.88
Sequence 9	30.12	23.65	30.93	24.16	1.65	1.11	1.71	1.16
Sequence 10	21.70	19.61	28.12	20.32	1.34	0.92	1.42	0.96
Sequence 11	30.17	21.46	18.55	20.90	1.67	1.07	1.08	1.06
Sequence 12	47.71	37.94	32.90	37.32	2.37	1.62	1.40	1.58
Sequence 13	38.19	32.02	32.90	32.01	1.90	1.20	1.46	1.23
Sequence 14	23.38	23.42	18.55	22.72	1.31	1.00	0.91	0.99
Sequence 15	30.07	28.68	32.14	28.82	1.87	1.46	1.78	1.48
Sequence 16	19.43	15.66	20.55	15.99	1.32	0.95	1.30	0.98
Sequence 17	43.29	36.71	31.07	35.83	2.66	1.85	1.63	1.80
Sequence 18	20.59	16.14	14.07	15.31	1.17	0.81	0.78	0.77
Sequence 19	18.87	25.78	35.03	26.49	1.30	1.65	2.35	1.70
Sequence 20	30.99	29.51	33.11	29.62	2.14	1.84	2.09	1.86
Sequence 21	32.78	20.48	14.06	19.52	1.69	0.97	0.68	0.92
Sequence 22	34.42	26.14	26.01	25.96	1.78	1.12	1.33	1.13
Sequence 23	36.11	29.64	28.96	29.49	1.89	1.27	1.41	1.29

Sequence 24	45.88	32.18	42.71	33.05	2.68	1.86	2.56	1.92
Sequence 25	25.06	18.49	21.33	18.62	1.50	0.86	1.16	0.88
Sequence 26	37.81	26.34	31.61	26.67	2.43	1.54	1.89	1.58
Sequence 27	25.95	14.61	16.64	14.65	1.52	0.73	0.99	0.75
Sequence 28	67.11	44.50	54.91	45.18	3.93	2.31	3.13	2.38
Sequence 29	74.92	66.98	59.20	65.99	3.90	3.44	2.76	3.36
Sequence 30	43.49	27.38	17.98	26.31	2.20	1.30	0.93	1.26
Sequence 31	54.15	61.00	60.87	60.93	3.15	3.58	3.65	3.59
Sequence 32	20.67	13.46	14.75	13.45	1.26	0.62	0.85	0.64
Sequence 33	25.13	14.50	18.60	14.69	1.48	0.66	1.01	0.69
Sequence 34	36.24	39.29	38.09	38.60	2.14	2.11	2.29	2.11
Sequence 35	29.15	20.23	25.33	20.56	1.91	1.13	1.54	1.17
Sequence 36	19.80	23.61	23.88	23.22	1.09	1.34	1.39	1.33
Sequence 37	25.15	16.87	22.11	17.21	1.62	0.97	1.29	1.00
Sequence 38	55.64	35.48	35.41	35.37	3.54	1.97	2.06	1.97
Sequence 39	64.86	47.78	54.42	48.07	3.81	2.37	2.93	2.41
Sequence 40	17.56	14.47	20.32	14.94	1.24	0.80	1.23	0.85
Sequence 41	63.09	38.20	53.64	39.29	3.94	2.03	3.14	2.14
Sequence 42	20.32	14.61	15.99	14.60	1.23	0.79	0.96	0.80
Sequence 43	34.43	23.47	22.62	23.23	1.84	1.12	1.21	1.12
Sequence 44	33.46	23.31	20.39	22.90	1.78	1.02	1.05	1.02
Sequence 45	27.41	19.93	21.93	20.06	1.59	0.85	1.11	0.87
Sequence 46	31.83	23.36	28.90	23.79	1.72	0.97	1.34	1.00
Sequence 47	64.76	80.09	80.83	80.09	3.75	4.42	4.50	4.43
Sequence 48	24.60	17.35	20.33	17.57	1.49	0.67	1.00	0.70
Sequence 49	30.47	24.93	19.62	24.30	1.69	1.09	1.03	1.08

Sequence 50	29.17	21.63	15.99	20.82	1.63	1.05	0.91	1.03
Sequence 51	48.05	51.23	49.82	50.87	2.82	2.98	2.95	2.97
Sequence 52	44.06	29.34	34.78	29.60	2.71	1.50	1.69	1.51
Sequence 53	23.66	17.97	21.18	17.97	1.62	0.99	1.21	1.01
Sequence 54	17.30	20.35	25.04	20.34	1.11	1.17	1.43	1.18
Sequence 55	33.85	19.94	25.39	20.31	2.15	1.08	1.48	1.11
Sequence 56	37.84	32.85	32.09	32.38	2.60	2.22	1.97	2.20
Sequence 57	35.69	41.99	56.08	43.29	2.46	2.82	3.69	2.91
Sequence 58	54.34	55.95	49.63	55.25	2.59	2.81	2.44	2.77
Sequence 59	23.00	16.45	19.61	16.65	1.44	0.76	1.10	0.78
Sequence 60	39.06	24.89	21.82	24.51	1.98	1.09	1.12	1.09
Sequence 61	24.86	15.64	14.61	15.23	1.33	0.82	0.77	0.80
Sequence 62	32.16	22.62	29.02	23.14	2.09	1.19	1.69	1.22
Sequence 63	62.46	45.71	51.54	46.17	3.63	2.85	3.21	2.88
Sequence 64	22.89	15.52	18.62	15.68	1.44	0.89	1.18	0.92
Sequence 65	26.22	17.75	18.75	17.49	1.55	1.06	1.19	1.06
Sequence 66	27.38	21.40	27.31	21.69	1.68	1.07	1.36	1.09
Sequence 67	40.99	44.40	49.68	44.76	2.60	2.66	2.91	2.68
Sequence 68	42.07	41.85	32.24	40.56	2.37	2.12	1.67	2.05
Sequence 69	20.61	23.92	26.26	23.85	1.47	1.59	1.70	1.59
Sequence 70	37.75	31.31	30.70	30.64	2.24	1.76	1.93	1.73
Sequence 71	45.46	48.71	52.97	49.04	2.70	2.97	3.27	3.00
Sequence 72	45.93	43.02	40.68	42.68	2.30	2.00	1.95	2.00
Sequence 73	40.16	37.59	47.40	38.44	2.71	2.36	3.00	2.43
Sequence 74	62.71	43.09	57.02	44.32	4.02	2.31	3.28	2.40
Sequence 75	18.41	15.25	21.51	15.69	1.25	0.83	1.17	0.86

Sequence 76	22.35	23.01	21.21	22.72	1.25	1.27	1.19	1.26
Sequence 77	34.75	27.33	29.30	27.45	1.80	1.14	1.39	1.16
Sequence 78	17.65	16.89	21.26	17.21	1.21	1.06	1.40	1.09
Sequence 79	29.72	28.68	23.76	27.92	1.37	1.42	1.19	1.38
Sequence 80	19.81	14.72	16.29	14.73	1.32	0.81	1.02	0.83
Sequence 81	61.94	63.38	55.84	62.58	3.09	3.23	2.77	3.18
Sequence 82	32.54	23.80	23.98	23.72	1.73	0.98	1.18	0.99
Sequence 83	25.22	18.67	23.43	19.03	1.50	0.80	1.19	0.84
Sequence 84	27.36	20.38	17.36	19.85	1.49	1.03	0.96	1.02
Sequence 85	51.47	42.02	33.88	41.14	2.54	1.97	1.64	1.92
Sequence 86	52.69	42.23	43.61	42.27	3.02	1.90	2.06	1.90
Sequence 87	20.09	12.89	19.99	13.51	1.38	0.67	1.07	0.71
Sequence 88	35.14	27.94	27.63	27.52	2.08	1.57	1.57	1.56
Sequence 89	64.22	58.30	61.14	58.43	4.34	3.68	3.84	3.69
Sequence 90	22.90	21.99	20.91	21.80	1.02	1.13	1.10	1.13
Sequence 91	36.77	28.75	31.74	28.91	2.33	1.55	1.79	1.56
Sequence 92	25.19	24.47	27.46	24.01	1.30	1.32	1.69	1.29
Sequence 93	50.65	36.81	39.76	36.93	3.12	1.90	2.12	1.91
Sequence 94	76.65	76.09	76.03	75.81	3.73	3.72	3.42	3.66
Sequence 95	21.08	19.54	19.25	19.36	1.25	0.87	1.04	0.88
Sequence 96	27.69	40.89	33.58	40.05	1.39	1.83	1.59	1.81
Sequence 97	20.10	12.59	15.87	12.70	1.29	0.76	1.00	0.78
Sequence 98	31.64	26.74	15.71	25.33	1.70	1.29	0.79	1.24
Sequence 99	23.78	16.01	17.52	16.01	1.37	0.69	0.97	0.71
Sequence 100	20.38	13.60	16.45	13.65	1.21	0.73	0.93	0.74
Median	32.16	26.14	27.46	25.60	1.78	1.29	1.41	1.26

D.5. Main Engine Cylinder 4

The parameter analysed refers to the exhaust gas outlet temperature. A graphical representation of such a parameter is expressed in Fig. D.5.1. The descriptive statistics is also introduced in Table E.5.1.

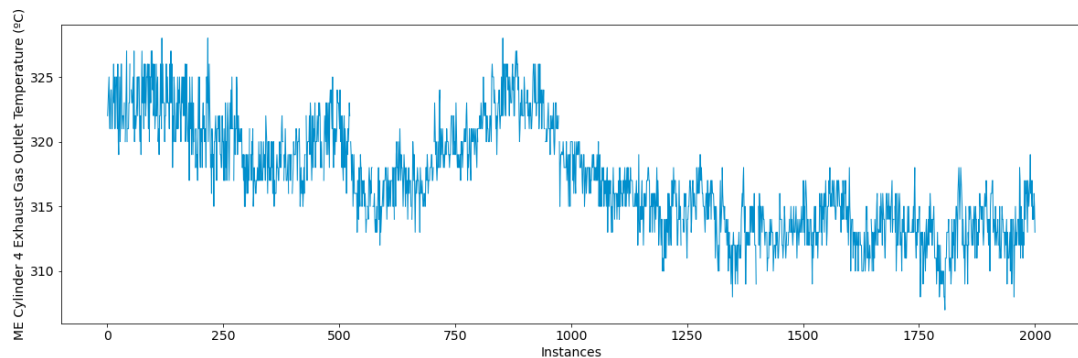


Fig. D.5.1. Graphical representation of the exhaust gas outlet temperature parameter.

Table D.5.1. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
Cyl. 4 Exh. Gas Out. Temp	317.12	4.10	307	314	317	320	328

As part of the data pre-processing phase, the identification of operational states step has been implemented (see section 4.5. *A Novel Framework for the Identification of Steady States* for a comprehensive explanation of such a step). In total, only one operational state has been identified, as perceived in Fig. D.5.2.

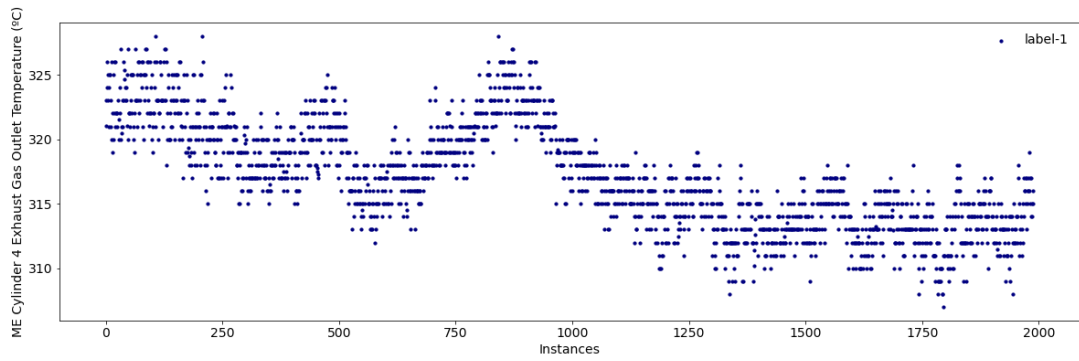
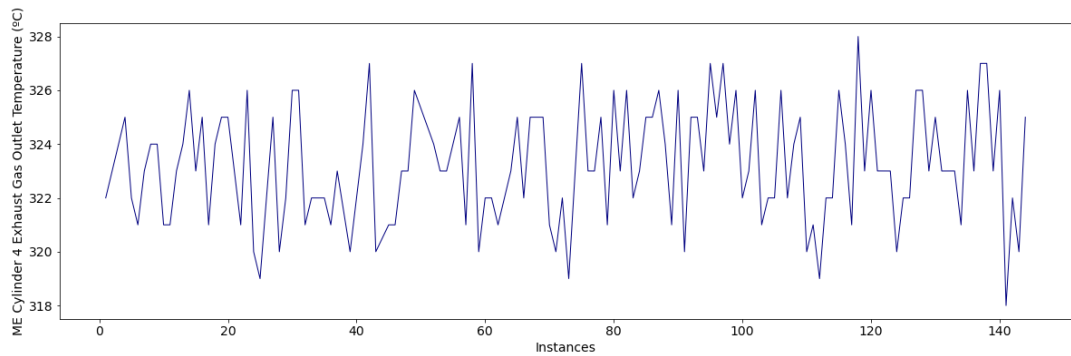


Fig. D.5.2. Identification of the operational states for the monitored parameter.

Subsequently, due to the lack of fault data, both collective anomalies and degradation patterns are simulated. Some examples are presented in Figs. E.5.4 – E.4.5. Examples of normal sequences are also introduced in Fig. D.5.3.



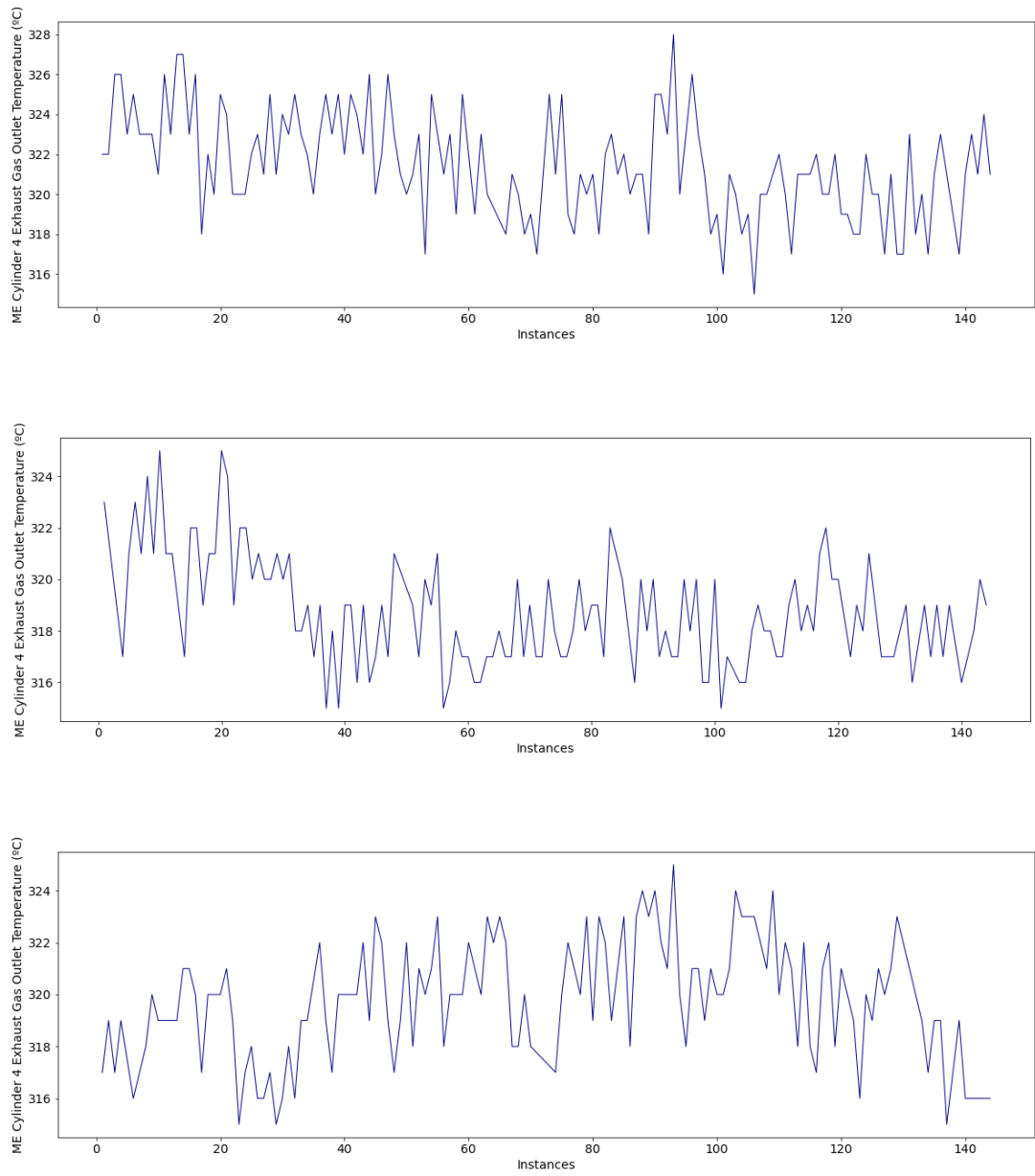


Fig. D.5.3. Example of normal sequences.

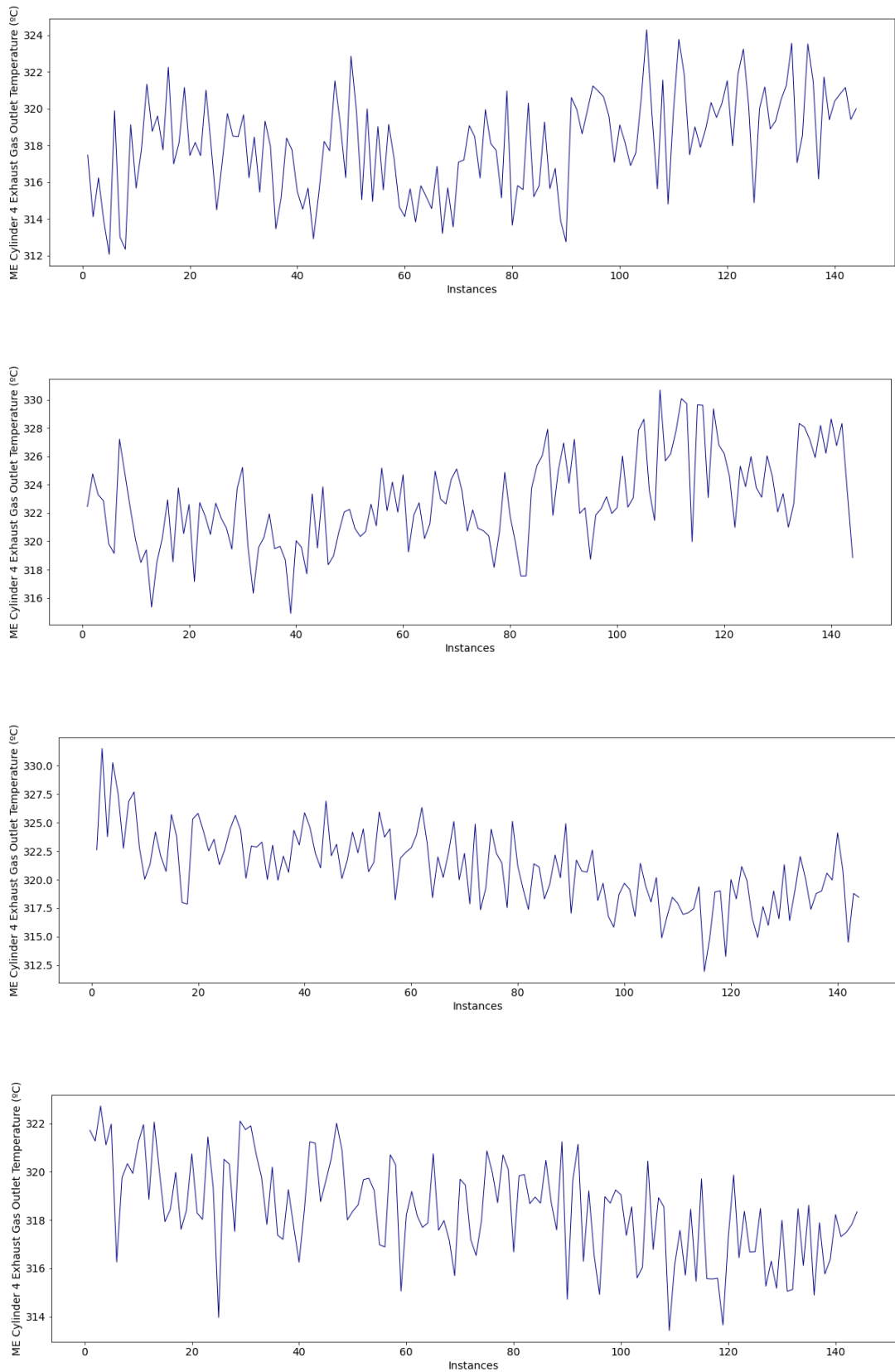
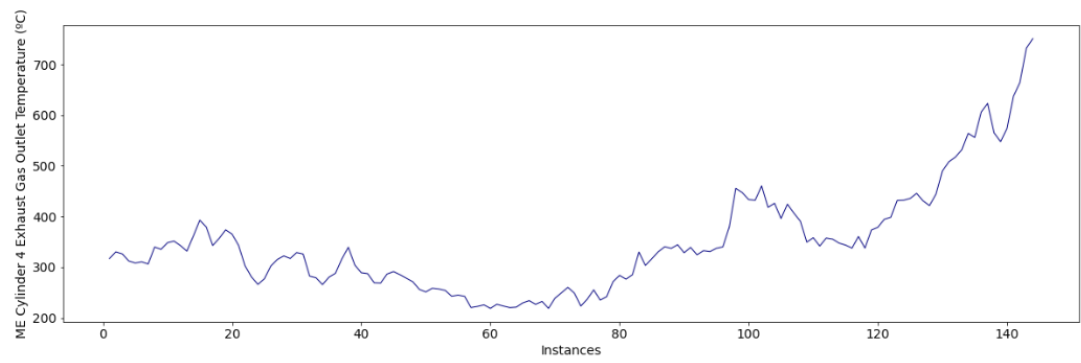
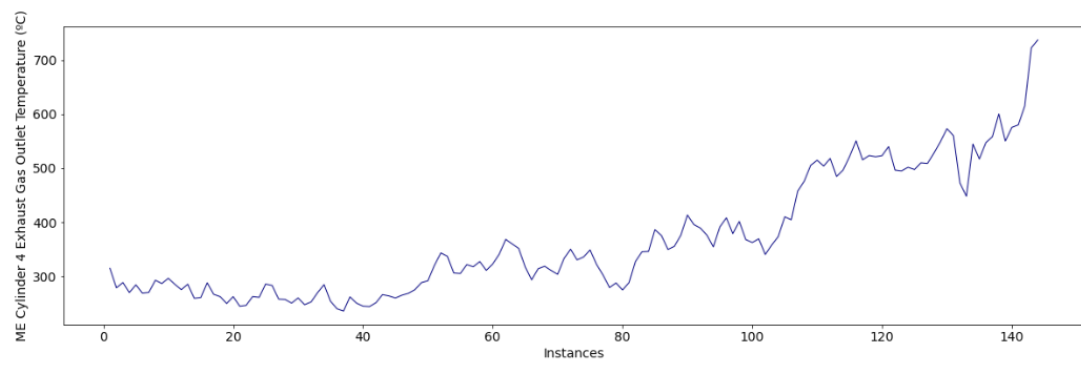
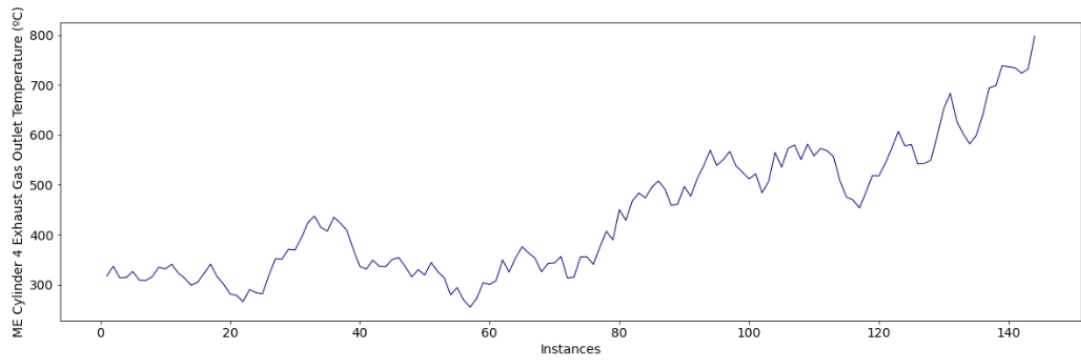


Fig. D.5.4. Example of sequences with collective anomalies.



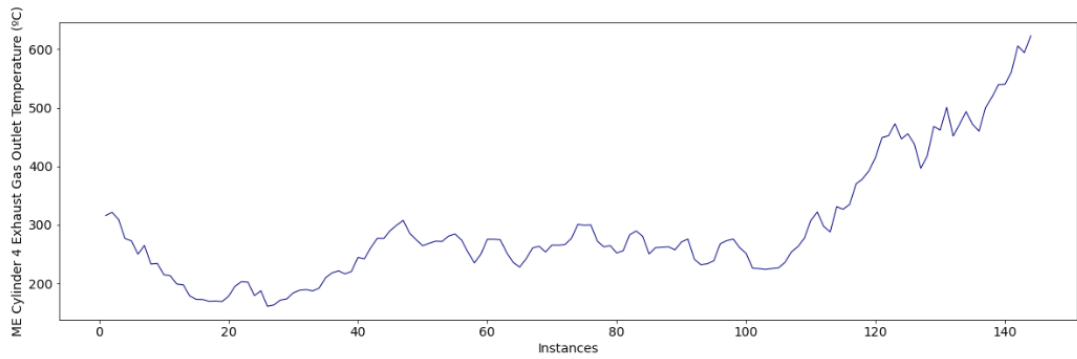


Fig. D.5.5. Example of sequences with degradation patterns.

As part of the MA framework, the subsequent module to be applied is the diagnostic analytics module. Accordingly, the fault detection step is implemented as stated in section 5.3.1. *Fault Detection*. As perceived in the histograms (Figs. E.5.6 – E.5.8), a simple threshold is adequate in this case study to distinguish the normal sequences from the abnormal sequences.

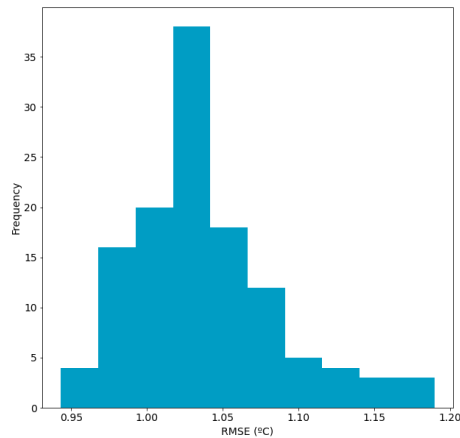


Fig. D.5.6. Histogram of the reconstructed errors of the normal sequences (test set).

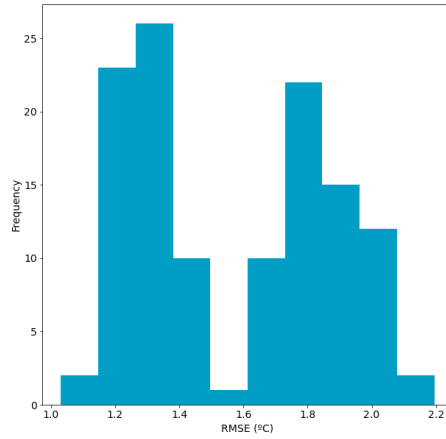


Fig. D.5.7. Histogram of the reconstructed errors of the sequences with collective anomalies.

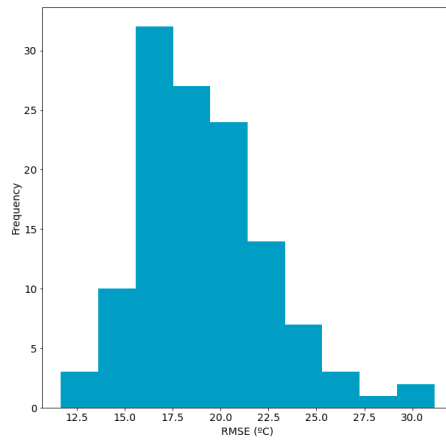


Fig. D.5.8. Histogram of the reconstructed errors of the sequences with degradation patterns.

Examples of reconstructed sequences are also introduced in Figs. E.5.9 – E.5.11.

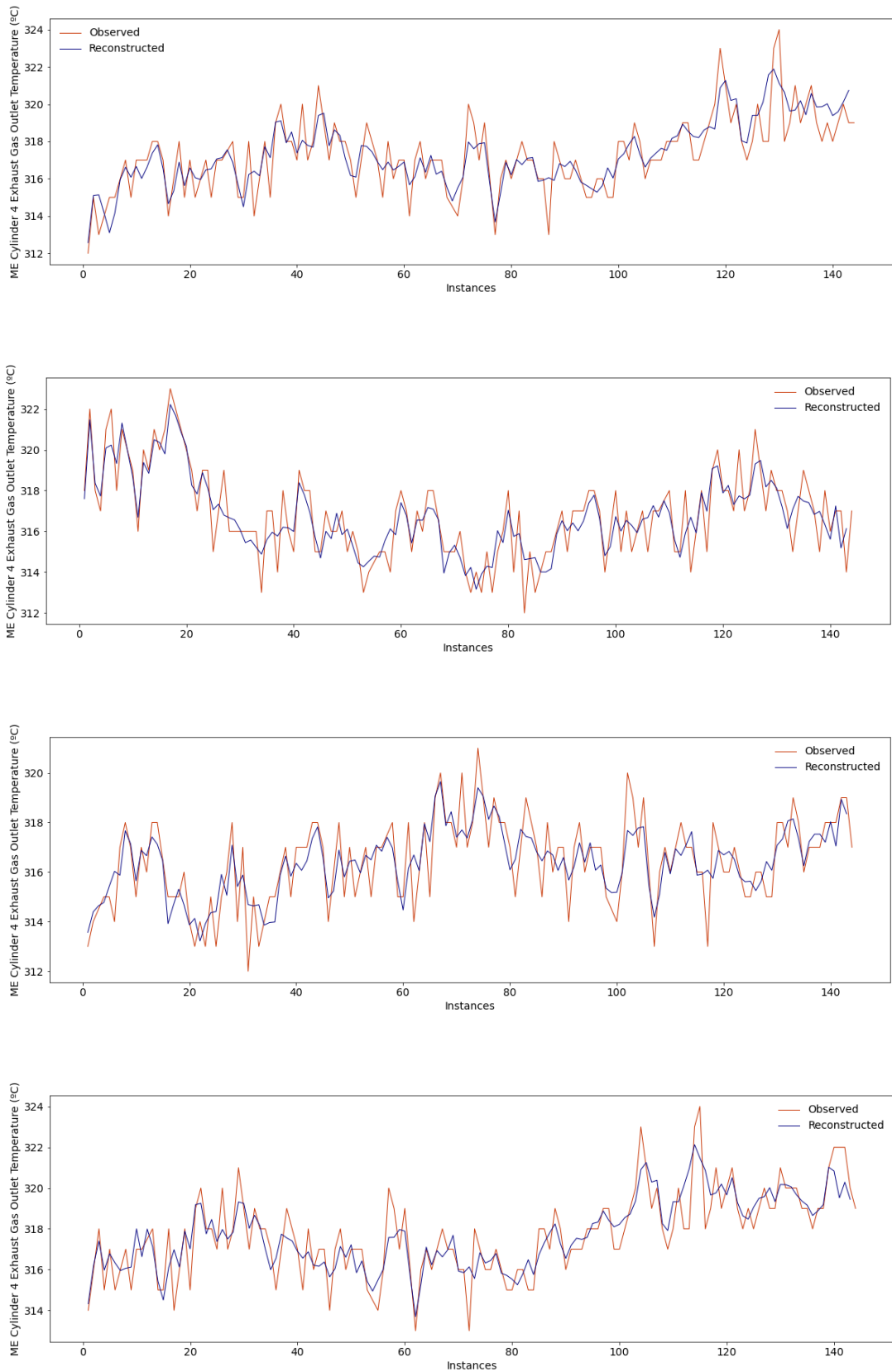


Fig. D.5.9. Example of normal reconstructed sequences (test set).

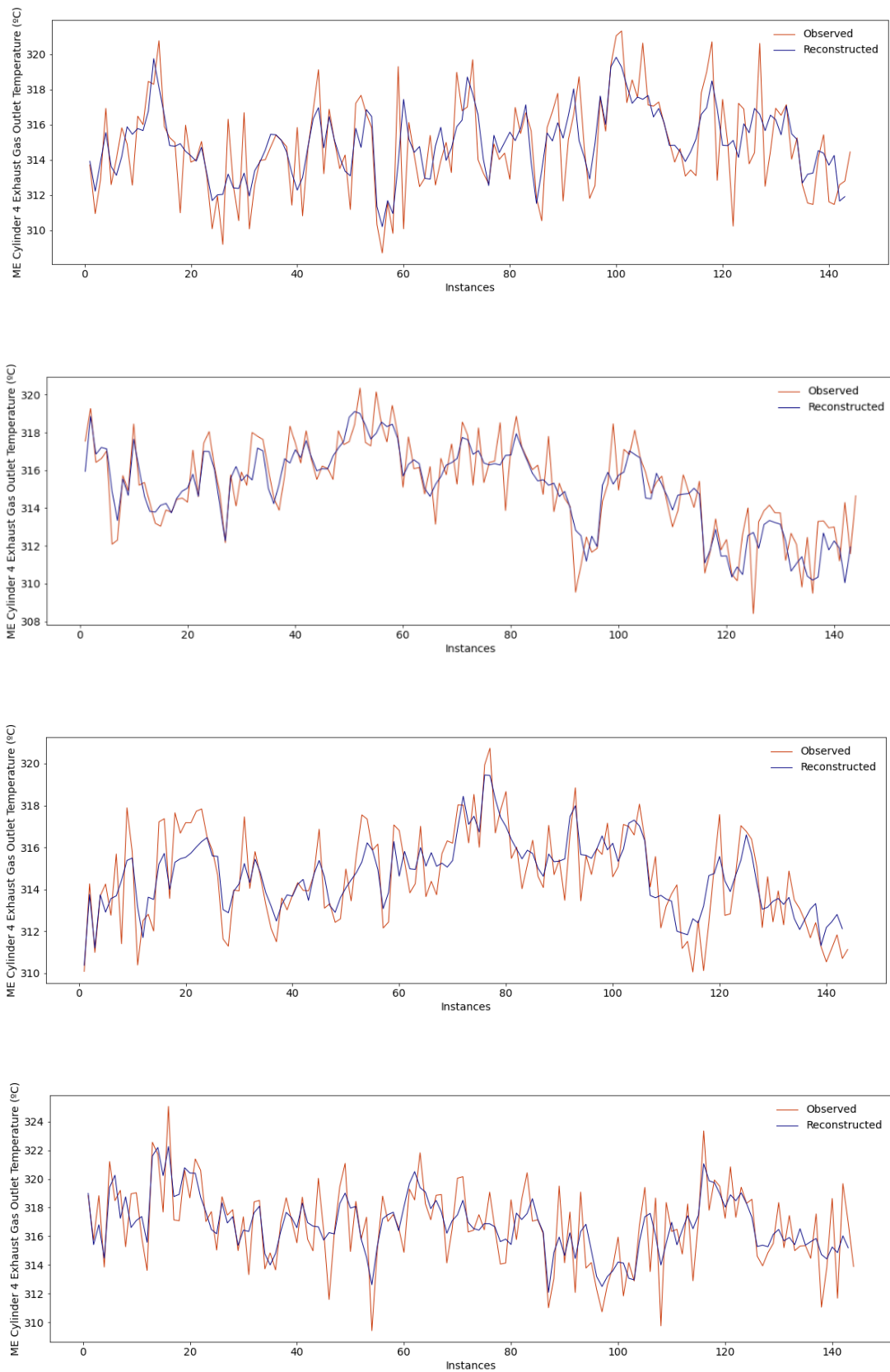


Fig. D.5.10. Example of reconstructed sequences with collective anomalies.

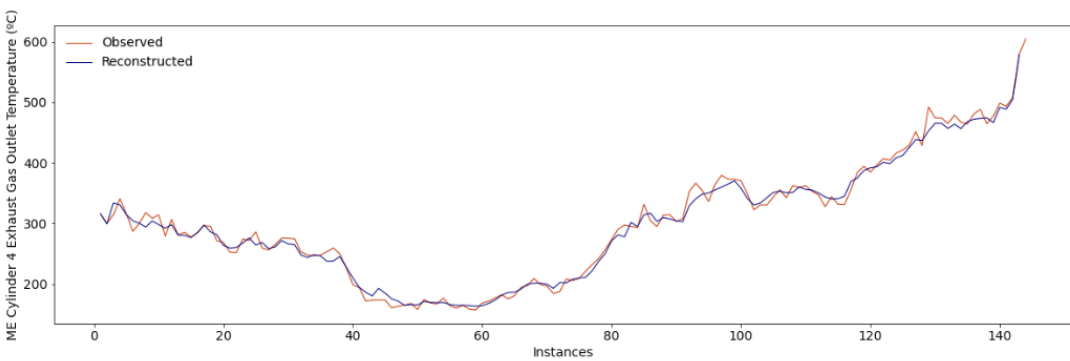
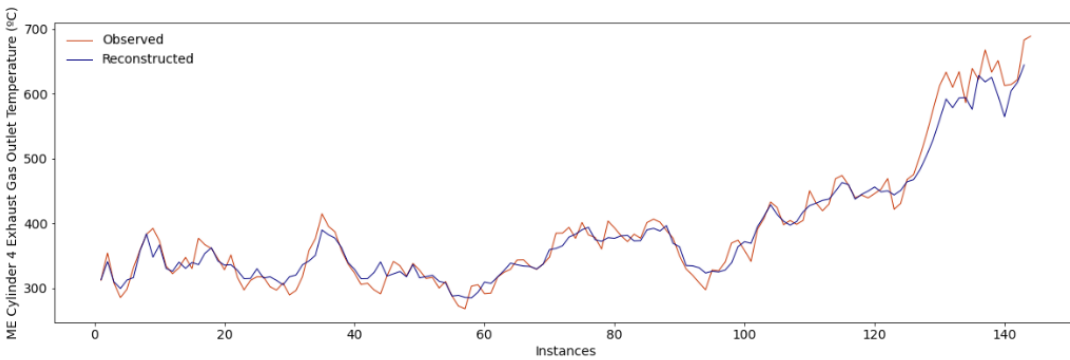
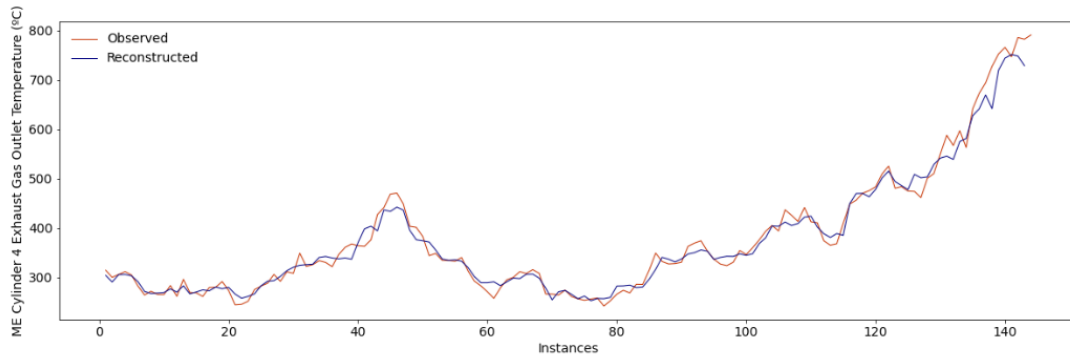
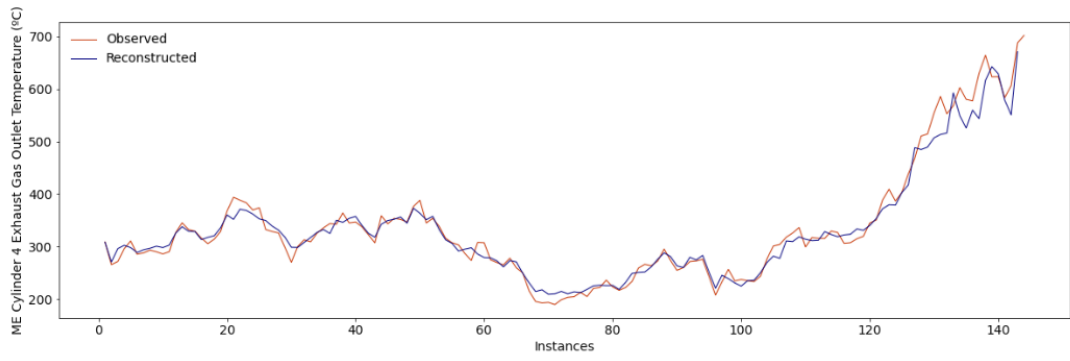


Fig. D.5.11. Example of reconstructed sequences with degradation patterns.

Those sequences detected as anomalous are then classified into two categories: sequences with collective anomalies, and sequences with degradation patterns. Accordingly, as the implemented approach refers to a time series imaging approach (see section 5.3.2. *Fault Identification*), the anomalous sequences detected are transformed into images. Examples of these can be perceived in Figs. E.5.12 – E.5.13.

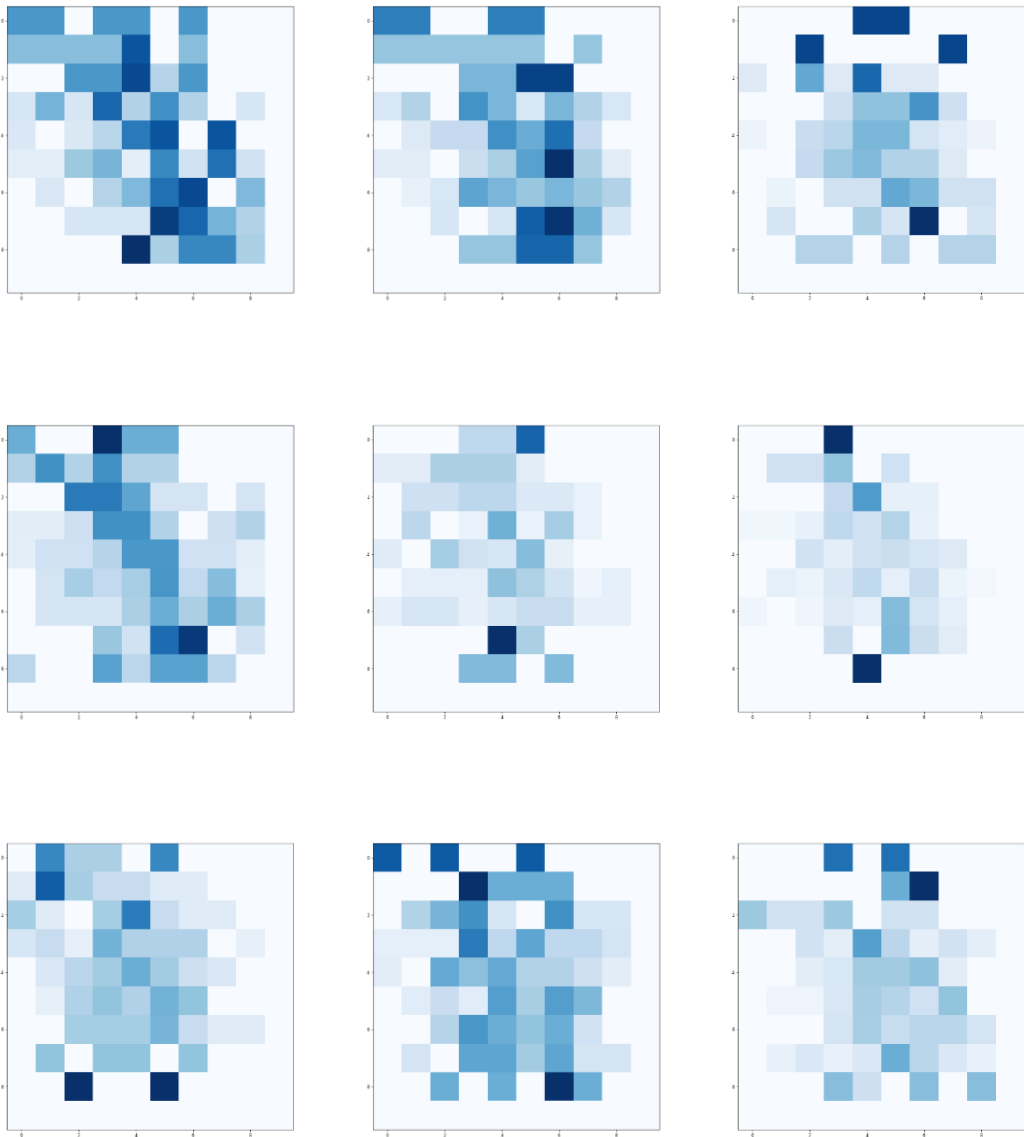


Fig. D.5.12. Images with collective anomalies.

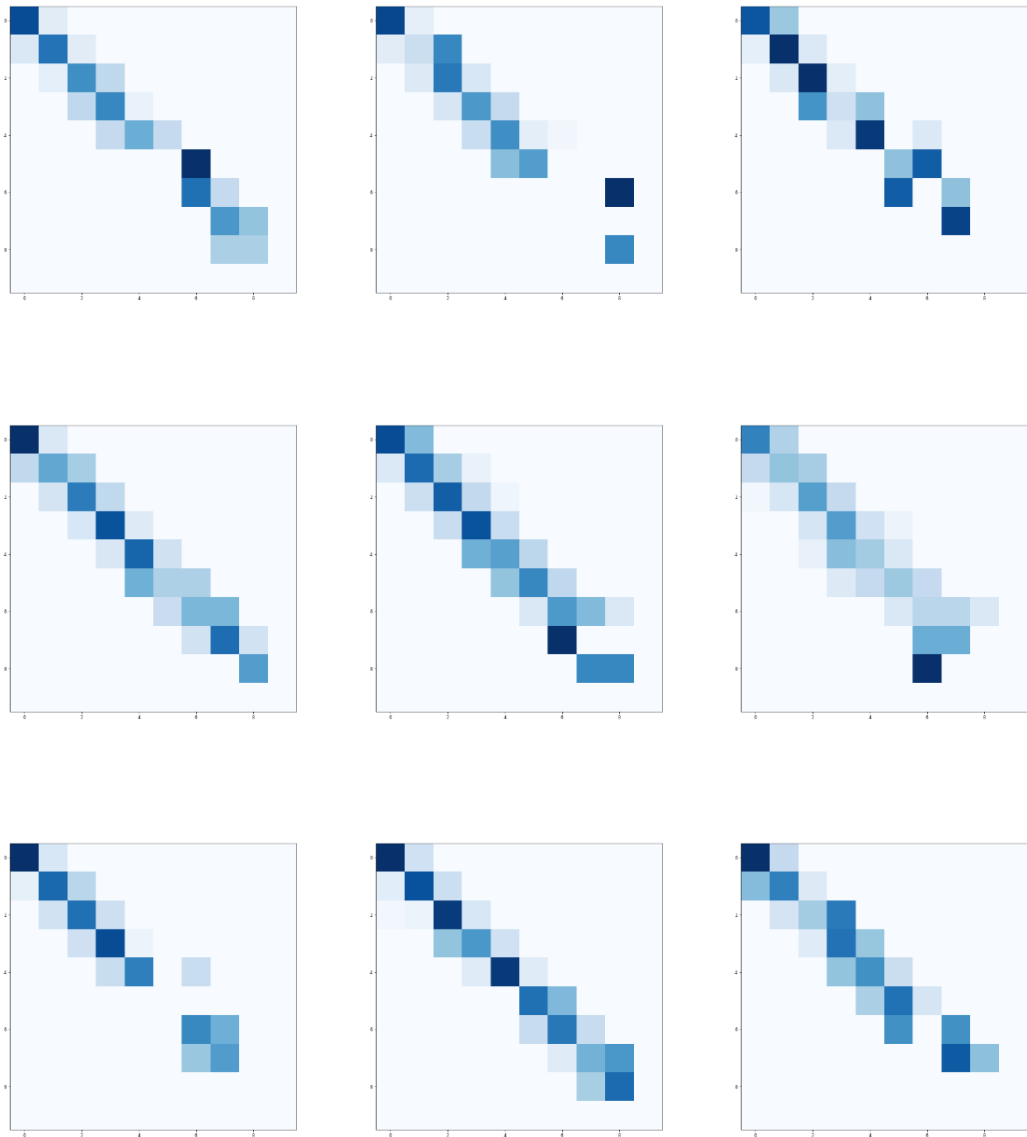
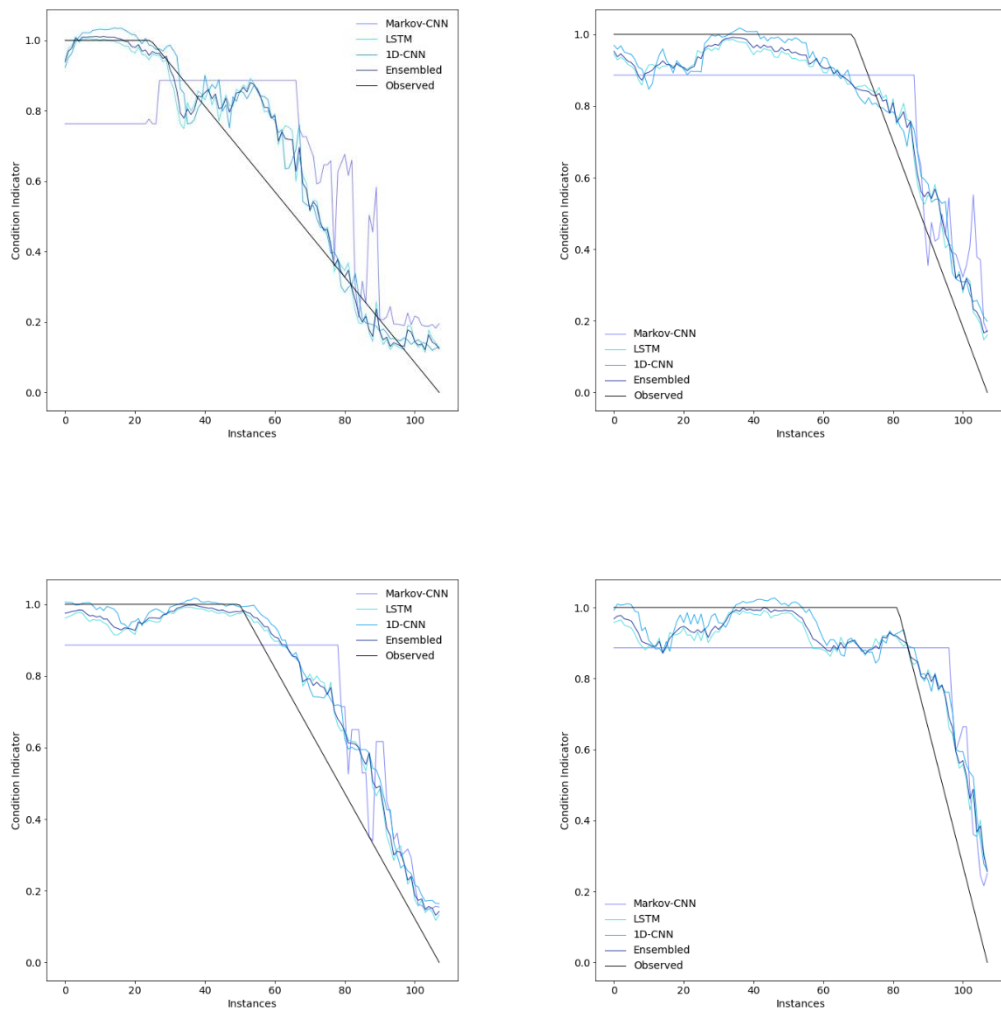


Fig. D.5.13. Images with degradation patterns.

As it can be perceived, the two categories can be easily distinguished. This aspect facilitated the achievement of the maximum accuracy score in this process, thus classifying all the images adequately.

By adequately classifying such images, the sequences with degradation patterns are selected so that the RUL can be predicted. Examples of such a prediction are presented in Fig. D.5.14. The RMSE and Maintenance Score of the first 100 sequences are also presented in Table E.5.2.



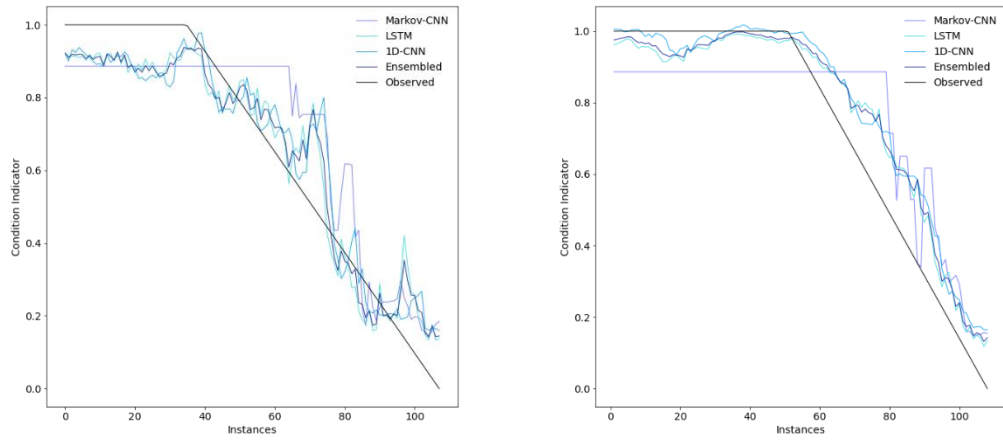


Fig. D.5.14. Examples of condition indicator.

Table D.5.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.

Sequences	RMSE (Minutes)				Maintenance Score			
	Markov-CNN	LSTM	1D-CNN	Ensembled	Markov-CNN	LSTM	1D-CNN	Ensembled
Sequence 1	23.42	21.33	25.49	22.41	1.38	0.97	1.09	0.98
Sequence 2	54.17	58.06	56.08	57.19	2.63	3.03	2.83	2.96
Sequence 3	26.68	19.33	17.77	18.50	1.51	1.13	0.90	1.06
Sequence 4	29.99	35.26	38.74	35.82	1.97	2.44	2.61	2.48
Sequence 5	47.88	30.33	26.30	28.72	3.02	1.90	1.55	1.78
Sequence 6	49.53	34.63	28.89	32.00	3.11	2.15	1.63	1.97
Sequence 7	21.30	21.87	24.13	22.17	1.32	1.29	1.40	1.32
Sequence 8	30.39	14.60	13.76	13.98	1.70	0.66	0.61	0.61
Sequence 9	40.22	41.42	43.66	41.81	2.64	2.77	2.79	2.77
Sequence 10	37.27	13.32	15.64	13.34	1.64	0.68	0.78	0.66
Sequence 11	25.18	22.74	25.25	23.02	1.29	1.06	1.19	1.09
Sequence 12	40.06	33.47	38.93	34.88	2.04	1.60	1.91	1.69
Sequence 13	16.87	19.11	21.53	19.26	1.06	1.11	1.21	1.11
Sequence 14	64.48	50.45	55.02	51.39	3.37	2.37	2.49	2.39
Sequence 15	18.14	19.42	21.22	19.73	1.26	1.11	1.02	1.08
Sequence 16	24.84	15.20	16.75	15.36	1.45	0.84	0.85	0.82
Sequence 17	45.01	33.73	35.37	34.10	2.17	1.38	1.44	1.38
Sequence 18	32.58	28.13	36.96	30.00	1.97	1.60	2.13	1.74
Sequence 19	67.17	51.51	49.65	50.68	4.69	3.36	3.14	3.29
Sequence 20	24.27	16.58	19.05	16.96	1.41	0.87	0.72	0.81
Sequence 21	47.44	55.87	56.76	55.93	2.89	3.66	3.85	3.72
Sequence 22	22.32	24.98	27.86	25.38	1.19	1.23	1.24	1.23
Sequence 23	19.56	12.73	15.06	13.05	1.20	0.75	0.82	0.76

Sequence 24	24.27	24.31	27.12	24.60	1.60	1.43	1.64	1.46
Sequence 25	50.94	49.25	49.90	49.24	3.06	3.03	3.11	3.05
Sequence 26	17.32	20.44	19.47	19.46	1.10	1.02	0.95	0.97
Sequence 27	25.51	21.31	27.96	22.74	1.45	1.21	1.47	1.27
Sequence 28	19.48	23.70	26.33	24.20	1.26	1.52	1.62	1.53
Sequence 29	27.53	18.49	20.78	18.80	1.54	0.94	0.97	0.93
Sequence 30	26.81	20.94	20.99	20.30	1.35	0.87	0.94	0.87
Sequence 31	37.44	25.95	25.53	25.59	1.94	1.28	1.26	1.27
Sequence 32	33.69	28.88	27.24	28.17	1.54	1.49	1.41	1.46
Sequence 33	19.07	21.44	22.42	21.03	1.27	1.25	1.27	1.23
Sequence 34	17.53	23.10	23.64	22.53	1.22	1.51	1.40	1.46
Sequence 35	39.86	30.81	32.39	31.14	2.04	1.42	1.44	1.42
Sequence 36	48.77	37.53	38.14	37.49	2.48	1.63	1.70	1.62
Sequence 37	61.25	47.71	48.03	47.24	4.21	3.36	3.33	3.34
Sequence 38	24.29	24.94	23.37	23.99	1.60	1.41	1.33	1.36
Sequence 39	38.78	37.27	41.94	38.25	2.45	1.91	2.20	1.96
Sequence 40	21.00	13.64	13.84	13.43	1.28	0.83	0.74	0.80
Sequence 41	82.73	76.34	78.55	76.71	4.01	3.84	3.98	3.85
Sequence 42	17.78	23.12	26.84	23.65	1.23	1.54	1.67	1.56
Sequence 43	29.87	34.76	35.51	34.31	2.04	2.38	2.40	2.38
Sequence 44	57.59	53.27	49.90	52.05	2.81	2.72	2.61	2.68
Sequence 45	32.83	13.61	13.59	12.77	1.61	0.67	0.72	0.63
Sequence 46	19.29	21.19	22.13	21.12	1.27	1.27	1.26	1.24
Sequence 47	25.58	18.10	20.60	18.62	1.40	0.91	0.97	0.92
Sequence 48	30.73	26.54	26.66	25.92	1.70	1.23	1.27	1.23
Sequence 49	21.38	13.90	16.82	14.00	1.20	0.77	0.81	0.75

Sequence 50	38.55	36.12	33.36	34.39	1.99	2.12	1.87	2.03
Sequence 51	22.17	20.33	23.94	21.22	1.38	0.88	0.83	0.86
Sequence 52	24.95	21.41	21.47	20.83	1.31	1.12	1.12	1.09
Sequence 53	25.02	27.69	27.00	26.99	1.65	1.77	1.65	1.72
Sequence 54	23.15	28.85	31.76	29.29	1.50	1.55	1.70	1.57
Sequence 55	36.97	36.30	40.07	36.83	2.24	2.08	2.22	2.09
Sequence 56	22.69	14.50	15.38	14.58	1.29	0.72	0.68	0.70
Sequence 57	37.45	32.36	36.26	33.14	2.23	2.20	2.40	2.26
Sequence 58	39.13	25.96	27.19	26.11	2.07	1.23	1.24	1.21
Sequence 59	55.81	45.11	49.50	46.06	3.57	2.50	2.75	2.54
Sequence 60	33.30	18.13	17.65	17.70	1.74	0.86	0.80	0.83
Sequence 61	28.09	18.77	23.68	19.71	1.78	0.96	1.00	0.95
Sequence 62	57.37	53.13	52.83	52.97	2.79	2.49	2.52	2.48
Sequence 63	42.60	27.64	27.68	26.57	2.62	1.51	1.49	1.46
Sequence 64	51.95	60.87	60.18	60.49	2.63	2.87	2.71	2.82
Sequence 65	34.21	23.74	28.59	24.67	2.18	1.45	1.73	1.52
Sequence 66	46.16	30.51	33.47	30.95	2.87	1.92	2.06	1.94
Sequence 67	24.96	21.56	27.10	22.92	1.43	1.10	1.29	1.13
Sequence 68	40.35	36.71	39.51	36.84	2.37	2.15	2.19	2.15
Sequence 69	56.89	50.92	50.99	50.67	2.88	2.66	2.53	2.61
Sequence 70	56.85	61.65	65.20	62.25	2.91	3.52	3.78	3.57
Sequence 71	39.53	36.56	40.29	37.49	1.95	1.58	1.70	1.59
Sequence 72	64.87	44.57	43.25	43.58	4.22	2.83	2.69	2.78
Sequence 73	41.09	53.70	54.30	53.59	2.41	3.05	3.01	3.03
Sequence 74	50.74	46.97	46.89	46.29	2.23	2.57	2.48	2.53
Sequence 75	65.60	38.69	42.99	39.54	3.96	1.94	2.06	1.93

Sequence 76	38.46	26.39	30.91	27.41	1.94	1.15	1.38	1.19
Sequence 77	16.32	15.92	18.04	16.06	1.08	0.85	0.92	0.85
Sequence 78	26.48	40.91	42.62	40.85	1.45	2.50	2.53	2.49
Sequence 79	42.06	18.81	18.60	18.45	2.08	0.88	0.91	0.88
Sequence 80	56.27	37.28	43.81	39.02	3.43	1.82	2.08	1.88
Sequence 81	48.03	38.59	43.74	39.63	2.51	1.96	2.33	2.05
Sequence 82	21.35	19.65	23.37	20.41	1.32	0.87	0.84	0.83
Sequence 83	24.91	18.74	20.01	18.54	1.43	1.14	1.17	1.12
Sequence 84	17.25	24.82	25.99	24.49	1.13	1.55	1.61	1.54
Sequence 85	19.90	13.17	12.90	12.35	1.06	0.66	0.62	0.61
Sequence 86	28.97	27.46	30.52	27.85	1.72	1.67	1.77	1.68
Sequence 87	22.07	19.04	20.92	19.34	1.33	0.99	0.99	0.98
Sequence 88	17.13	18.63	21.07	18.90	1.03	1.15	1.25	1.15
Sequence 89	16.57	23.89	24.13	23.50	1.13	1.55	1.51	1.53
Sequence 90	25.89	29.77	33.49	30.49	1.74	1.87	2.05	1.92
Sequence 91	29.30	17.50	20.04	17.58	1.28	0.90	1.01	0.91
Sequence 92	49.05	38.23	41.24	38.60	2.45	1.75	1.95	1.81
Sequence 93	19.32	15.78	18.07	15.84	1.13	0.76	0.89	0.77
Sequence 94	18.32	21.47	23.32	21.66	1.24	1.30	1.40	1.32
Sequence 95	23.91	19.21	22.76	19.57	1.28	0.81	0.91	0.81
Sequence 96	45.58	32.09	31.40	31.53	2.24	1.41	1.40	1.40
Sequence 97	23.63	18.11	18.72	18.08	1.42	0.96	0.85	0.91
Sequence 98	24.06	17.45	16.15	16.89	1.15	0.88	0.77	0.84
Sequence 99	22.19	23.40	24.33	23.28	1.45	1.44	1.38	1.40
Sequence 100	43.92	37.18	34.12	35.81	2.59	2.32	2.10	2.25
Median	29.12	25.61	27.12	25.59	1.70	1.44	1.44	1.42

D.6. Main Engine Cylinder 5

The parameter analysed refers to the exhaust gas outlet temperature. A graphical representation of such a parameter is expressed in Fig. D.6.1. The descriptive statistics is also introduced in Table E.6.1.

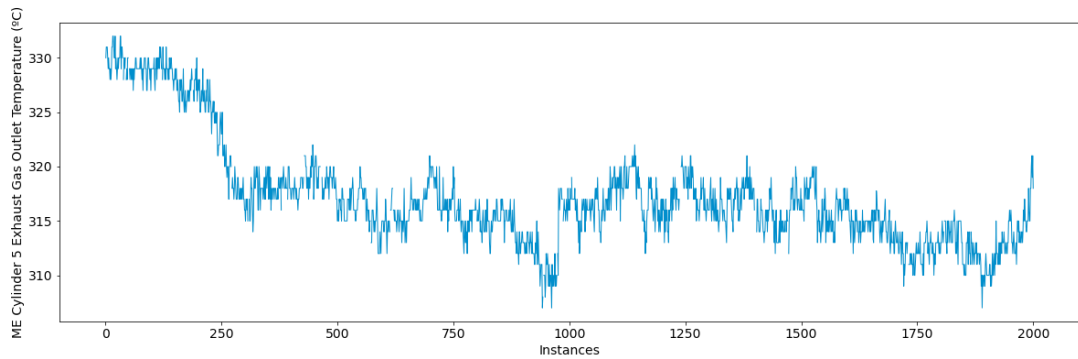


Fig. D.6.1. Graphical representation of the exhaust gas outlet temperature parameter

Table D.6.1. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
Cyl. 5 Exh. Gas Out. Temp	317.21	4.81	307	314	316	318	332

As part of the data pre-processing phase, the identification of operational states step has been implemented (see section 4.5. *A Novel Framework for the Identification of Steady States* for a comprehensive explanation of such a step). In total, only one operational state has been identified, as perceived in Fig. D.6.2.

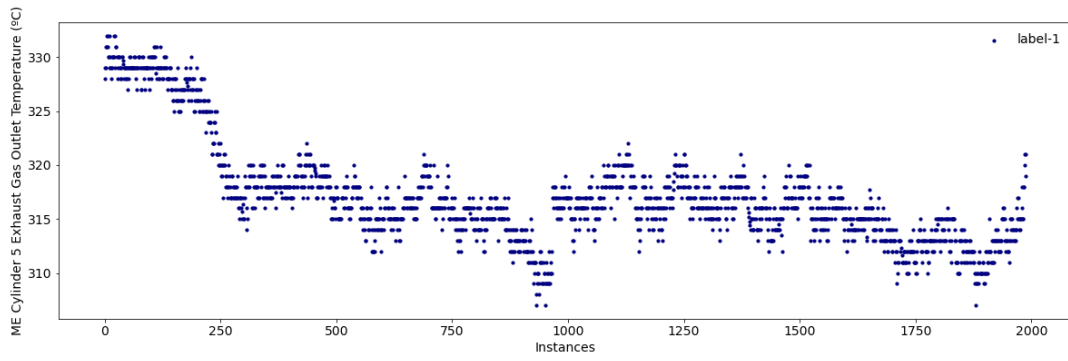
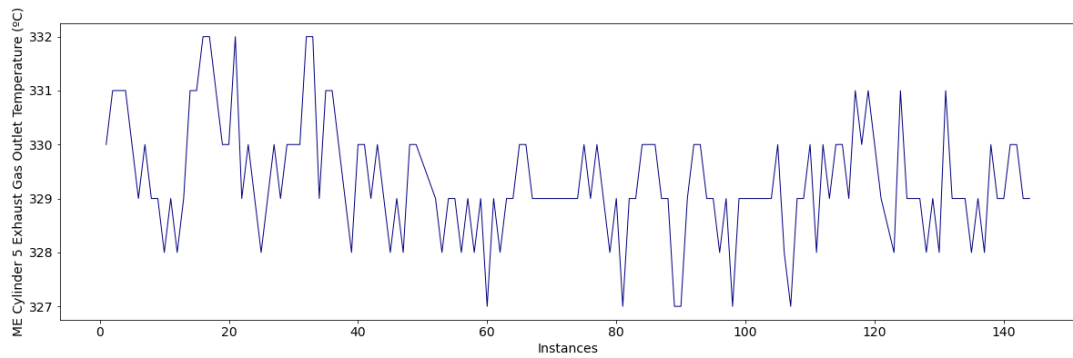


Fig. D.6.2. Identification of the operational states for the monitored parameter.

Subsequently, due to the lack of fault data, both collective anomalies and degradation patterns are simulated. Some examples are presented in Figs. E.6.4 – E.4.5. Examples of normal sequences are also introduced in Fig. D.6.3.



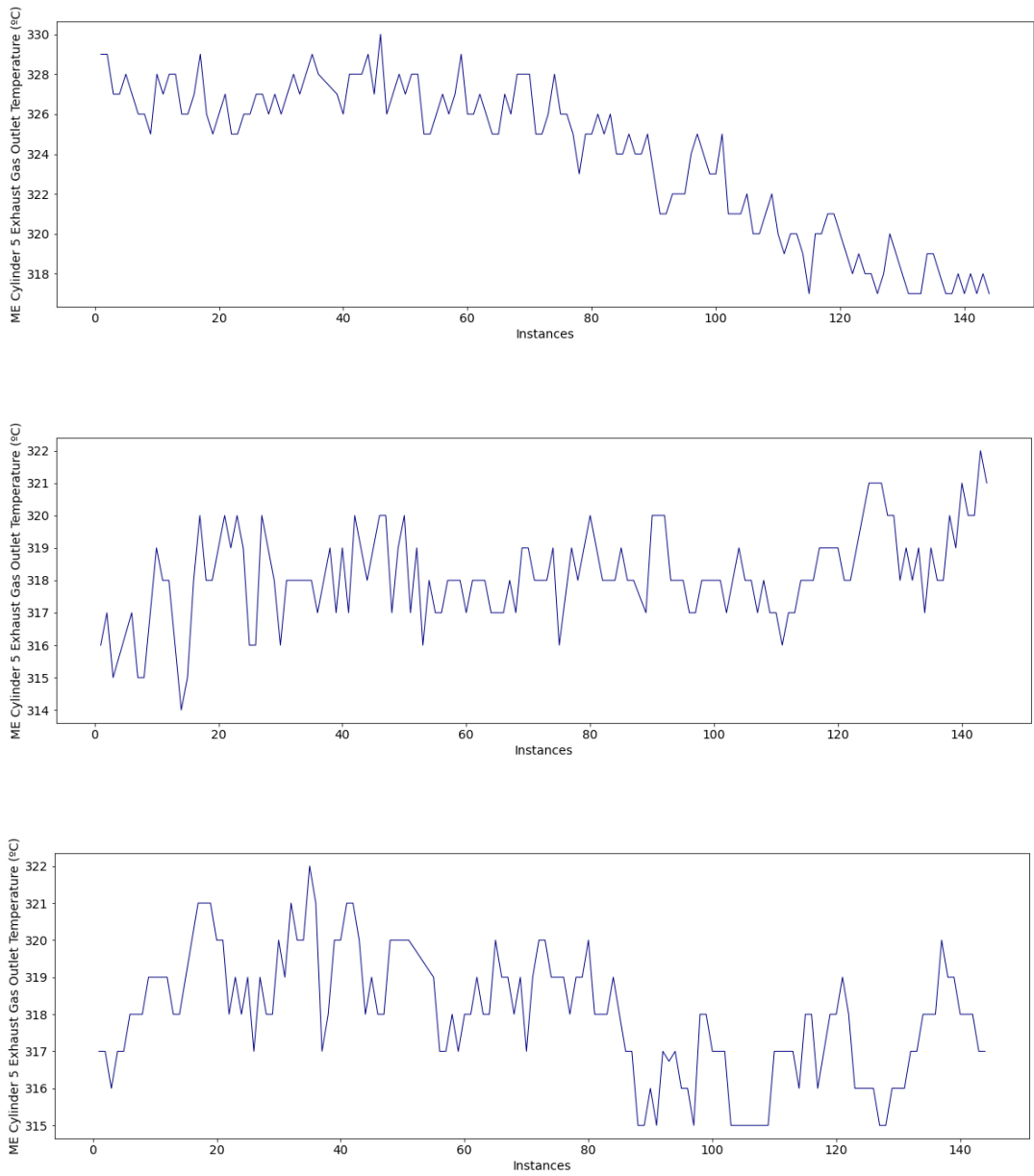


Fig. D.6.3. Example of normal sequences.

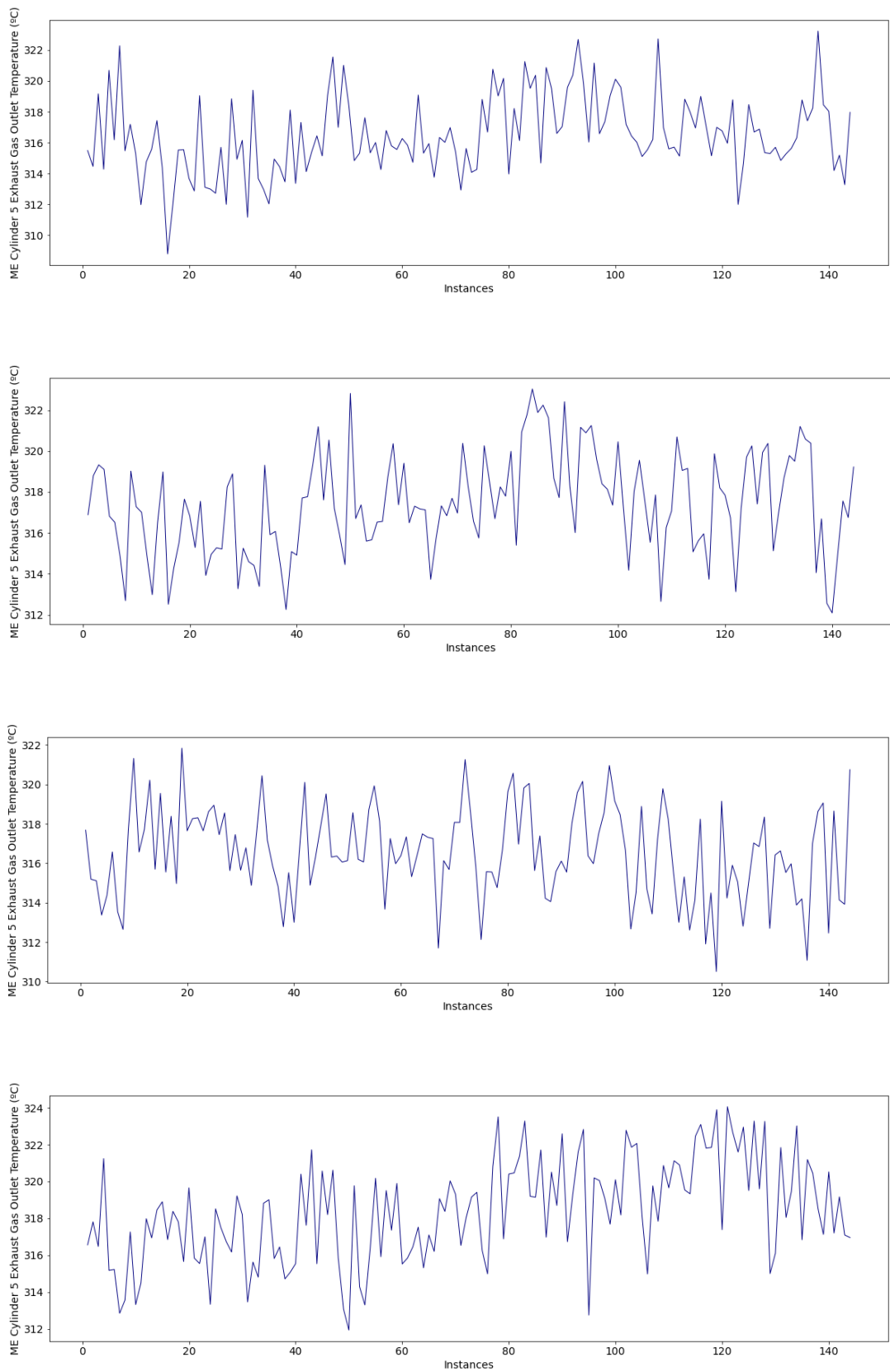


Fig. D.6.4. Example of sequences with collective anomalies.

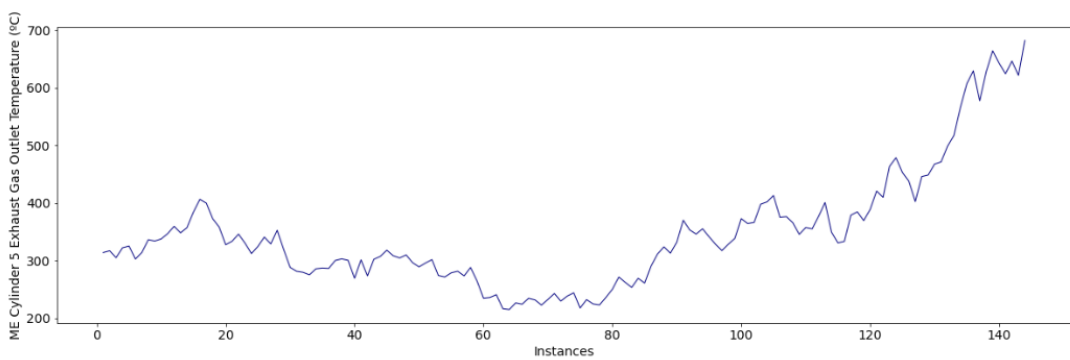
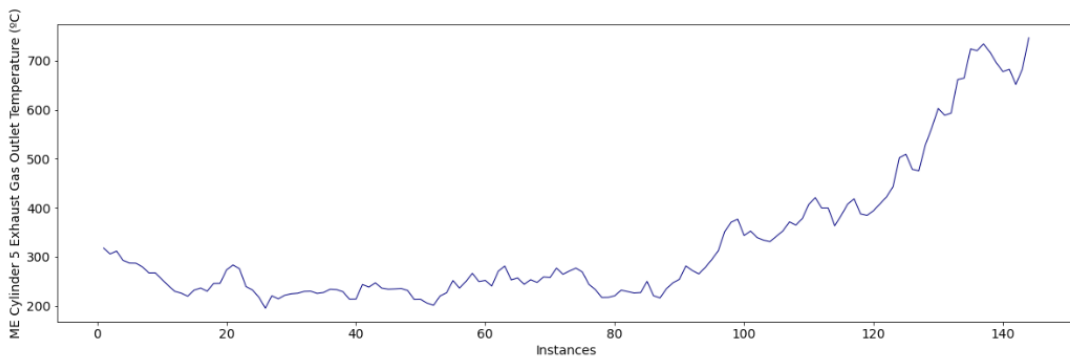
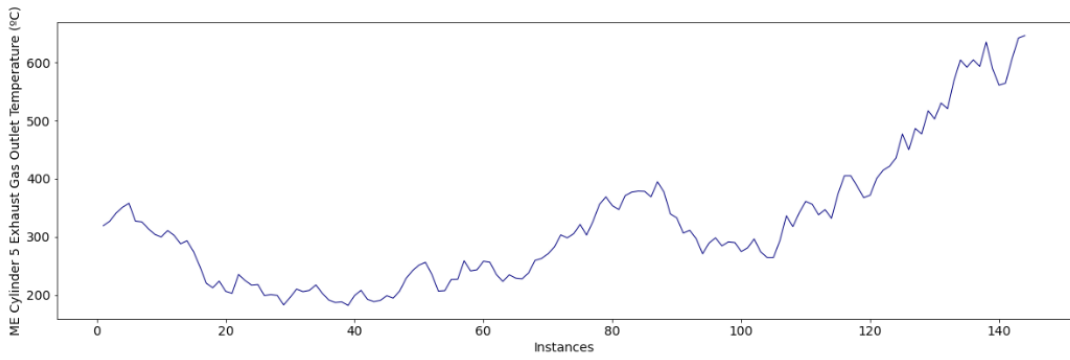
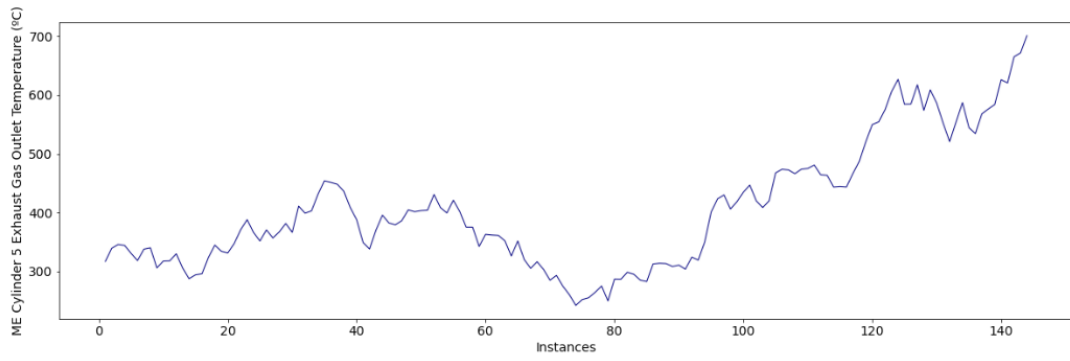


Fig. D.6.5. Example of sequences with degradation patterns.

As part of the MA framework, the subsequent module to be applied is the diagnostic analytics module. Accordingly, the fault detection step is implemented as stated in section 5.3.1. *Fault Detection*. As perceived in the histograms (Figs. E.6.6 – E.6.8), a simple threshold is adequate in this case study to distinguish the normal sequences from the abnormal sequences.

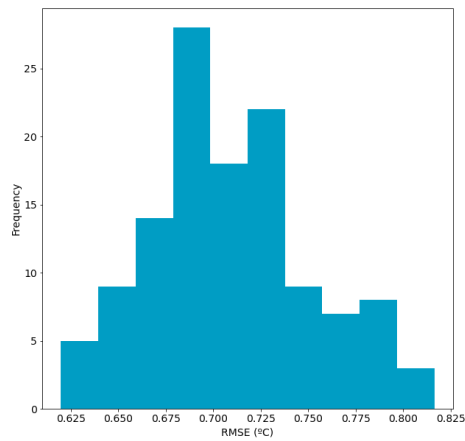


Fig. D.6.6. Histogram of the reconstructed errors of the normal sequences (test set).

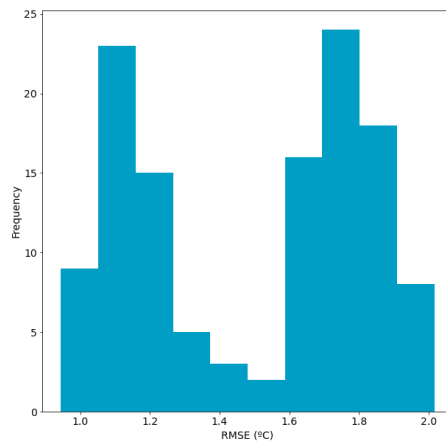


Fig. D.6.7. Histogram of the reconstructed errors of the sequences with collective anomalies.

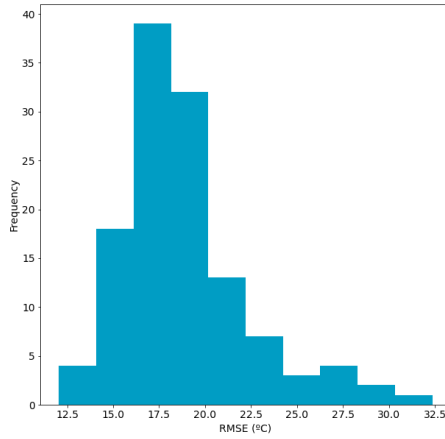
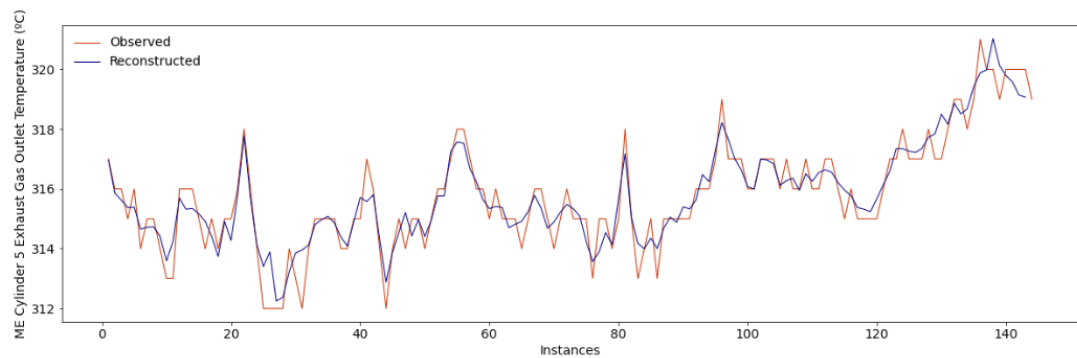
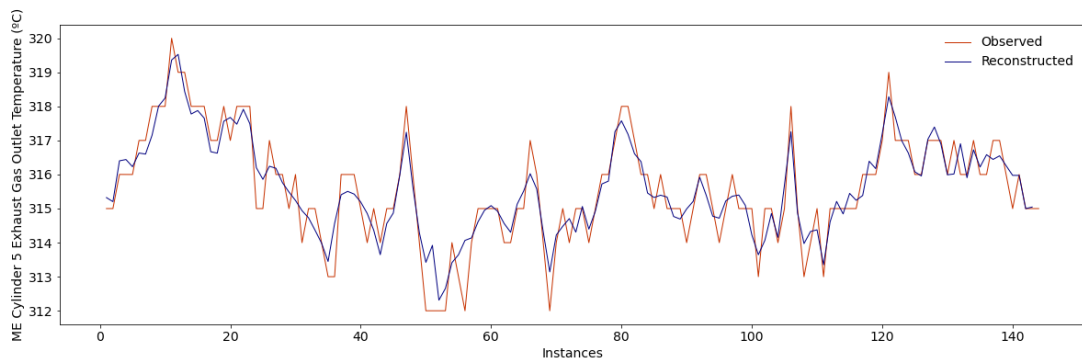


Fig. D.6.8. Histogram of the reconstructed errors of the sequences with degradation patterns.

Examples of reconstructed sequences are also introduced in Figs. E.6.9 – E.6.11.



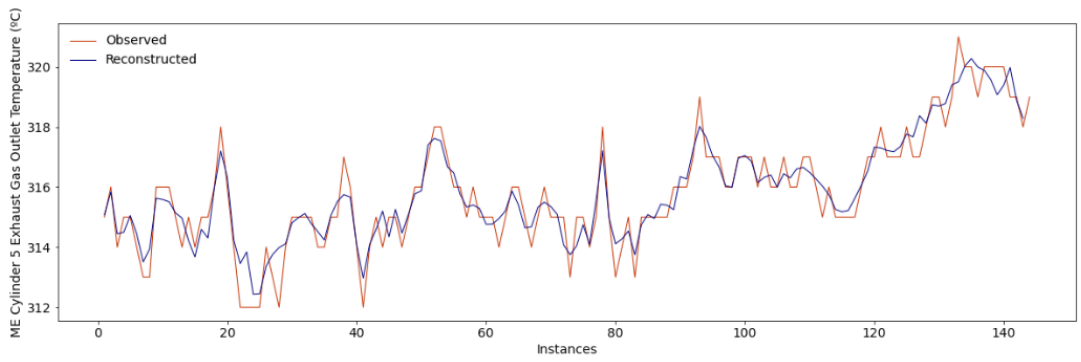
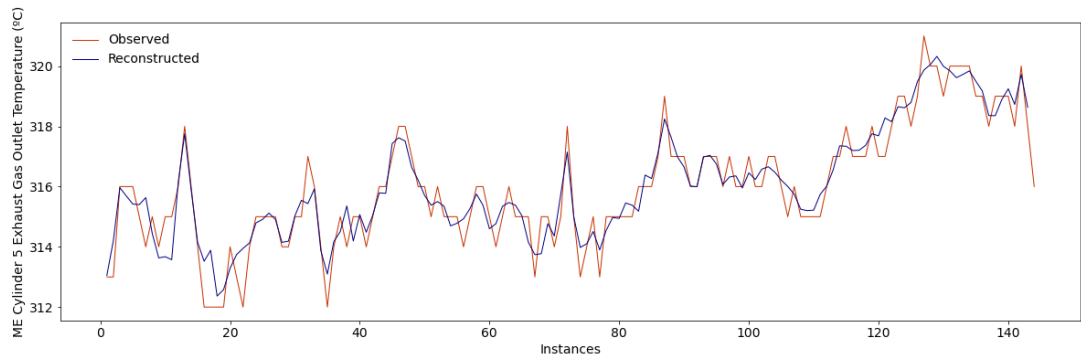


Fig. D.6.9. Example of normal reconstructed sequences (test set).

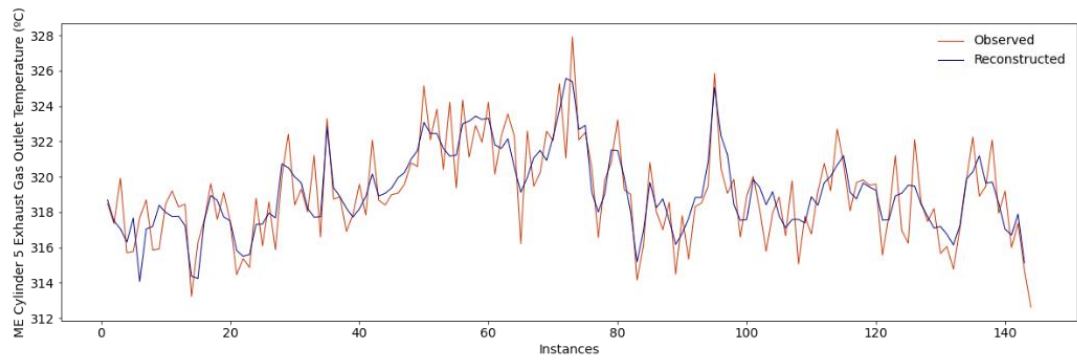




Fig. D.6.10. Example of reconstructed sequences with collective anomalies.

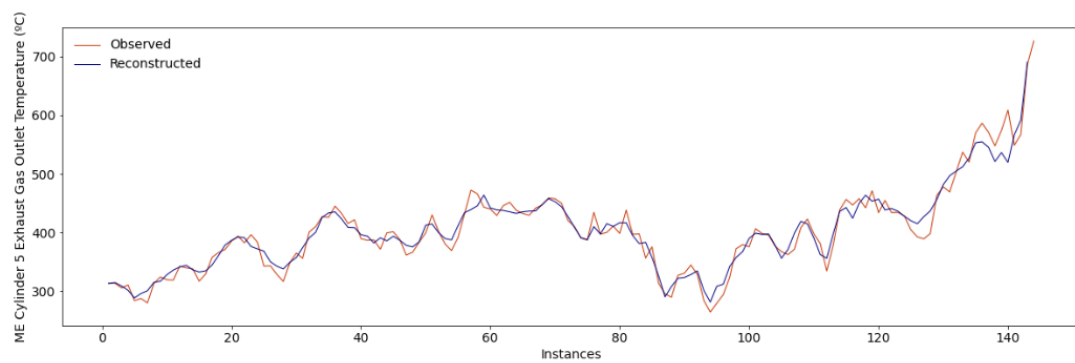
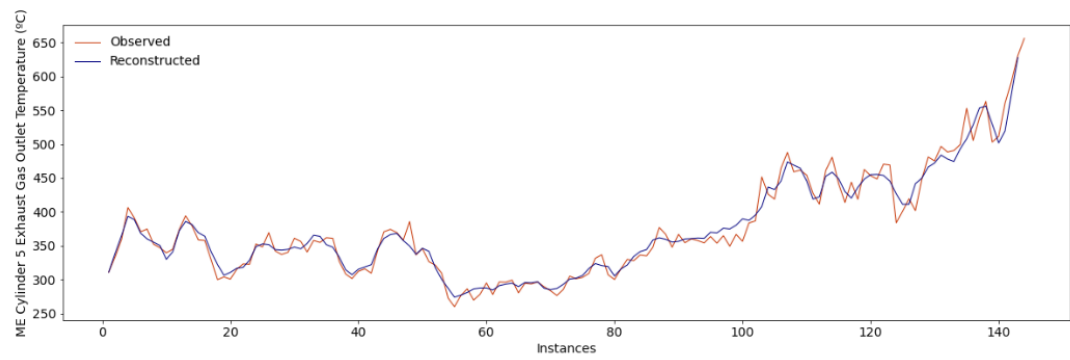
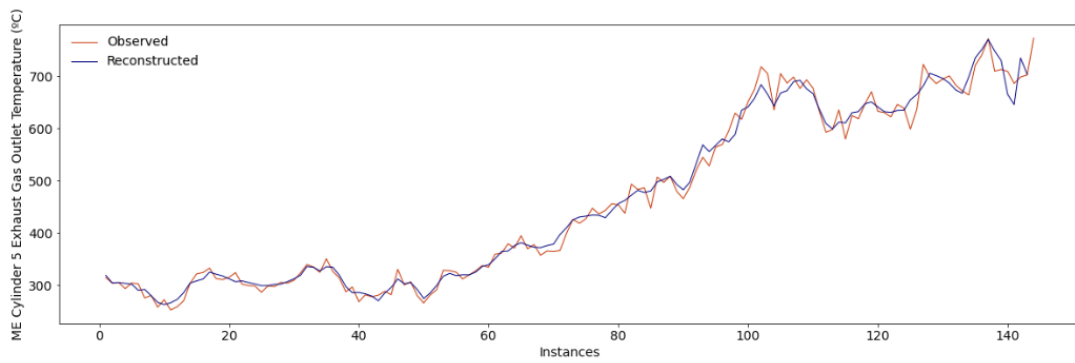
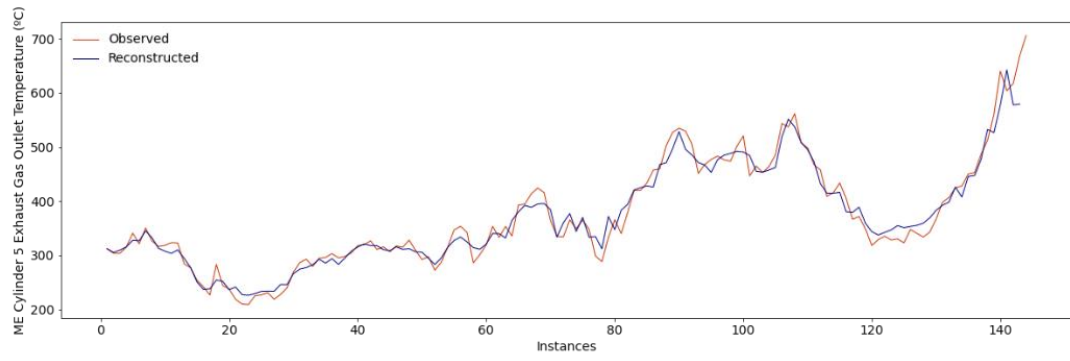


Fig. D.6.11. Example of reconstructed sequences with degradation patterns.

Those sequences detected as anomalous are then classified into two categories: sequences with collective anomalies, and sequences with degradation patterns. Accordingly, as the implemented approach refers to a time series imaging approach (see section 5.3.2. *Fault Identification*), the anomalous sequences detected are transformed into images. Examples of these can be perceived in Figs. E.6.12 – E.6.13.



Fig. D.6.12. Images with collective anomalies.

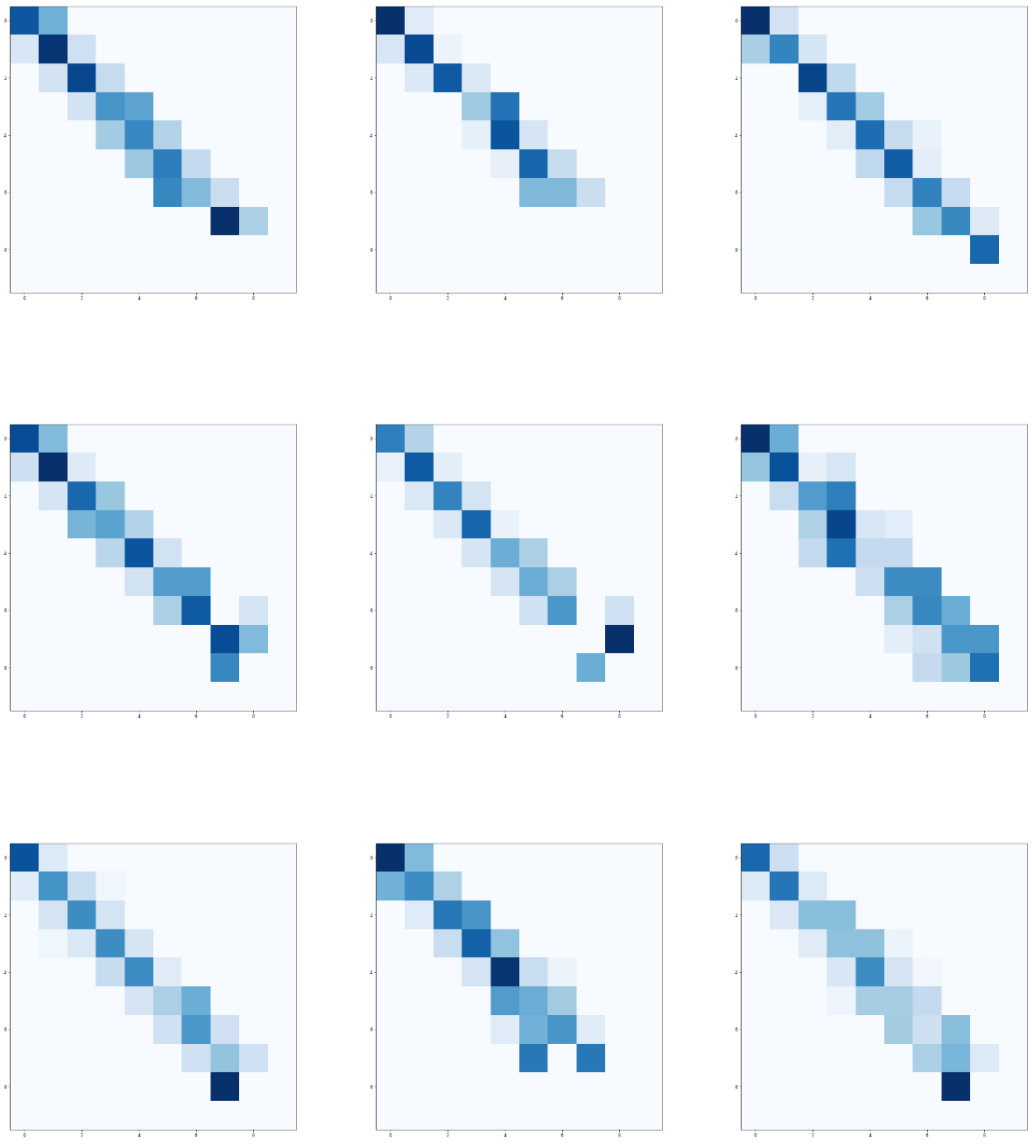
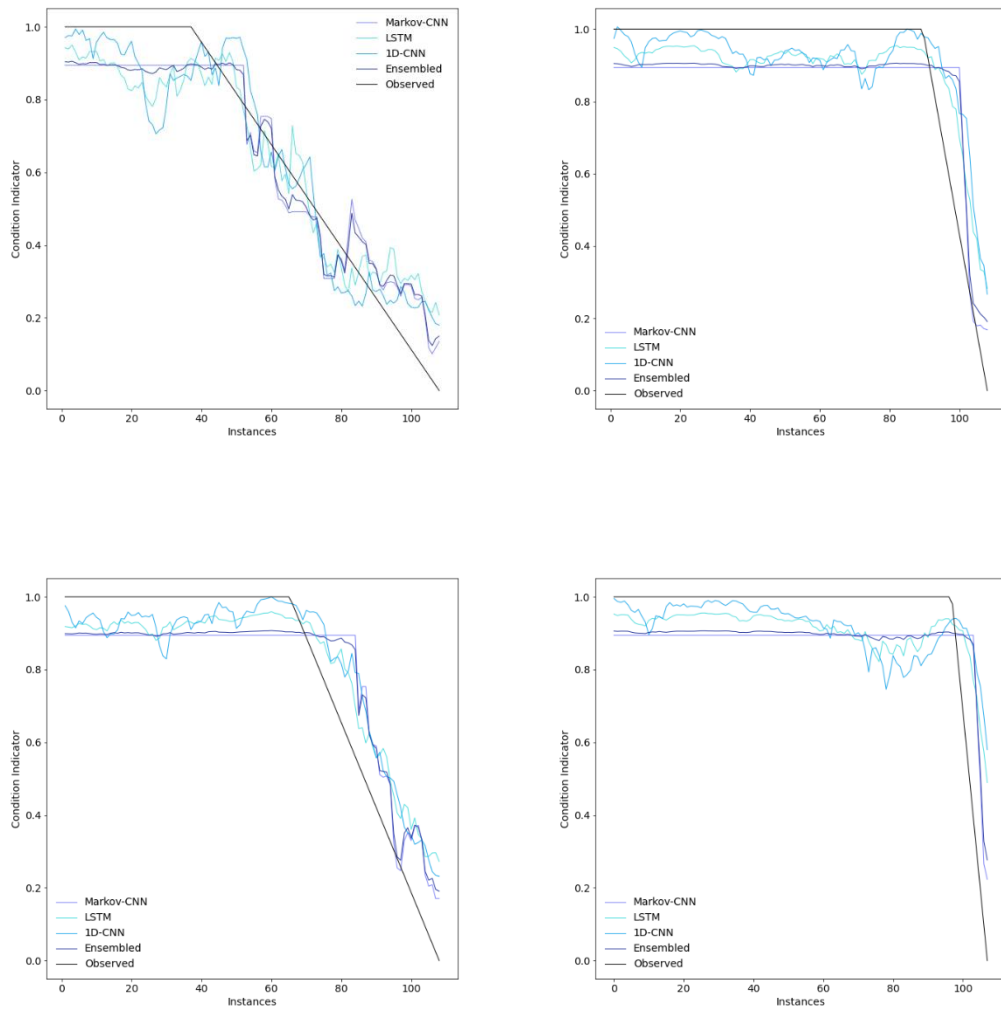


Fig. D.6.13. Images with degradation patterns.

As it can be perceived, the two categories can be easily distinguished. This aspect facilitated the achievement of the maximum accuracy score in this process, thus classifying all the images adequately.

By adequately classifying such images, the sequences with degradation patterns are selected so that the RUL can be predicted. Examples of such a prediction are presented in Fig. D.6.14. The RMSE and Maintenance Score of the first 100 sequences are also presented in Table E.6.2.



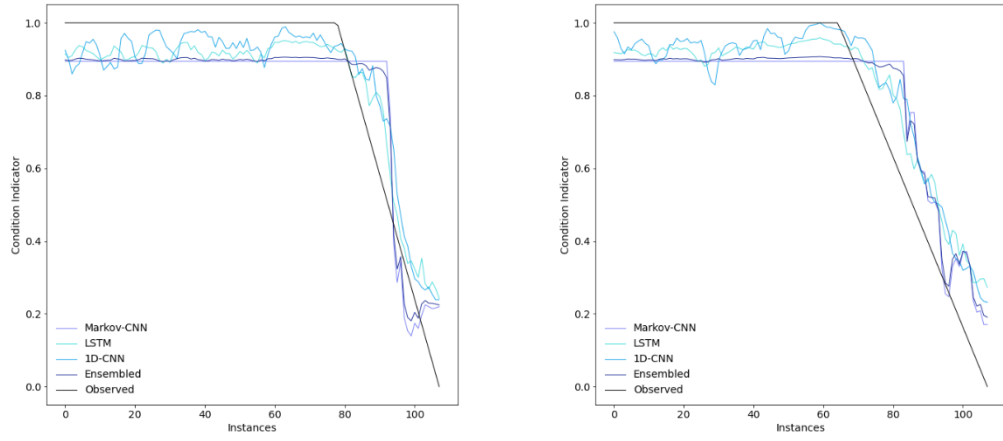


Fig. D.6.14. Examples of condition indicator.

Table D.6.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.

Sequences	RMSE (Minutes)				Maintenance Score			
	Markov-CNN	LSTM	1D-CNN	Ensembled	Markov-CNN	LSTM	1D-CNN	Ensembled
Sequence 1	66.23	50.37	50.37	62.82	4.16	3.18	3.00	3.96
Sequence 2	20.61	21.75	19.49	20.31	1.19	1.15	1.03	1.17
Sequence 3	54.35	42.02	45.21	50.86	3.55	2.68	2.89	3.27
Sequence 4	14.65	19.28	19.94	14.98	1.02	1.19	1.19	1.03
Sequence 5	19.47	20.71	21.67	19.19	1.20	1.22	1.24	1.20
Sequence 6	65.91	54.73	57.20	62.24	3.52	2.49	2.61	3.22
Sequence 7	21.64	21.66	20.49	20.84	1.46	1.43	1.23	1.42
Sequence 8	39.59	26.57	26.04	36.41	1.96	1.39	1.35	1.82
Sequence 9	41.35	50.50	51.77	42.31	1.98	2.66	2.63	2.00
Sequence 10	59.61	46.27	49.13	55.63	3.72	3.19	3.20	3.59
Sequence 11	19.29	16.65	16.79	18.48	1.18	0.95	0.89	1.13
Sequence 12	60.34	44.11	46.98	56.20	4.07	2.79	2.87	3.80
Sequence 13	43.61	44.58	36.30	43.61	2.21	2.20	1.75	2.20
Sequence 14	55.12	45.23	42.33	53.01	2.68	2.17	1.99	2.57
Sequence 15	32.02	29.06	23.47	31.21	1.61	1.48	1.12	1.58
Sequence 16	14.39	17.02	16.15	14.39	0.92	1.01	0.89	0.91
Sequence 17	54.99	48.40	43.96	53.53	2.75	2.47	2.24	2.69
Sequence 18	56.25	44.94	48.60	51.98	3.44	2.66	2.82	3.21
Sequence 19	21.33	29.65	31.83	22.36	1.29	1.41	1.42	1.29
Sequence 20	31.98	32.10	29.07	31.61	1.63	1.52	1.29	1.60
Sequence 21	40.52	20.33	20.74	35.72	2.44	1.20	0.77	2.19
Sequence 22	23.47	18.33	17.16	21.65	1.28	0.93	0.86	1.17
Sequence 23	21.01	18.79	16.29	19.93	1.13	0.93	0.80	1.05

Sequence 24	43.49	30.40	30.17	40.34	2.12	1.59	1.53	2.00
Sequence 25	36.77	21.13	20.10	32.89	2.23	1.23	1.09	2.03
Sequence 26	19.40	15.18	17.13	18.15	1.19	0.91	1.05	1.13
Sequence 27	18.53	13.81	13.79	17.26	1.19	0.86	0.77	1.10
Sequence 28	59.13	38.68	42.26	54.37	3.74	2.56	2.68	3.50
Sequence 29	22.15	13.27	11.86	19.89	1.21	0.71	0.58	1.10
Sequence 30	35.79	21.72	19.68	32.84	1.71	1.09	1.02	1.59
Sequence 31	21.66	23.20	24.27	21.39	1.18	1.13	1.11	1.17
Sequence 32	28.82	20.12	16.79	26.87	1.44	1.01	0.77	1.34
Sequence 33	56.92	52.67	60.60	55.71	3.69	3.46	4.03	3.64
Sequence 34	44.10	21.83	18.70	39.39	2.14	1.10	0.87	1.92
Sequence 35	19.97	19.07	19.96	19.48	1.28	1.13	1.10	1.25
Sequence 36	27.46	26.82	20.05	26.93	1.29	1.29	0.91	1.27
Sequence 37	33.05	21.53	18.62	30.08	1.56	1.19	0.99	1.48
Sequence 38	30.12	26.50	30.25	29.24	1.55	1.22	1.14	1.48
Sequence 39	70.34	55.63	57.06	67.13	4.17	3.52	3.46	4.04
Sequence 40	28.78	26.15	23.50	27.07	1.41	1.29	0.99	1.37
Sequence 41	17.63	18.70	19.50	17.12	1.10	1.04	0.98	1.08
Sequence 42	31.15	47.30	51.68	32.53	1.46	2.28	2.56	1.59
Sequence 43	32.19	29.77	32.58	30.87	2.02	1.94	1.97	1.99
Sequence 44	51.06	54.23	53.70	51.23	3.27	3.39	3.03	3.29
Sequence 45	18.11	18.05	18.77	17.36	1.18	0.97	0.88	1.12
Sequence 46	68.67	37.34	41.26	60.80	3.83	2.46	2.73	3.54
Sequence 47	22.53	17.49	15.28	20.58	1.01	0.90	0.82	0.92
Sequence 48	28.80	26.15	23.44	28.06	1.53	1.30	1.09	1.48
Sequence 49	26.56	34.50	30.12	27.36	1.40	1.57	1.35	1.43

Sequence 50	22.46	19.64	19.08	21.49	1.21	1.00	0.86	1.16
Sequence 51	60.81	38.78	41.12	55.59	3.75	2.39	2.36	3.45
Sequence 52	50.59	42.82	43.85	48.51	2.81	2.74	2.68	2.79
Sequence 53	76.52	45.15	48.33	68.15	4.72	2.52	2.67	4.25
Sequence 54	38.50	27.35	33.99	34.74	1.73	1.59	2.01	1.62
Sequence 55	25.79	22.53	21.73	24.51	1.42	1.08	0.98	1.33
Sequence 56	27.72	32.30	29.79	27.86	1.31	1.70	1.49	1.34
Sequence 57	25.86	17.91	19.75	23.67	1.53	0.97	0.92	1.41
Sequence 58	35.12	31.64	30.50	34.14	1.75	1.46	1.27	1.69
Sequence 59	30.17	15.47	14.32	26.77	1.53	0.84	0.79	1.37
Sequence 60	68.64	46.15	47.26	63.81	4.02	2.83	2.98	3.77
Sequence 61	35.88	43.00	51.46	36.95	2.40	2.95	3.53	2.50
Sequence 62	55.42	34.28	37.19	50.24	3.56	2.33	2.40	3.30
Sequence 63	21.49	31.65	37.90	23.04	1.52	2.24	2.58	1.66
Sequence 64	16.60	20.04	17.93	16.00	0.88	1.04	0.90	0.86
Sequence 65	45.86	31.70	29.55	42.31	2.92	2.11	1.84	2.76
Sequence 66	25.18	18.47	18.02	23.67	1.41	1.00	0.88	1.32
Sequence 67	20.00	18.87	20.02	19.24	1.24	0.94	0.83	1.18
Sequence 68	19.22	16.75	15.05	17.57	1.04	0.79	0.80	0.95
Sequence 69	29.31	24.89	23.85	28.08	1.46	1.19	1.05	1.39
Sequence 70	26.28	29.05	28.22	26.41	1.43	1.31	1.20	1.41
Sequence 71	22.81	22.17	23.06	22.37	1.32	1.10	1.12	1.28
Sequence 72	20.77	17.47	18.84	19.45	1.26	0.94	1.03	1.19
Sequence 73	25.64	24.10	20.33	24.91	1.36	1.24	1.02	1.33
Sequence 74	33.08	15.63	14.46	29.24	1.53	0.84	0.68	1.38
Sequence 75	30.52	28.93	29.07	28.83	1.98	1.87	1.66	1.95

Sequence 76	45.16	43.81	38.39	44.70	2.21	1.99	1.68	2.16
Sequence 77	23.54	24.06	23.01	21.87	1.15	1.15	1.09	1.02
Sequence 78	17.77	18.40	22.65	17.40	1.18	1.01	1.17	1.14
Sequence 79	26.22	18.27	17.94	24.42	1.41	0.98	0.76	1.32
Sequence 80	37.60	30.58	27.54	36.04	1.82	1.39	1.19	1.74
Sequence 81	59.90	43.38	47.20	55.79	3.83	2.84	2.93	3.62
Sequence 82	23.48	17.51	19.97	22.03	1.32	0.93	0.81	1.24
Sequence 83	23.29	28.95	31.07	24.08	1.20	1.37	1.37	1.22
Sequence 84	25.76	26.50	23.39	25.29	1.39	1.28	1.11	1.37
Sequence 85	50.78	38.10	39.37	48.06	2.82	2.29	2.16	2.71
Sequence 86	22.48	22.71	23.74	21.94	1.11	1.12	1.19	1.08
Sequence 87	25.50	22.53	26.35	24.19	1.39	1.42	1.42	1.39
Sequence 88	15.09	18.78	21.12	15.20	1.05	1.13	1.13	1.06
Sequence 89	45.00	30.17	37.45	41.62	2.12	1.43	1.60	1.97
Sequence 90	35.10	34.02	28.76	34.62	1.78	1.56	1.28	1.73
Sequence 91	20.41	17.47	17.29	19.05	1.12	0.90	0.75	1.04
Sequence 92	47.13	40.44	37.48	45.70	2.25	1.78	1.60	2.16
Sequence 93	52.60	58.74	63.04	51.81	3.16	3.65	3.81	3.24
Sequence 94	56.73	44.74	42.42	53.27	3.01	2.07	2.00	2.73
Sequence 95	51.56	39.37	40.34	48.57	3.65	2.70	2.64	3.45
Sequence 96	17.23	13.52	14.65	15.63	1.05	0.78	0.78	0.96
Sequence 97	32.58	19.20	17.60	29.63	1.62	0.98	0.80	1.49
Sequence 98	56.05	33.68	35.07	50.56	2.92	1.75	1.85	2.61
Sequence 99	19.48	19.43	22.85	19.16	1.25	1.03	1.04	1.20
Sequence 100	37.12	29.23	26.08	35.19	1.78	1.38	1.21	1.69
Median	32.02	27.35	27.54	30.59	1.61	1.39	1.24	1.59

D.7. Main Engine Cylinder 6

The parameter analysed refers to the exhaust gas outlet temperature. A graphical representation of such a parameter is expressed in Fig. D.7.1. The descriptive statistics is also introduced in Table E.7.1.

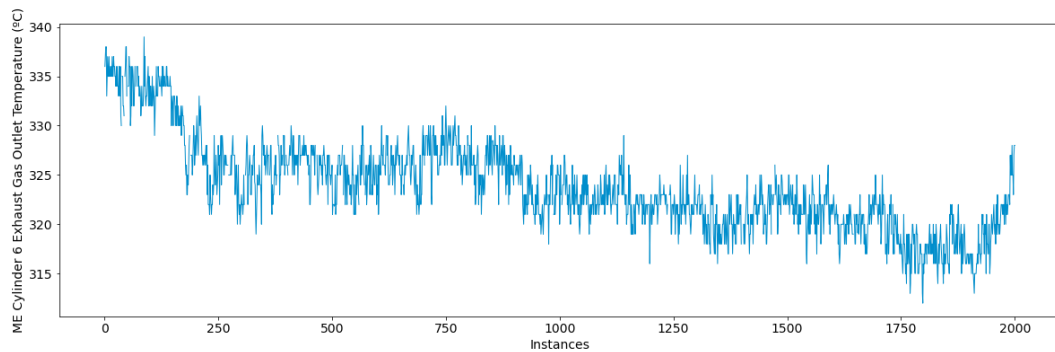


Fig. D.7.1. Graphical representation of the exhaust gas outlet temperature parameter.

Table D.7.1. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
Cyl. 6 Exh. Gas Out. Temp	323.85	4.59	312	321	323	327	339

As part of the data pre-processing phase, the identification of operational states step has been implemented (see section 4.5. *A Novel Framework for the Identification of Steady States* for a comprehensive explanation of such a step). In total, only one operational state has been identified, as perceived in Fig. D.7.2.

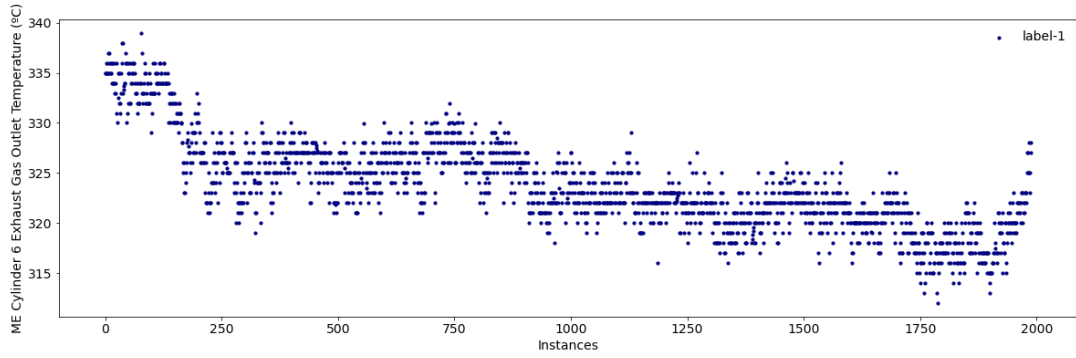
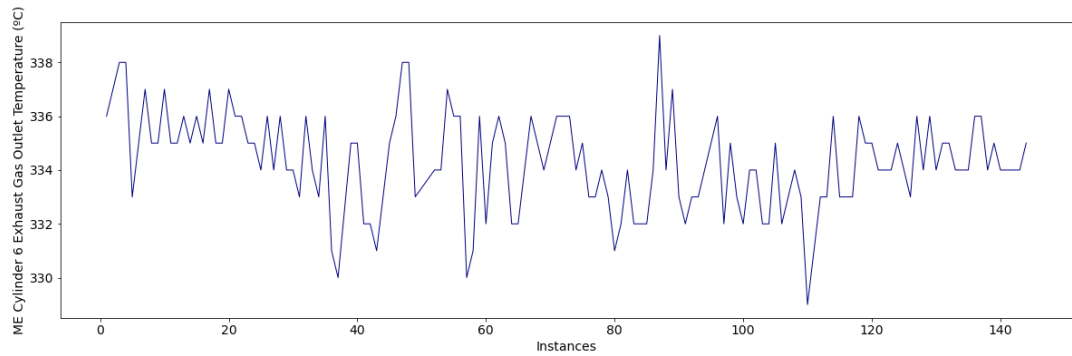


Fig. D.7.2. Identification of the operational states for the monitored parameter.

Subsequently, due to the lack of fault data, both collective anomalies and degradation patterns are simulated. Some examples are presented in Figs. E.7.4 – E.4.5. Examples of normal sequences are also introduced in Fig. D.7.3.



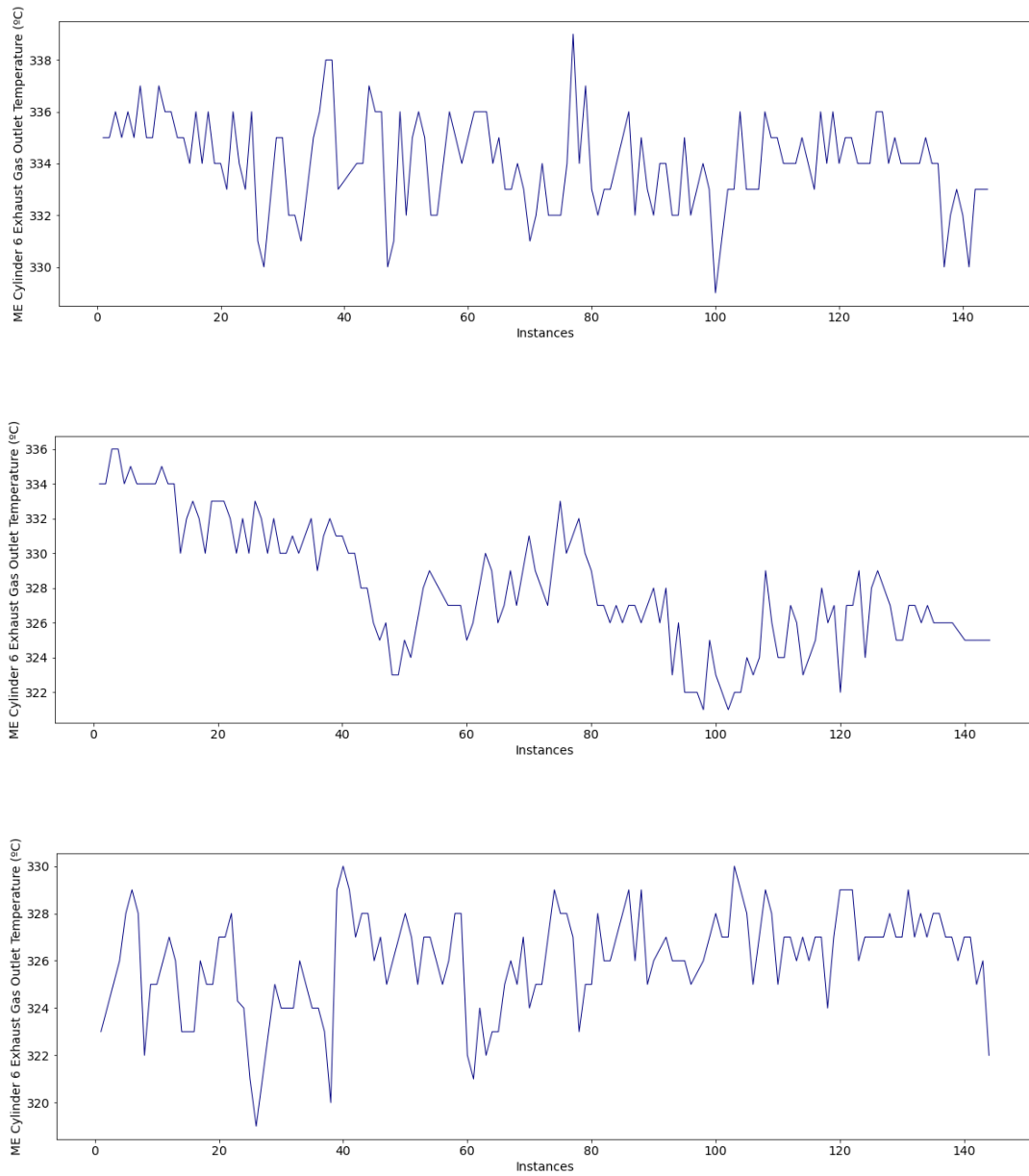


Fig. D.7.3. Example of normal sequences.

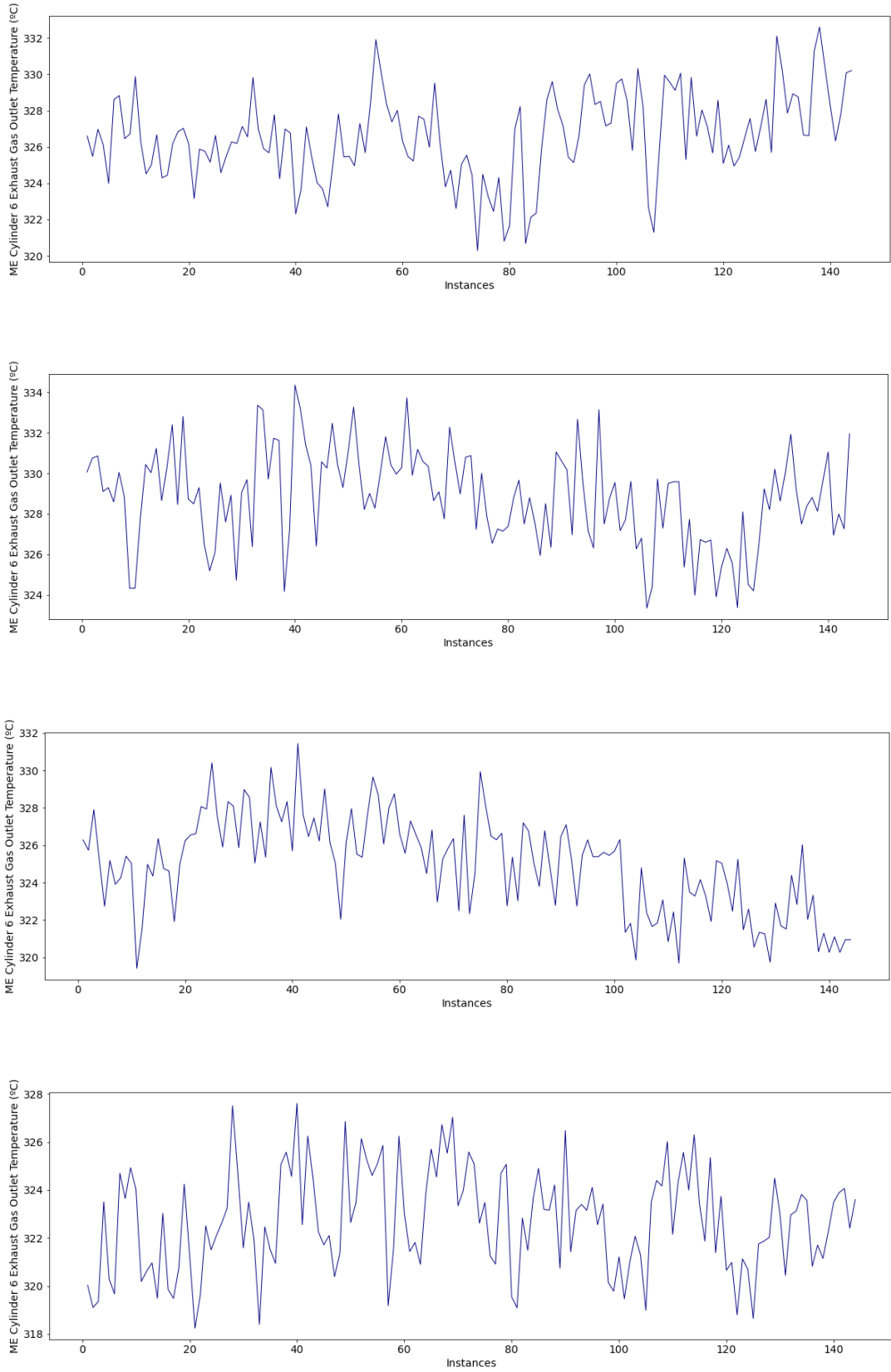


Fig. D.7.4. Example of sequences with collective anomalies.

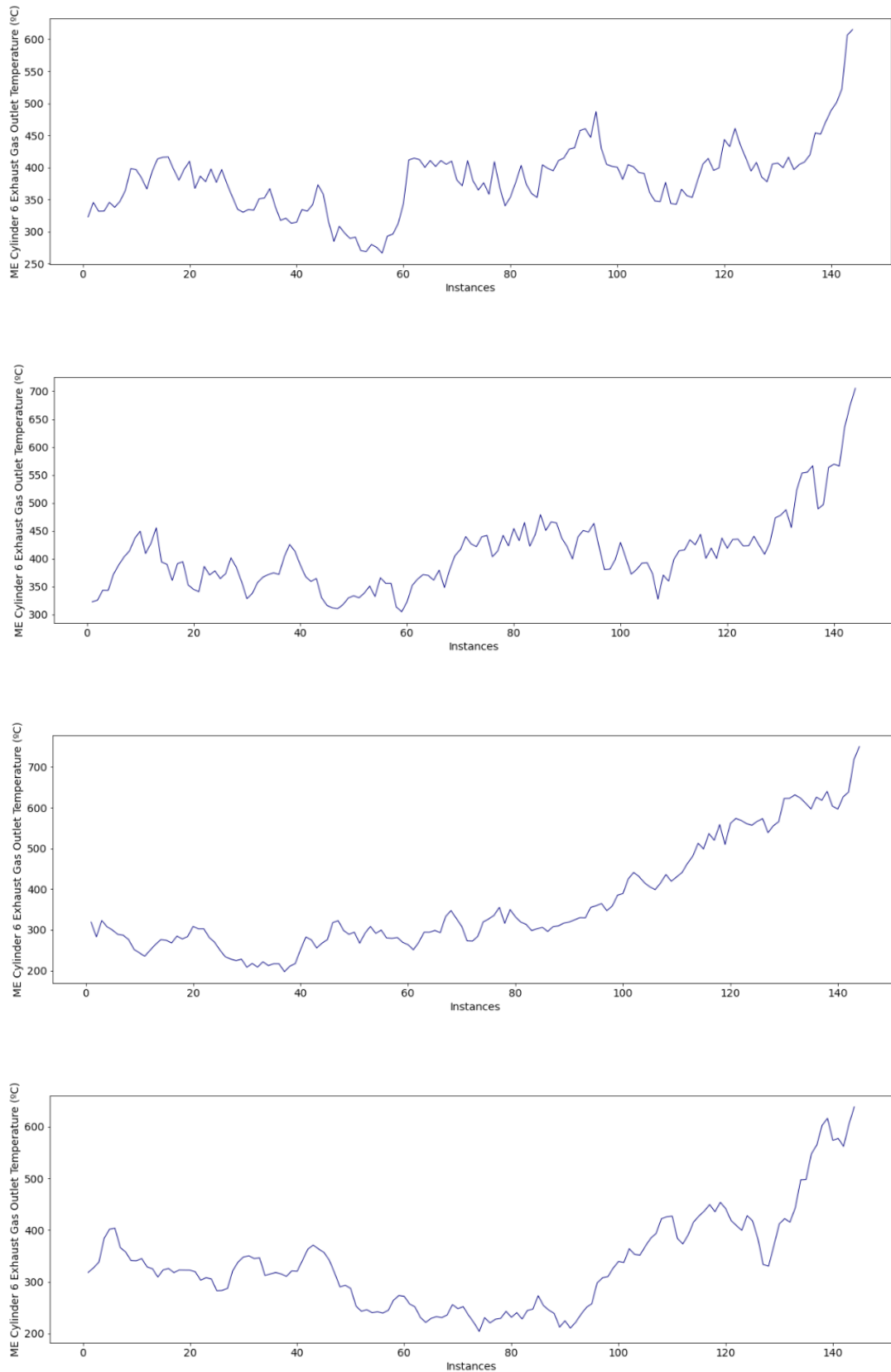


Fig. D.7.5. Example of sequences with degradation patterns.

As part of the MA framework, the subsequent module to be applied is the diagnostic analytics module. Accordingly, the fault detection step is implemented as stated in section 5.3.1. *Fault Detection*. As perceived in the histograms (Figs. E.7.6 – E.7.8), a simple threshold is adequate in this case study to distinguish the normal sequences from the abnormal sequences.

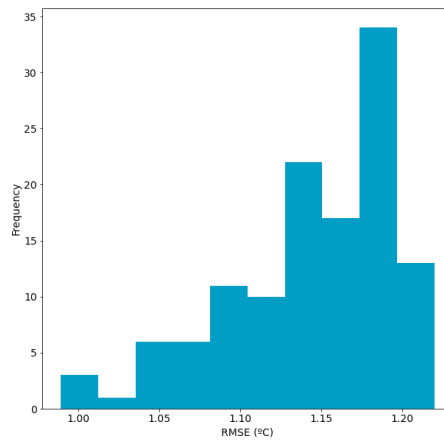


Fig. D.7.6. Histogram of the reconstructed errors of the normal sequences (test set).

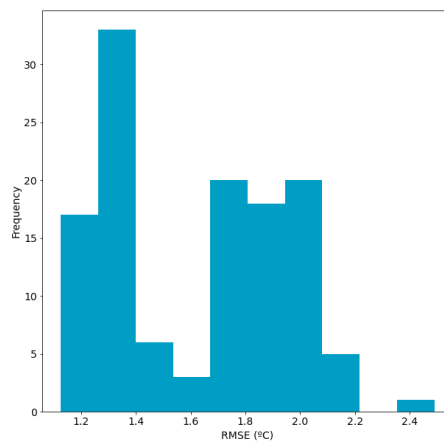


Fig. D.7.7. Histogram of the reconstructed errors of the sequences with collective anomalies.

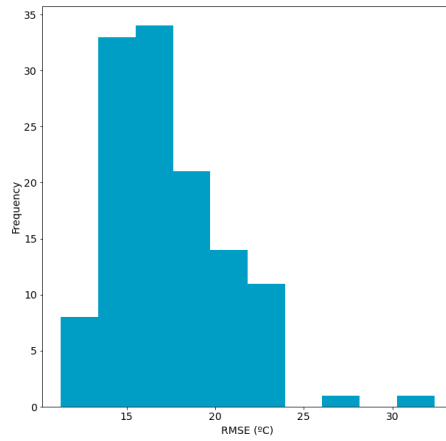
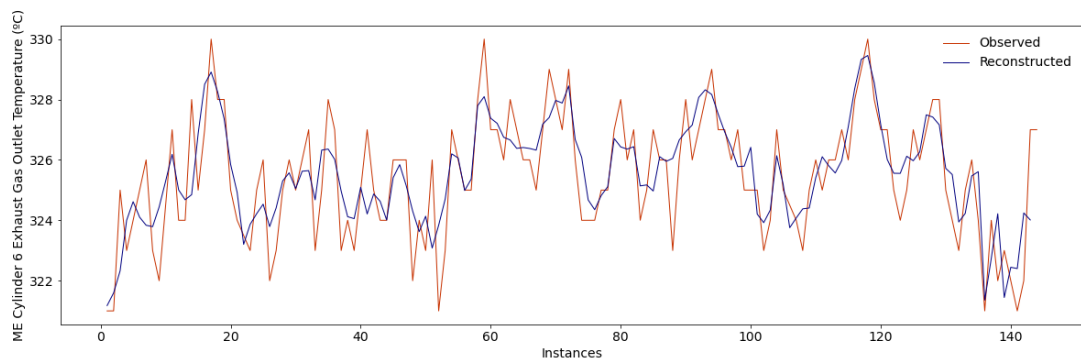
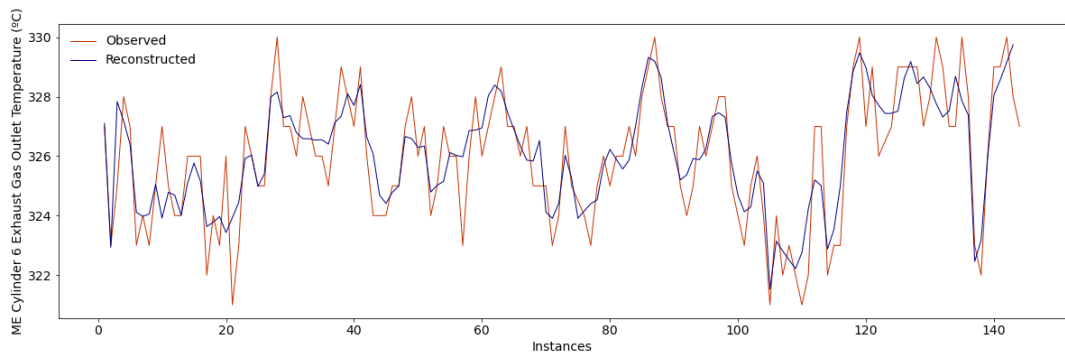


Fig. D.7.8. Histogram of the reconstructed errors of the sequences with degradation patterns.

Examples of reconstructed sequences are also introduced in Figs. E.7.9 – E.7.11.



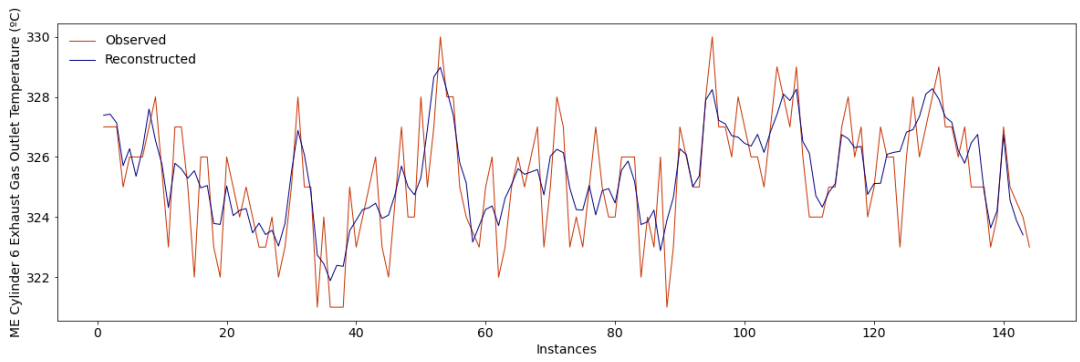
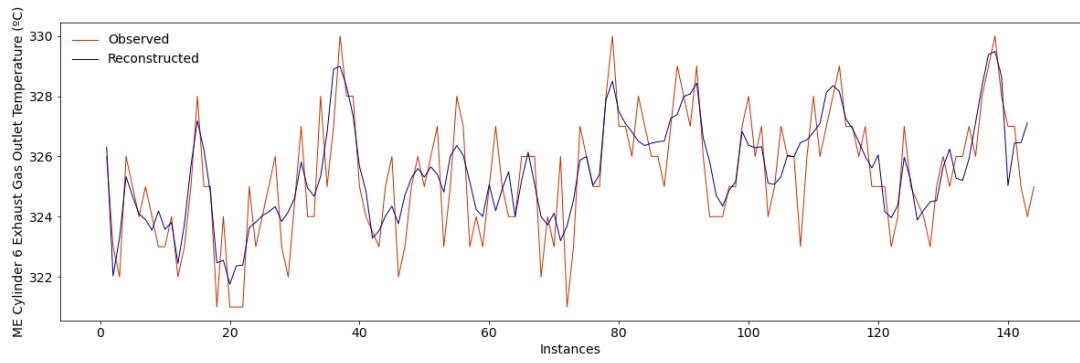
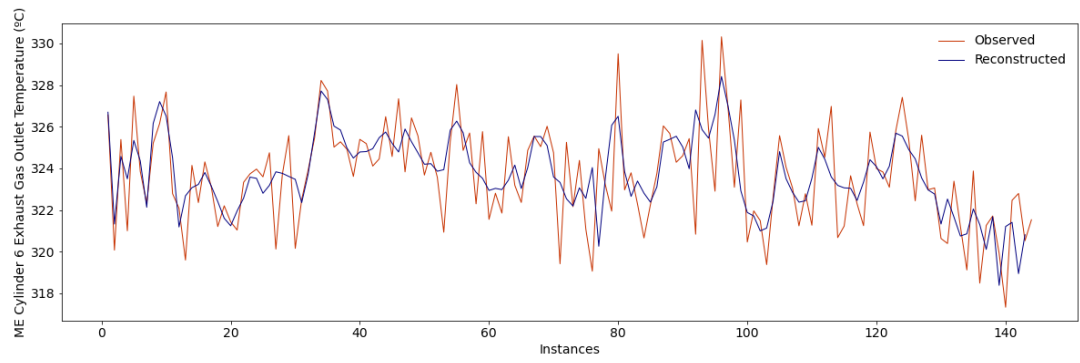


Fig. D.7.9. Example of normal reconstructed sequences (test set).



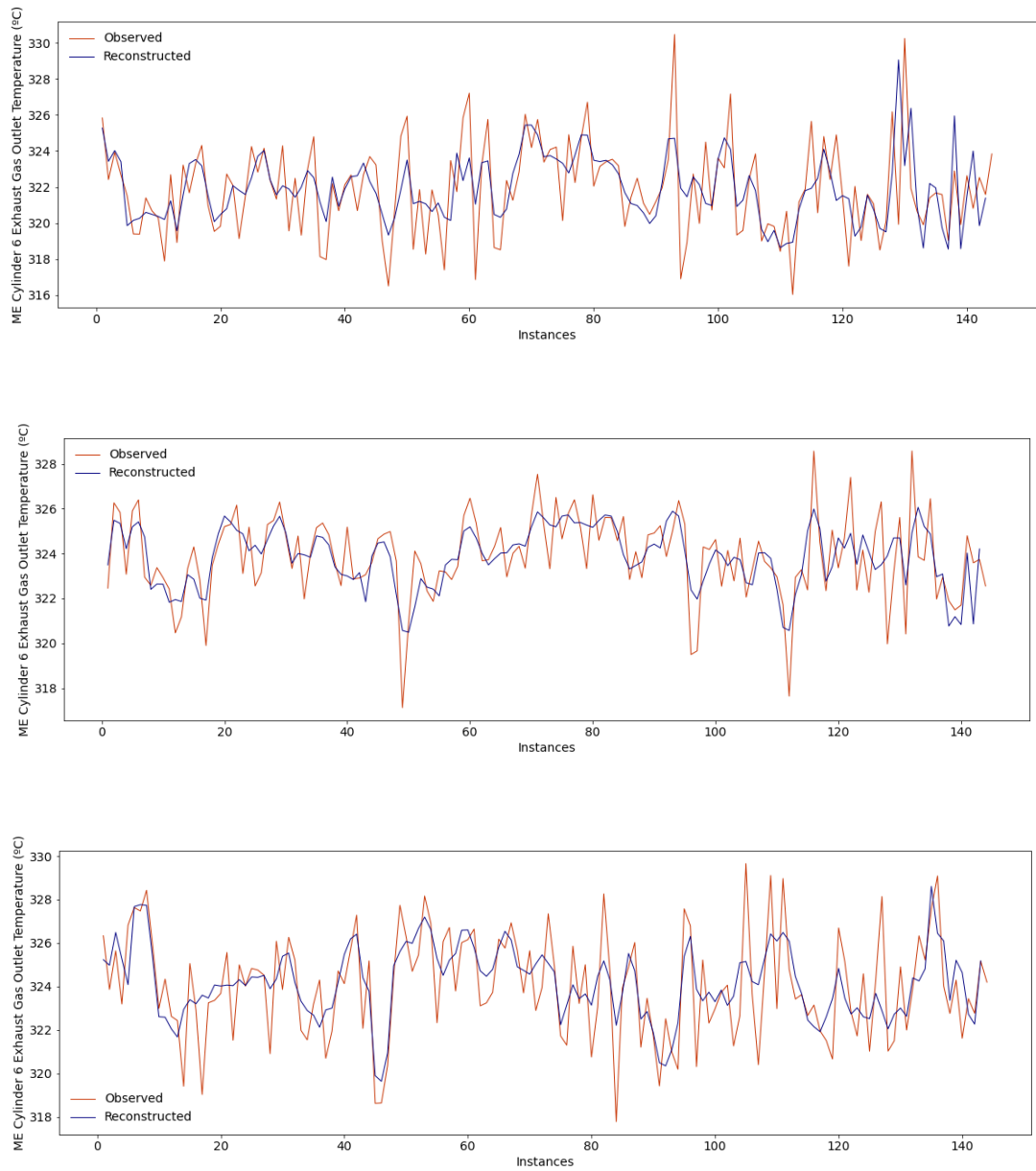


Fig. D.7.10. Example of reconstructed sequences with collective anomalies.

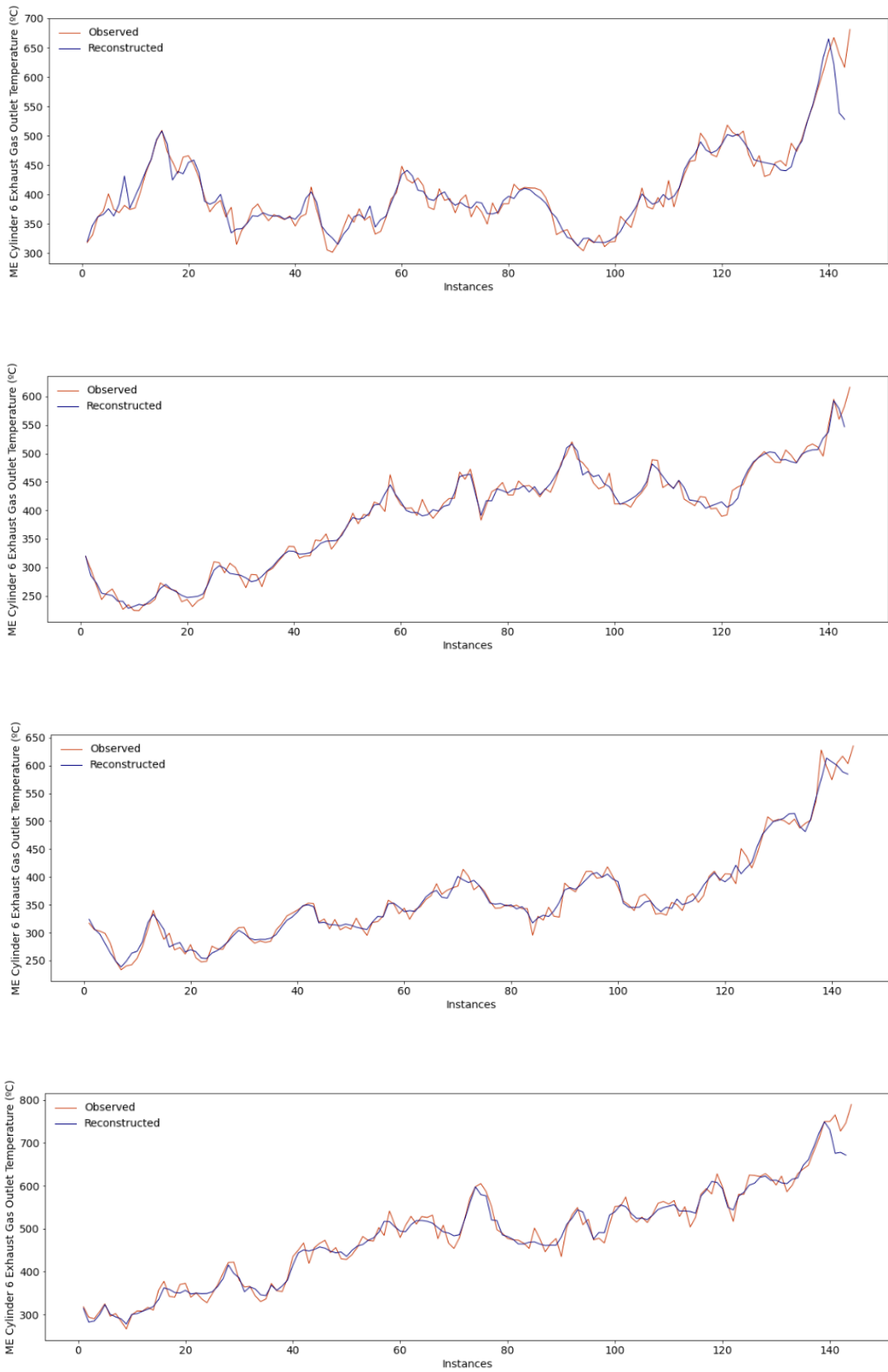


Fig. D.7.11. Example of reconstructed sequences with degradation patterns.

Those sequences detected as anomalous are then classified into two categories: sequences with collective anomalies, and sequences with degradation patterns. Accordingly, as the implemented approach refers to a time series imaging approach (see section 5.3.2. *Fault Identification*), the anomalous sequences detected are transformed into images. Examples of these can be perceived in Figs. E.7.12 – E.7.13.

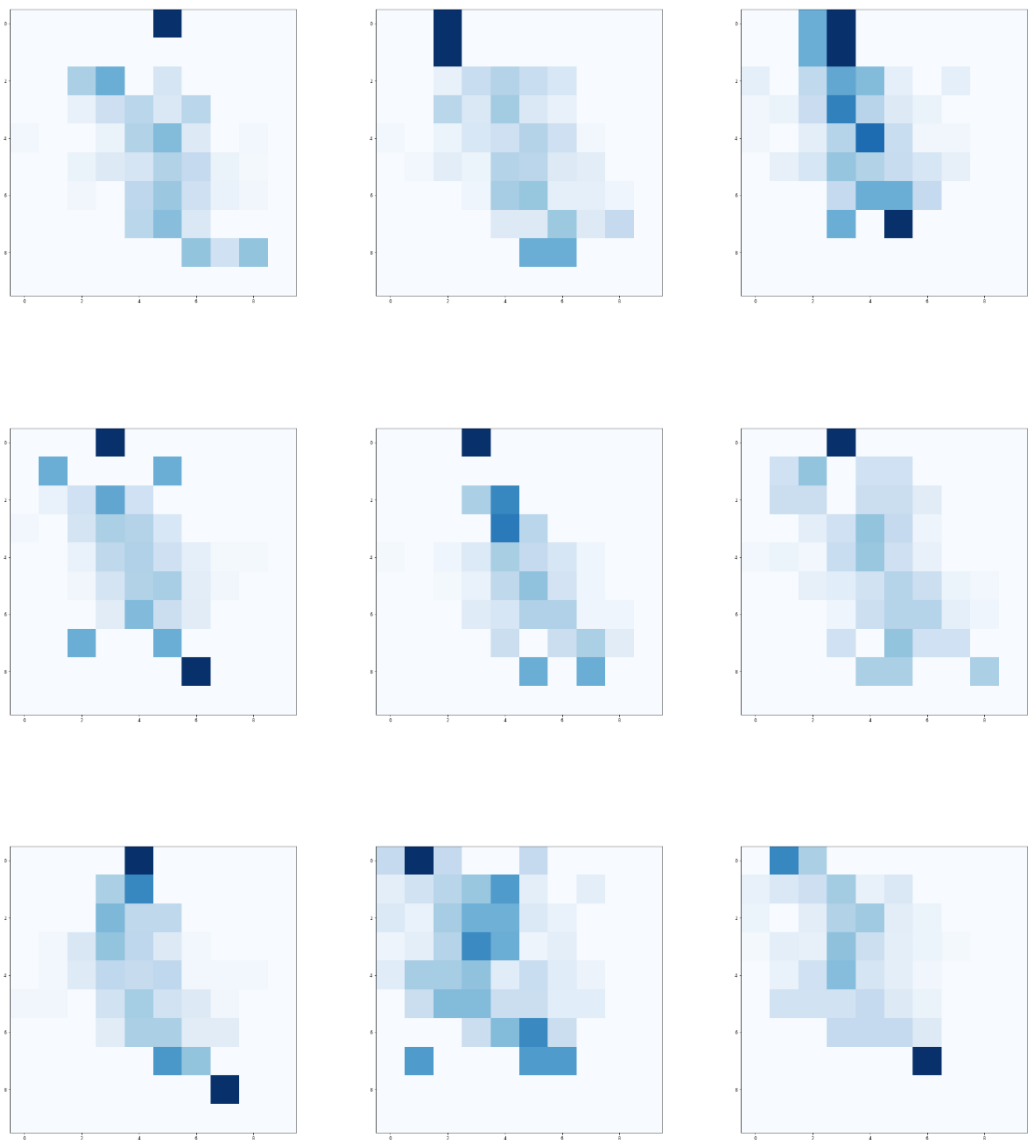


Fig. D.7.12. Images with collective anomalies.

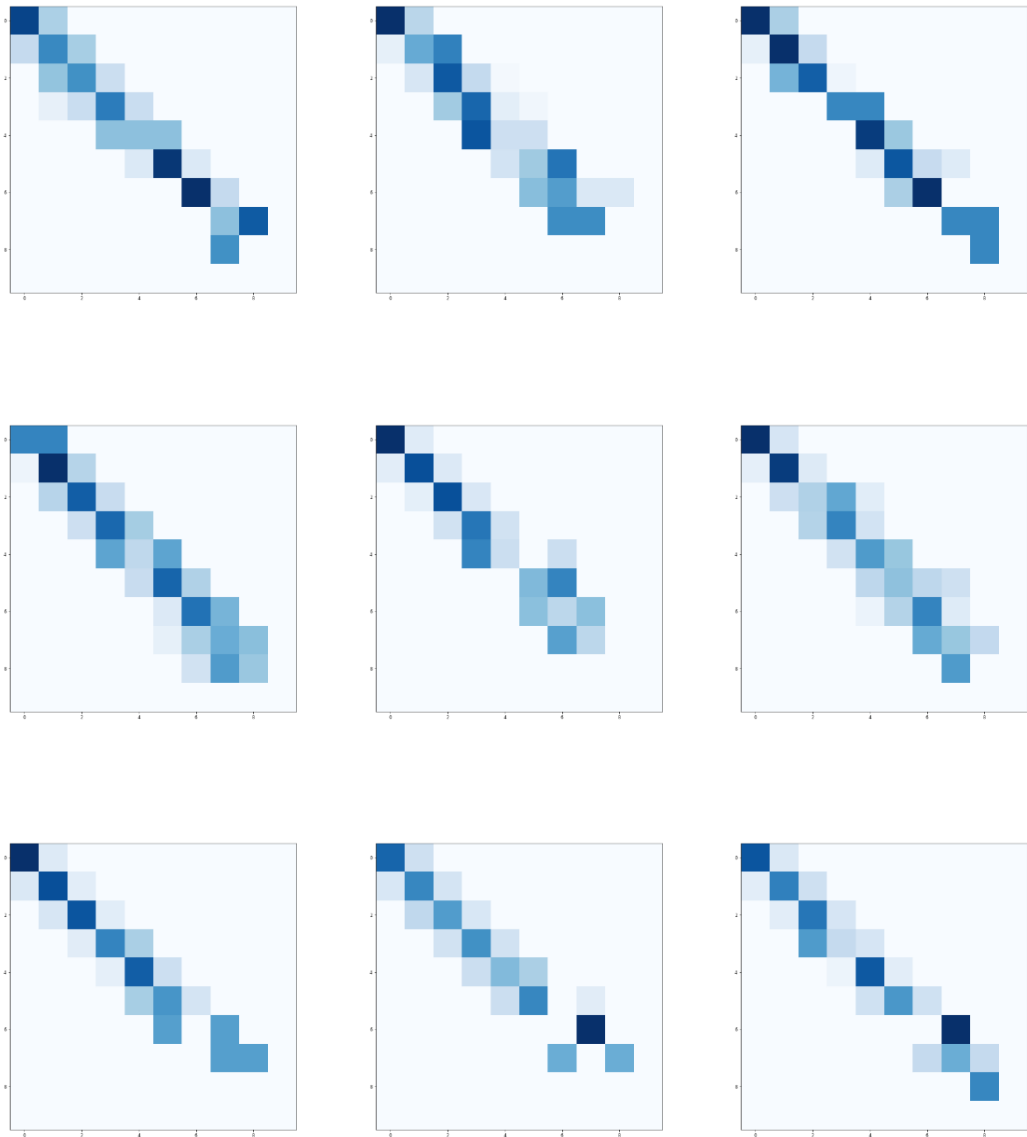
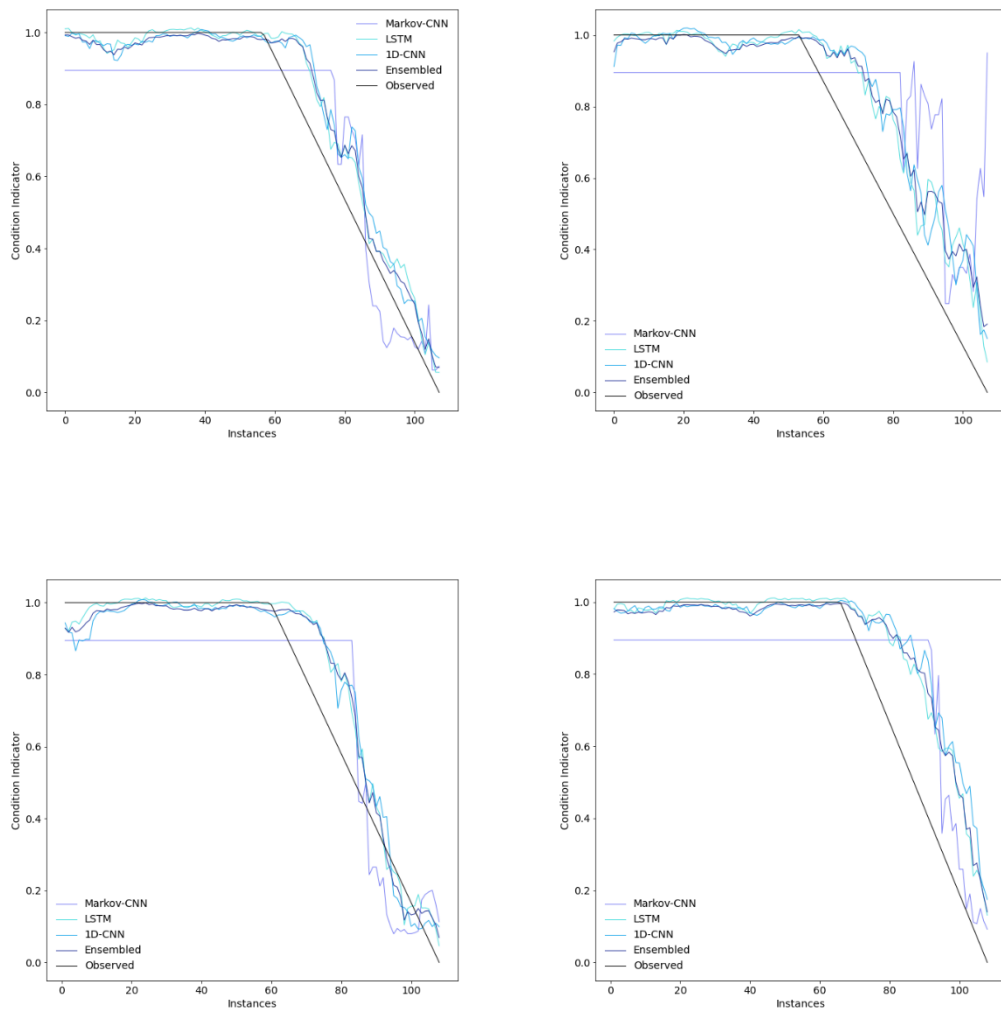


Fig. D.7.13. Images with degradation patterns.

As it can be perceived, the two categories can be easily distinguished. This aspect facilitated the achievement of the maximum accuracy score in this process, thus classifying all the images adequately.

By adequately classifying such images, the sequences with degradation patterns are selected so that the RUL can be predicted. Examples of such a prediction are presented in Fig. D.7.14. The RMSE and Maintenance Score of the first 100 sequences are also presented in Table E.7.2.



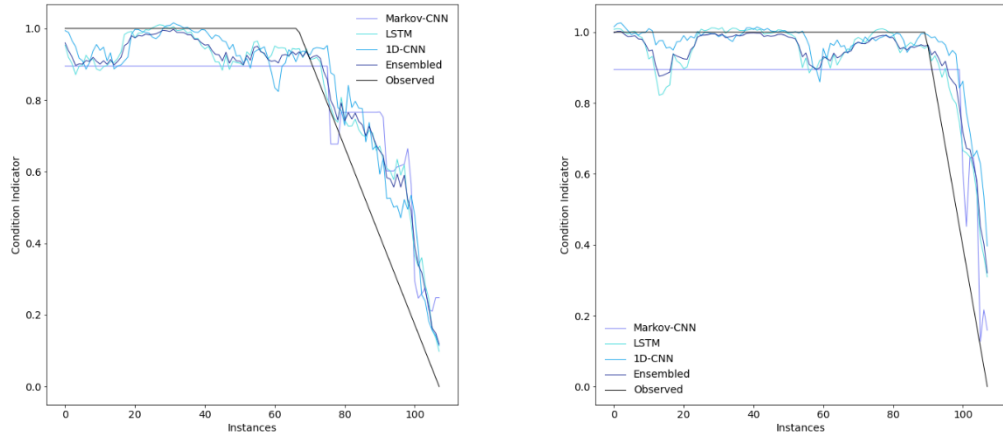


Fig. D.7.14. Examples of condition indicator.

Table D.7.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models.

Sequences	RMSE (Minutes)				Maintenance Score			
	Markov-CNN	LSTM	1D-CNN	Ensembled	Markov-CNN	LSTM	1D-CNN	Ensembled
Sequence 1	33.96	17.21	18.09	16.92	2.10	0.84	0.85	0.92
Sequence 2	31.93	18.00	18.08	18.40	1.69	1.08	0.94	1.07
Sequence 3	19.86	15.73	17.06	15.67	1.16	0.67	0.82	0.74
Sequence 4	25.20	17.89	18.65	17.04	1.44	1.06	1.13	1.03
Sequence 5	25.81	15.63	17.22	16.28	1.36	0.79	0.91	0.87
Sequence 6	45.04	43.08	38.86	39.80	2.54	2.42	2.17	2.28
Sequence 7	50.22	44.15	39.65	43.14	2.47	2.25	1.99	2.18
Sequence 8	54.91	54.52	56.64	54.96	2.70	2.67	2.74	2.68
Sequence 9	22.57	17.86	17.22	17.63	1.30	0.86	0.84	0.89
Sequence 10	36.20	47.72	43.90	44.80	1.72	2.03	1.85	1.92
Sequence 11	20.58	20.32	23.19	20.17	1.21	0.93	0.95	0.92
Sequence 12	30.63	25.91	33.57	28.24	1.87	1.35	1.77	1.52
Sequence 13	58.73	21.15	21.18	22.49	3.40	1.18	0.98	1.33
Sequence 14	37.29	35.37	35.73	35.26	1.87	1.77	1.64	1.72
Sequence 15	60.95	52.50	53.56	52.54	3.23	2.62	2.67	2.62
Sequence 16	33.46	15.58	18.45	16.96	1.66	0.66	0.78	0.77
Sequence 17	36.76	37.91	33.18	33.88	1.90	1.92	1.73	1.77
Sequence 18	29.03	26.11	25.99	25.91	1.48	1.00	1.04	1.01
Sequence 19	29.95	29.93	28.74	29.03	1.51	1.50	1.38	1.45
Sequence 20	24.14	27.32	29.11	26.70	1.47	1.51	1.63	1.52
Sequence 21	46.98	29.26	33.18	30.62	2.96	1.57	1.69	1.74
Sequence 22	44.63	28.84	33.30	30.67	2.57	1.77	1.85	1.85
Sequence 23	28.14	22.34	20.88	21.96	1.38	0.91	0.86	0.90

Sequence 24	17.10	11.69	12.72	11.86	1.09	0.44	0.58	0.52
Sequence 25	29.14	19.22	15.73	18.25	1.45	0.80	0.67	0.78
Sequence 26	50.22	31.76	35.94	34.17	3.06	1.91	1.83	2.00
Sequence 27	61.54	40.00	43.04	41.34	3.58	2.06	2.26	2.14
Sequence 28	22.47	16.11	16.48	15.96	1.26	0.72	0.70	0.73
Sequence 29	47.57	24.03	30.26	26.74	2.67	1.20	1.60	1.44
Sequence 30	22.45	21.25	27.78	22.59	1.27	1.14	1.45	1.24
Sequence 31	23.41	16.56	14.65	15.81	1.32	0.82	0.62	0.79
Sequence 32	41.01	45.64	50.46	44.92	2.30	2.67	2.93	2.69
Sequence 33	61.27	43.55	47.32	45.99	3.50	2.45	2.54	2.56
Sequence 34	34.35	19.11	24.09	21.61	1.61	0.63	0.86	0.76
Sequence 35	22.47	28.90	30.47	27.77	1.15	1.55	1.60	1.49
Sequence 36	20.02	14.08	17.28	15.24	1.25	0.46	0.65	0.58
Sequence 37	53.78	42.08	43.25	42.49	3.47	2.37	2.46	2.45
Sequence 38	59.13	50.97	55.75	52.12	3.24	2.60	2.97	2.67
Sequence 39	21.21	15.48	15.39	15.22	1.21	0.70	0.72	0.71
Sequence 40	40.77	51.81	54.16	50.75	2.13	2.65	2.90	2.64
Sequence 41	19.70	12.77	13.68	12.94	1.21	0.48	0.59	0.54
Sequence 42	56.15	52.86	50.52	52.28	2.69	2.50	2.37	2.45
Sequence 43	36.00	20.47	22.70	21.77	1.74	0.85	0.95	0.94
Sequence 44	39.11	17.97	18.54	19.48	1.80	0.88	0.86	0.95
Sequence 45	28.96	21.74	20.68	21.37	1.93	1.26	1.24	1.29
Sequence 46	24.98	20.44	24.96	21.75	1.38	0.84	0.96	0.91
Sequence 47	32.51	32.92	29.78	31.55	1.59	1.59	1.41	1.52
Sequence 48	23.72	14.27	14.52	13.69	1.24	0.68	0.70	0.66
Sequence 49	42.21	30.18	34.37	31.76	2.32	1.11	1.35	1.29

Sequence 50	65.39	38.40	34.24	37.75	3.74	2.13	1.83	2.16
Sequence 51	25.96	21.98	25.60	22.46	1.43	0.94	1.14	1.04
Sequence 52	24.86	21.88	25.96	22.82	1.33	0.80	0.97	0.87
Sequence 53	29.46	24.89	26.24	24.71	1.39	1.22	1.30	1.23
Sequence 54	19.93	17.93	17.96	17.65	1.13	0.82	0.89	0.85
Sequence 55	31.70	21.51	22.25	22.33	1.64	0.76	0.87	0.84
Sequence 56	34.69	47.16	40.99	42.21	1.75	2.39	2.06	2.14
Sequence 57	20.20	17.11	19.20	16.86	1.22	0.92	0.98	0.94
Sequence 58	25.22	17.10	21.96	18.02	1.40	1.06	1.19	1.08
Sequence 59	56.34	39.27	44.22	41.63	3.17	2.12	2.50	2.26
Sequence 60	43.67	35.54	28.59	33.44	2.21	1.55	1.21	1.48
Sequence 61	26.38	15.73	18.04	16.16	1.39	0.65	0.75	0.70
Sequence 62	25.39	20.43	21.35	20.11	1.36	0.98	0.94	0.97
Sequence 63	25.22	12.84	18.46	12.92	1.34	0.62	0.90	0.66
Sequence 64	21.13	9.67	12.28	10.72	1.31	0.47	0.56	0.55
Sequence 65	30.51	33.93	34.66	33.23	1.41	1.68	1.63	1.61
Sequence 66	28.30	15.51	15.52	14.52	1.28	0.73	0.75	0.74
Sequence 67	18.33	20.92	22.43	20.30	1.01	0.93	1.01	0.92
Sequence 68	32.83	37.92	37.13	36.54	1.62	1.89	1.78	1.81
Sequence 69	17.12	13.20	13.38	12.89	1.00	0.50	0.55	0.52
Sequence 70	16.81	15.40	20.68	16.08	1.12	0.74	1.01	0.82
Sequence 71	29.84	20.59	15.96	18.85	1.96	1.12	0.73	1.07
Sequence 72	26.79	20.88	20.96	20.99	1.44	0.78	0.82	0.82
Sequence 73	30.97	42.32	39.51	39.89	1.64	2.05	1.91	1.94
Sequence 74	31.55	34.26	31.22	31.83	1.48	1.71	1.49	1.58
Sequence 75	17.73	22.00	26.99	20.46	1.16	1.01	1.17	0.98

Sequence 76	37.64	33.53	32.94	33.19	1.80	1.40	1.41	1.41
Sequence 77	77.34	68.75	75.98	70.57	4.12	3.22	3.66	3.25
Sequence 78	18.14	29.56	24.23	25.60	1.05	1.15	0.97	1.04
Sequence 79	23.66	23.36	21.16	22.08	1.27	0.95	0.88	0.90
Sequence 80	65.43	34.05	35.38	35.77	3.79	1.55	1.72	1.81
Sequence 81	43.88	18.89	22.16	20.34	2.45	0.87	1.05	0.99
Sequence 82	24.84	14.92	18.27	15.88	1.38	0.78	0.77	0.82
Sequence 83	26.43	27.78	28.49	26.12	1.53	1.51	1.53	1.42
Sequence 84	33.92	29.98	30.69	30.17	1.72	1.16	1.17	1.18
Sequence 85	35.52	31.29	25.14	29.02	1.66	1.56	1.22	1.45
Sequence 86	30.72	36.53	29.34	32.88	1.60	1.57	1.24	1.45
Sequence 87	33.58	28.78	26.64	28.08	1.69	1.20	1.10	1.22
Sequence 88	38.07	27.69	22.93	26.58	1.80	1.18	0.98	1.17
Sequence 89	25.76	21.28	29.69	22.79	1.71	1.24	1.71	1.39
Sequence 90	34.98	28.79	26.45	28.19	1.81	1.08	1.09	1.13
Sequence 91	21.43	34.46	24.79	29.47	1.06	1.68	1.18	1.43
Sequence 92	23.53	39.59	38.53	36.25	1.02	2.13	1.96	1.93
Sequence 93	64.13	44.14	34.13	41.84	3.99	2.63	2.02	2.53
Sequence 94	80.98	84.23	86.15	83.99	4.25	4.41	4.55	4.39
Sequence 95	51.04	29.68	32.00	29.46	2.71	1.43	1.62	1.58
Sequence 96	17.91	11.33	12.93	11.50	1.09	0.47	0.56	0.52
Sequence 97	36.99	19.71	22.34	21.37	1.84	1.01	1.21	1.14
Sequence 98	85.09	67.81	71.78	68.85	4.83	3.16	3.42	3.18
Sequence 99	26.29	25.78	25.98	24.67	1.25	1.34	1.37	1.29
Sequence 100	39.21	15.93	17.49	16.42	2.11	0.78	0.79	0.88
Median	30.51	24.03	25.96	24.67	1.61	1.16	1.17	1.22

D.8. Turbocharger (TC)

The parameter analysed refers to the exhaust gas outlet temperature. A graphical representation of such a parameter is expressed in Fig. D.8.1. The descriptive statistics is also introduced in Table E.8.1.

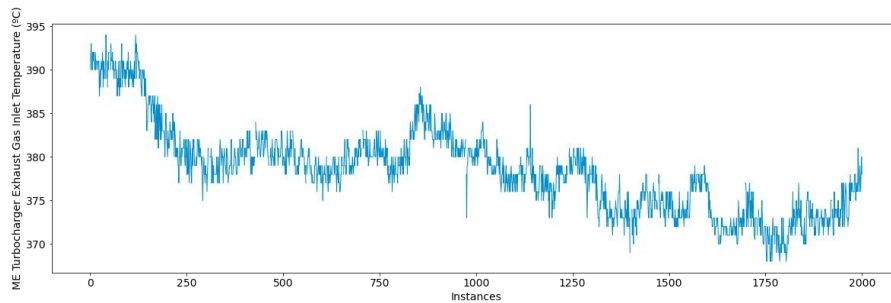


Fig. D.8.1. Graphical representation of the exhaust gas outlet temperature parameter.

Table D.8.1. Descriptive statistics of the monitored parameter.

	Mean	Std.	Min.	25%	50%	75%	Max.
TC Exh. Gas Out. Temp	378.52	4.98	368	375	378	381	394

As part of the data pre-processing phase, the identification of operational states step has been implemented (see section 4.5. *A Novel Framework for the Identification of Steady States* for a comprehensive explanation of such a step). In total, only one operational state has been identified, as perceived in Fig. D.8.2.

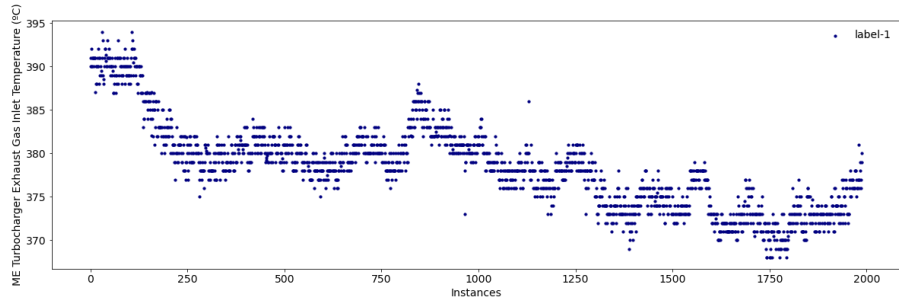
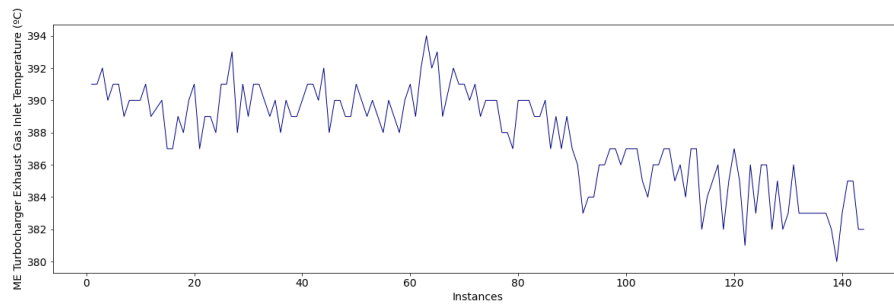
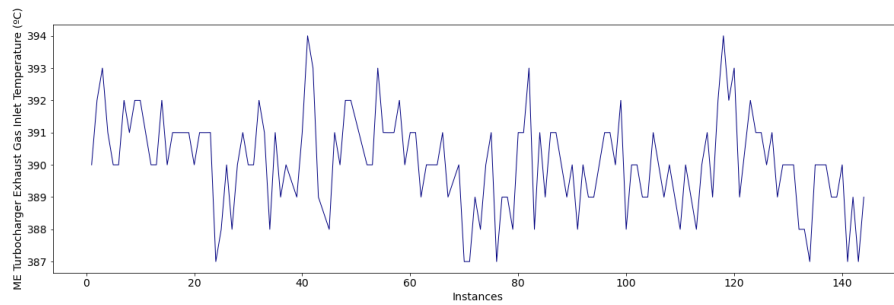


Fig. D.8.2. Identification of the operational states for the monitored parameter.

Subsequently, due to the lack of fault data, both collective anomalies and degradation patterns are simulated. Some examples are presented in Figs. E.8.4 – E.4.5. Examples of normal sequences are also introduced in Fig. D.8.3.



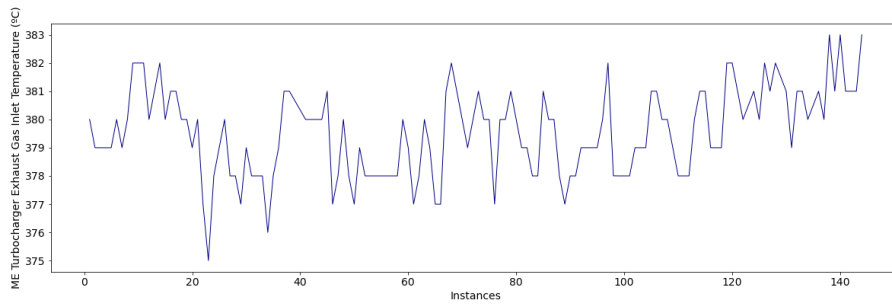
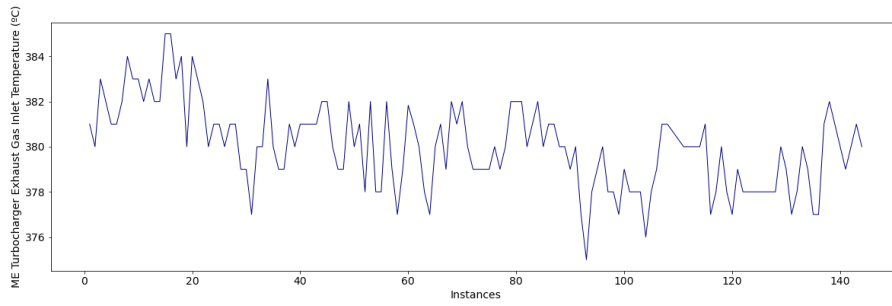
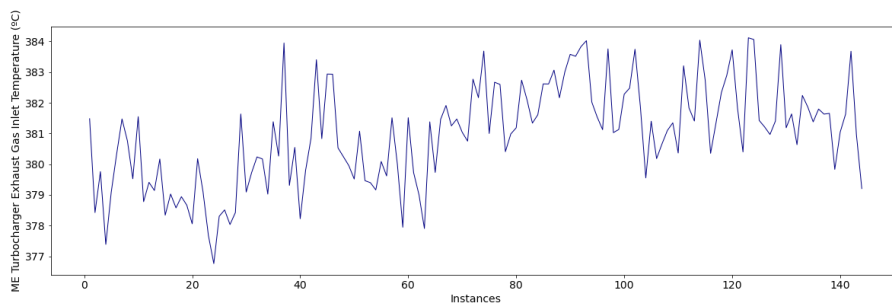
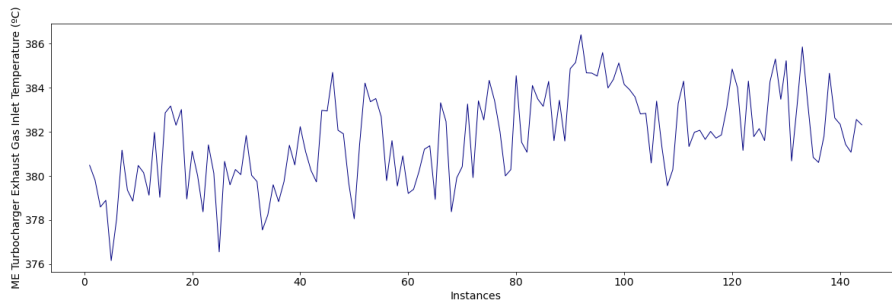


Fig. D.8.3. Example of normal sequences.



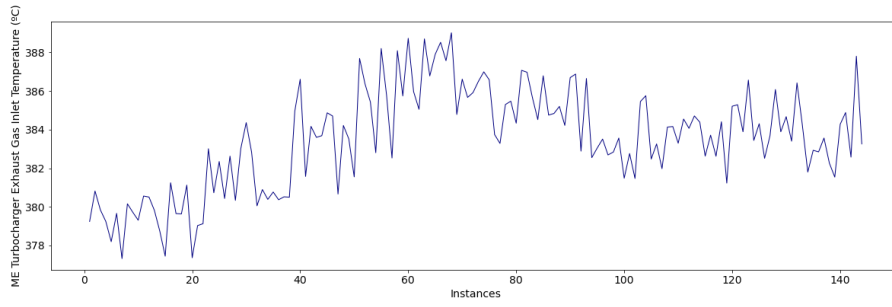
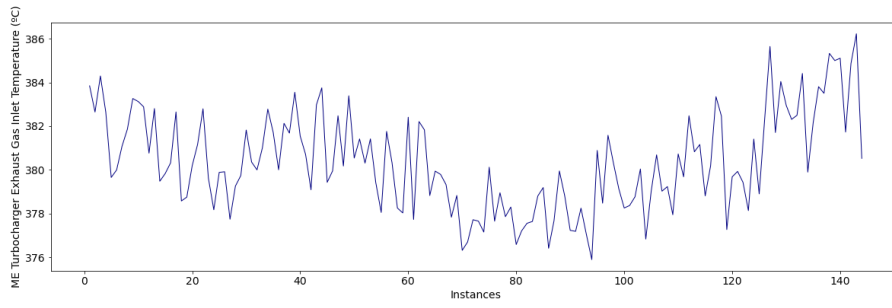
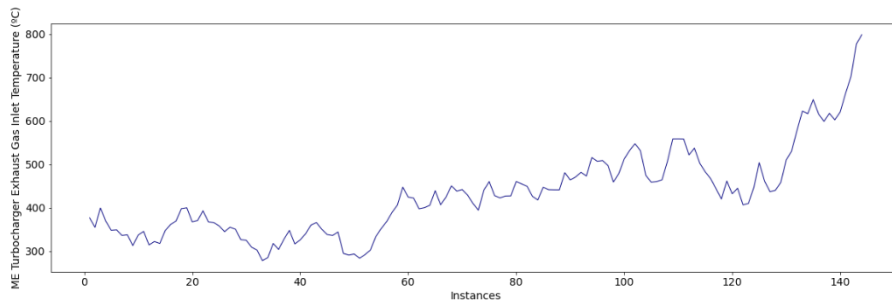
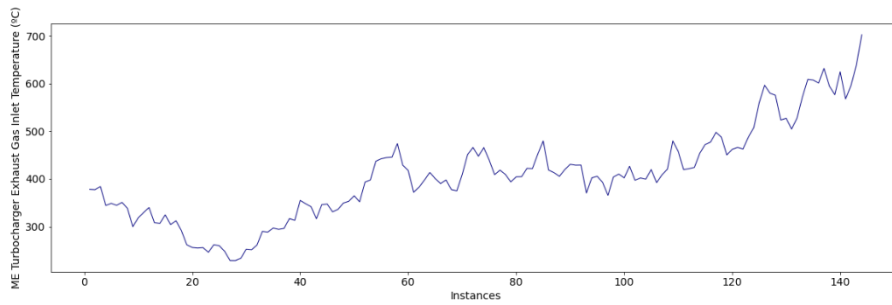


Fig. D.8.4. Example of sequences with collective anomalies.



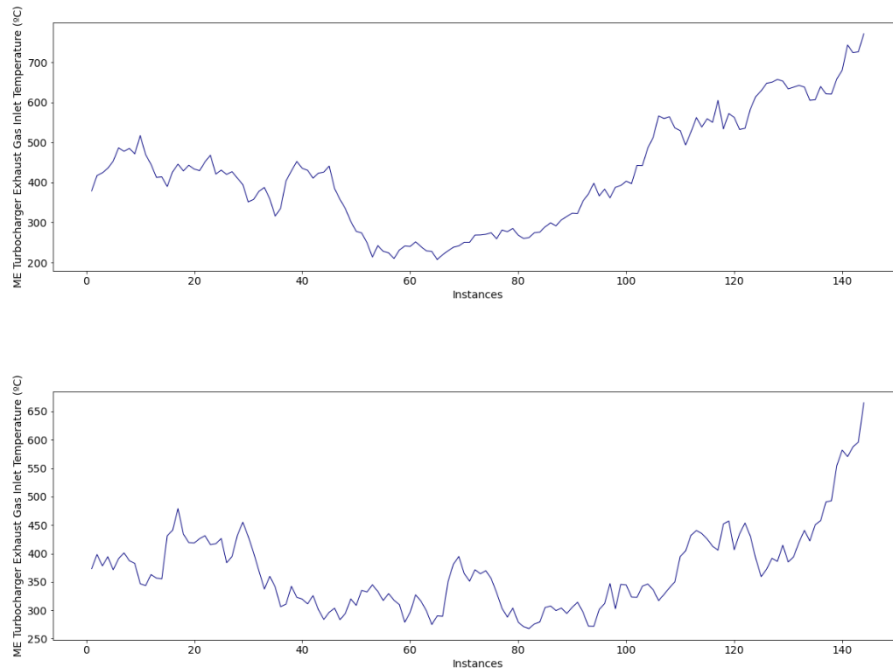


Fig. D.8.5. Example of sequences with degradation patterns.

As part of the MA framework, the subsequent module to be applied is the diagnostic analytics module. Accordingly, the fault detection step is implemented as stated in section 5.3.1. *Fault Detection*. As perceived in the histograms (Figs. E.8.6 – E.8.8), a simple threshold is adequate in this case study to distinguish the normal sequences from the abnormal sequences.

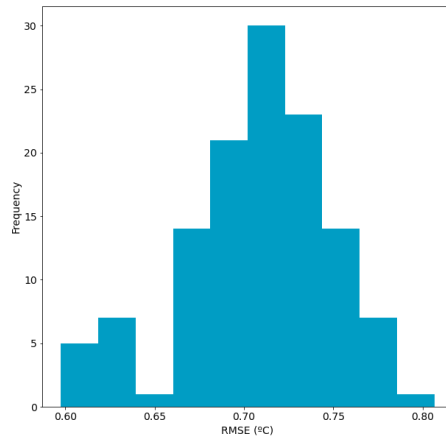


Fig. D.8.6. Histogram of the reconstructed errors of the normal sequences (test set).

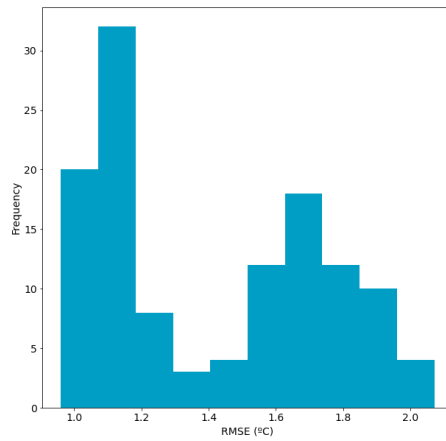


Fig. D.8.7. Histogram of the reconstructed errors of the sequences with collective anomalies.

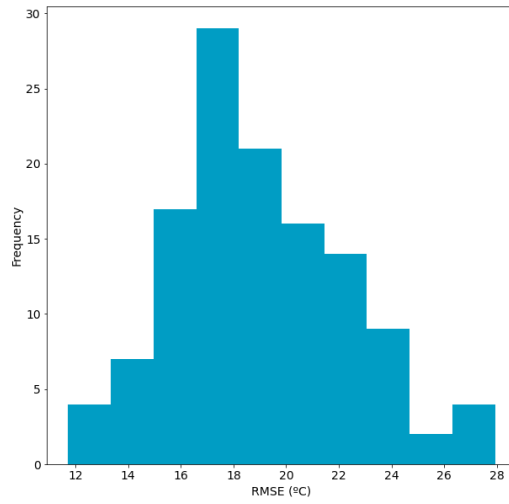
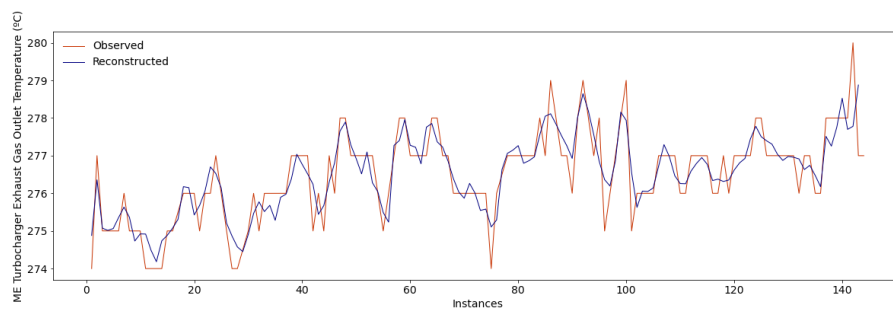
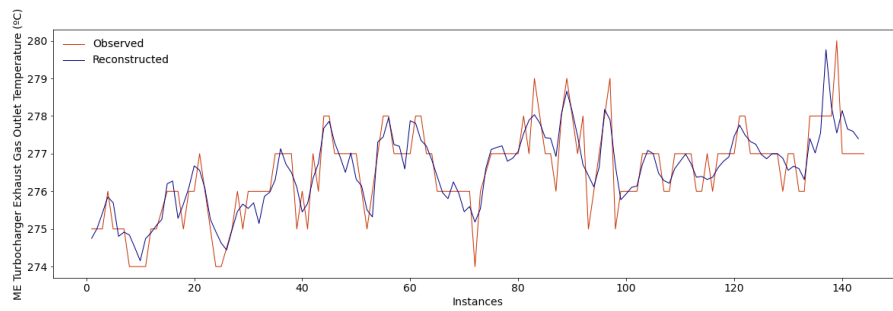


Fig. D.8.8. Histogram of the reconstructed errors of the sequences with degradation patterns.

Examples of reconstructed sequences are also introduced in Figs. E.8.9 – E.8.11.



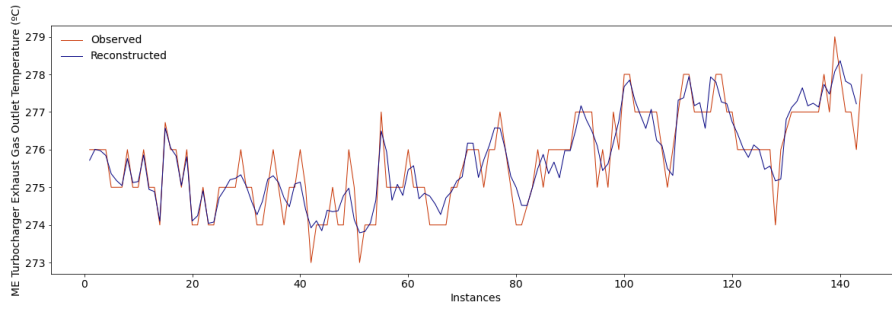
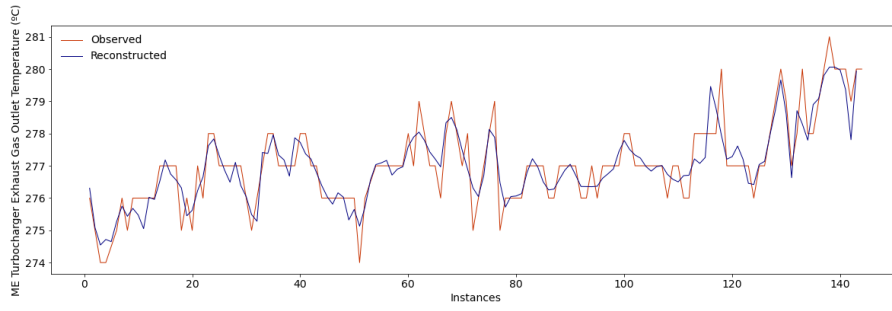
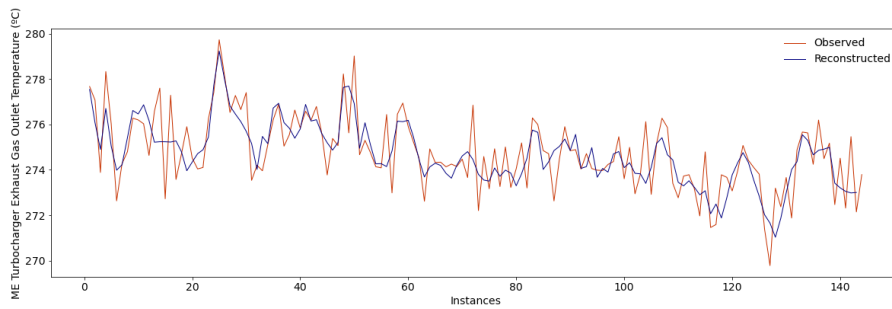
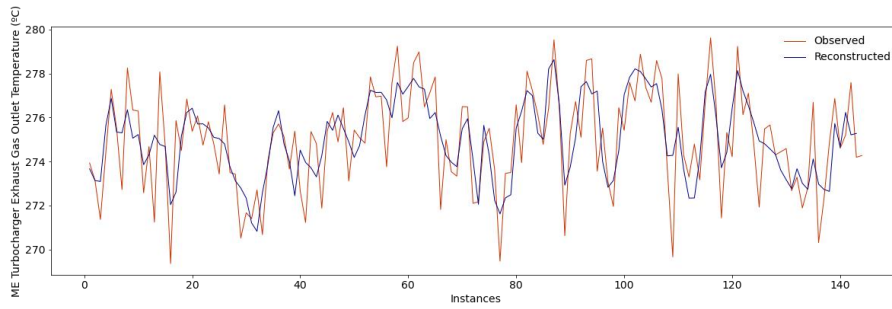


Fig. D.8.9. Example of normal reconstructed sequences (test set).



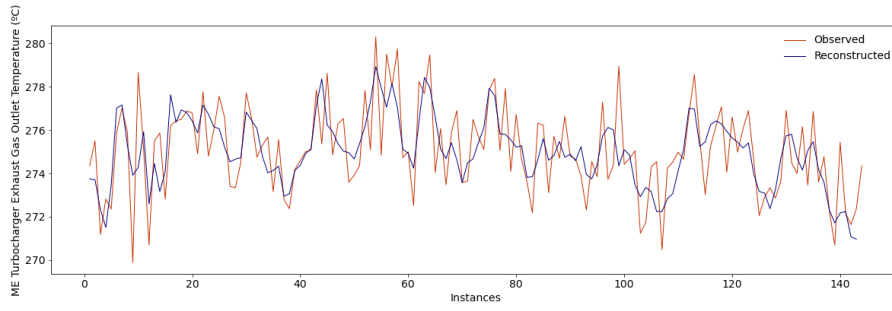
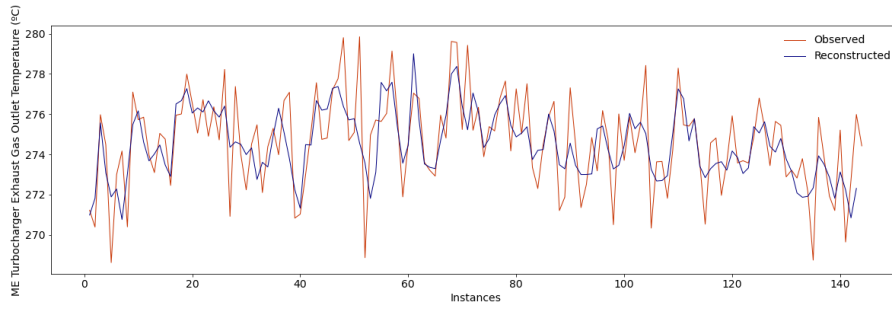
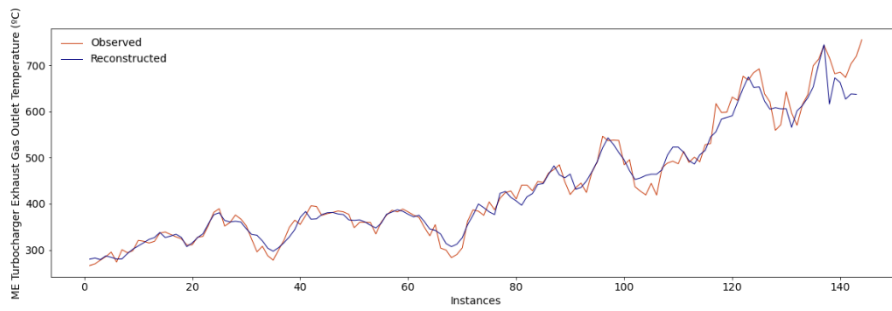
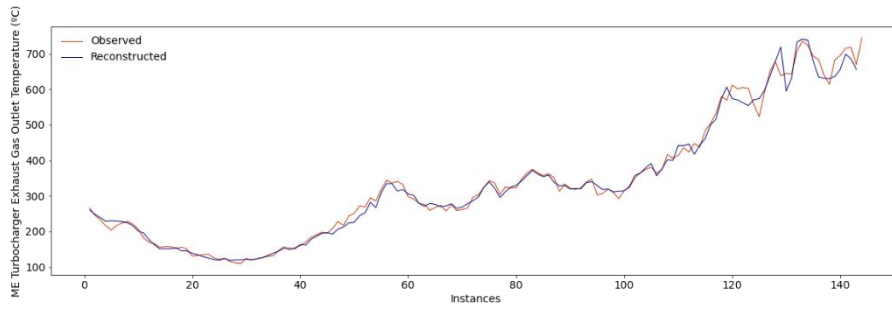


Fig. D.8.10. Example of reconstructed sequences with collective anomalies.



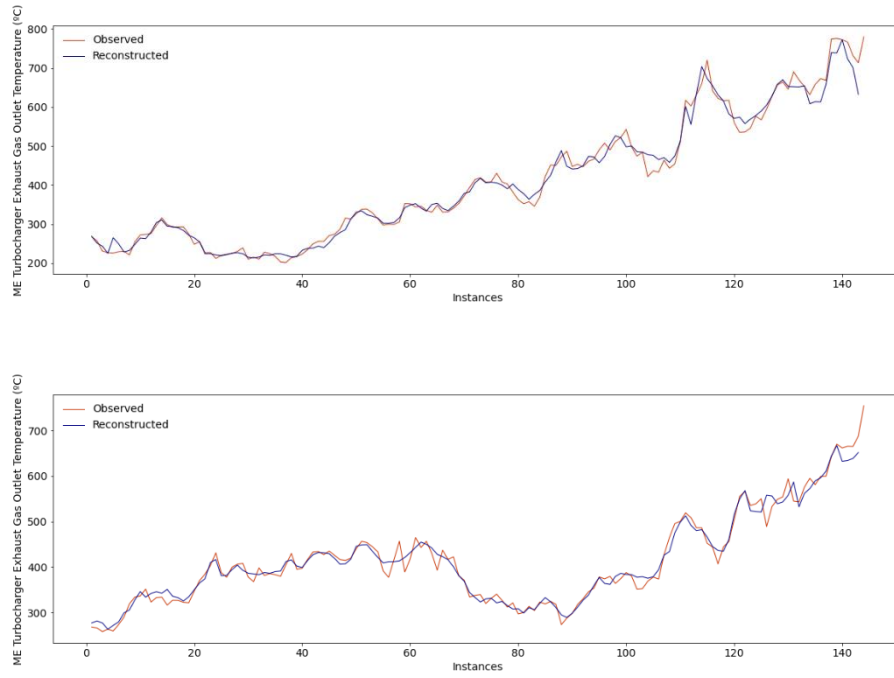
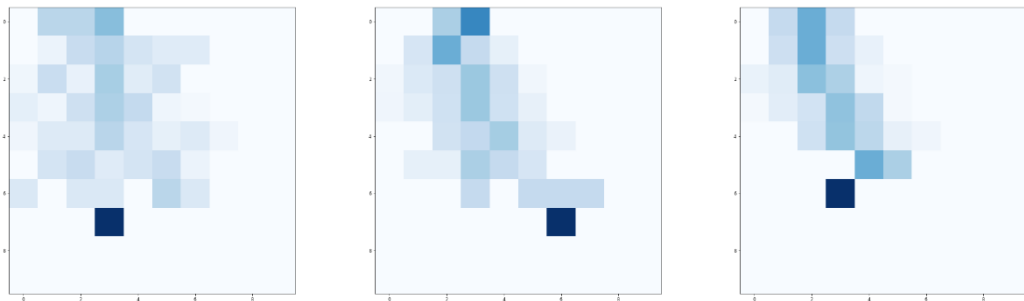


Fig. D.8.11. Example of reconstructed sequences with degradation patterns.

Those sequences detected as anomalous are then classified into two categories: sequences with collective anomalies, and sequences with degradation patterns. Accordingly, as the implemented approach refers to a time series imaging approach (see section 5.3.2. *Fault Identification*), the anomalous sequences detected are transformed into images. Examples of these can be perceived in Figs. E.8.12 – E.8.13.



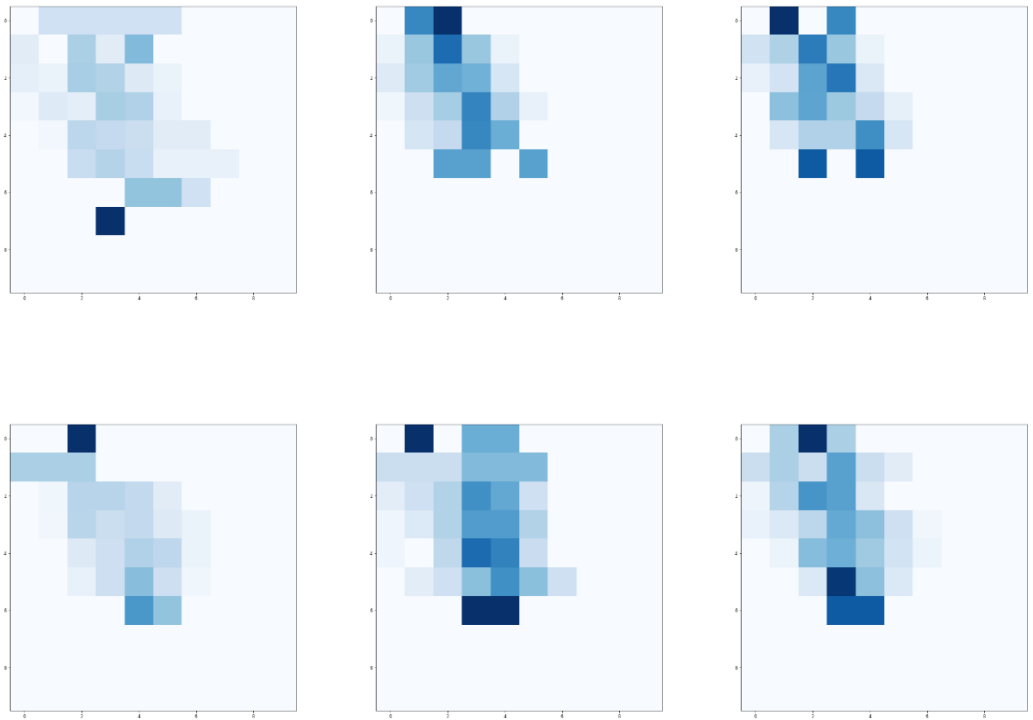
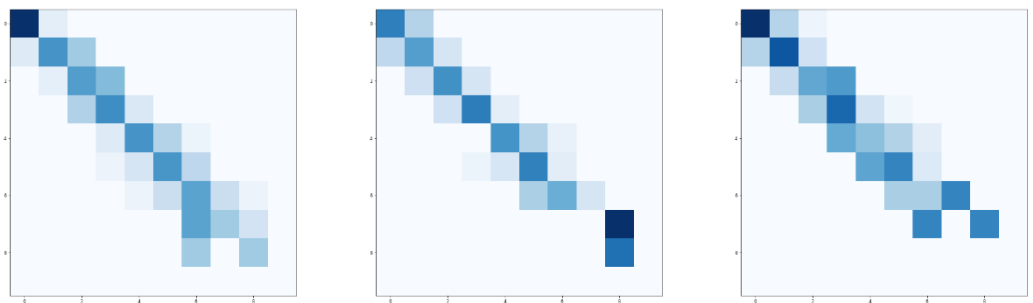


Fig. D.8.12. Images with collective anomalies.



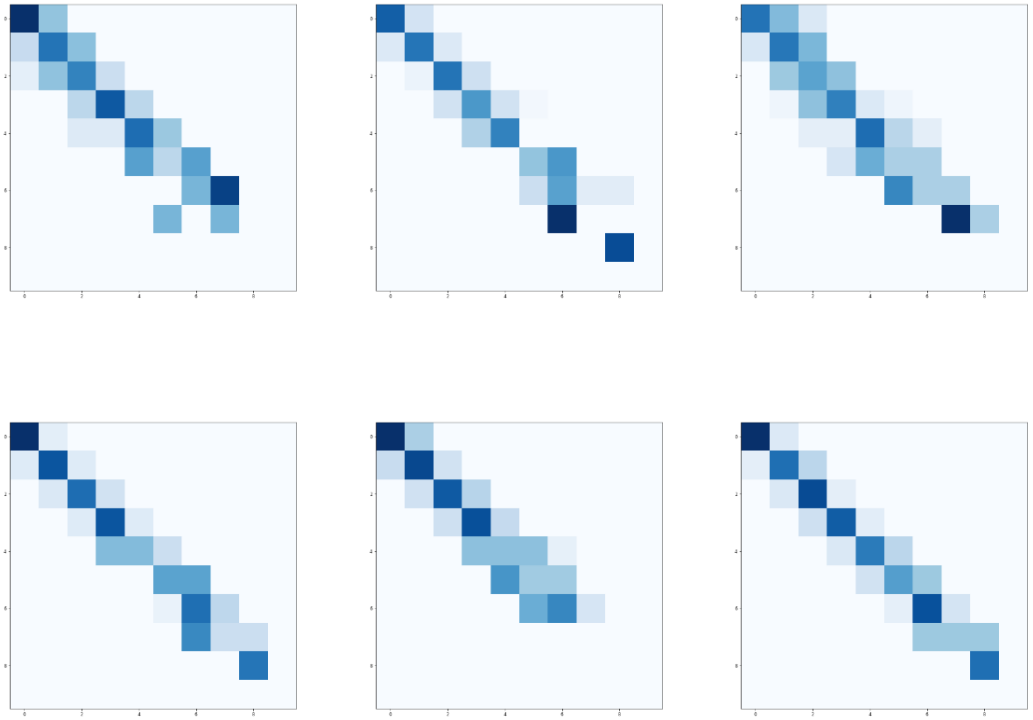


Fig. D.8.13. Images with degradation patterns.

As it can be perceived, the two categories can be easily distinguished. This aspect facilitated the achievement of the maximum accuracy score in this process, thus classifying all the images adequately.

By adequately classifying such images, the sequences with degradation patterns are selected so that the RUL can be predicted. Examples of such a prediction are presented in Fig. D.8.14. The RMSE and Maintenance Score of the first 100 sequences are also presented in Table E.8.2.

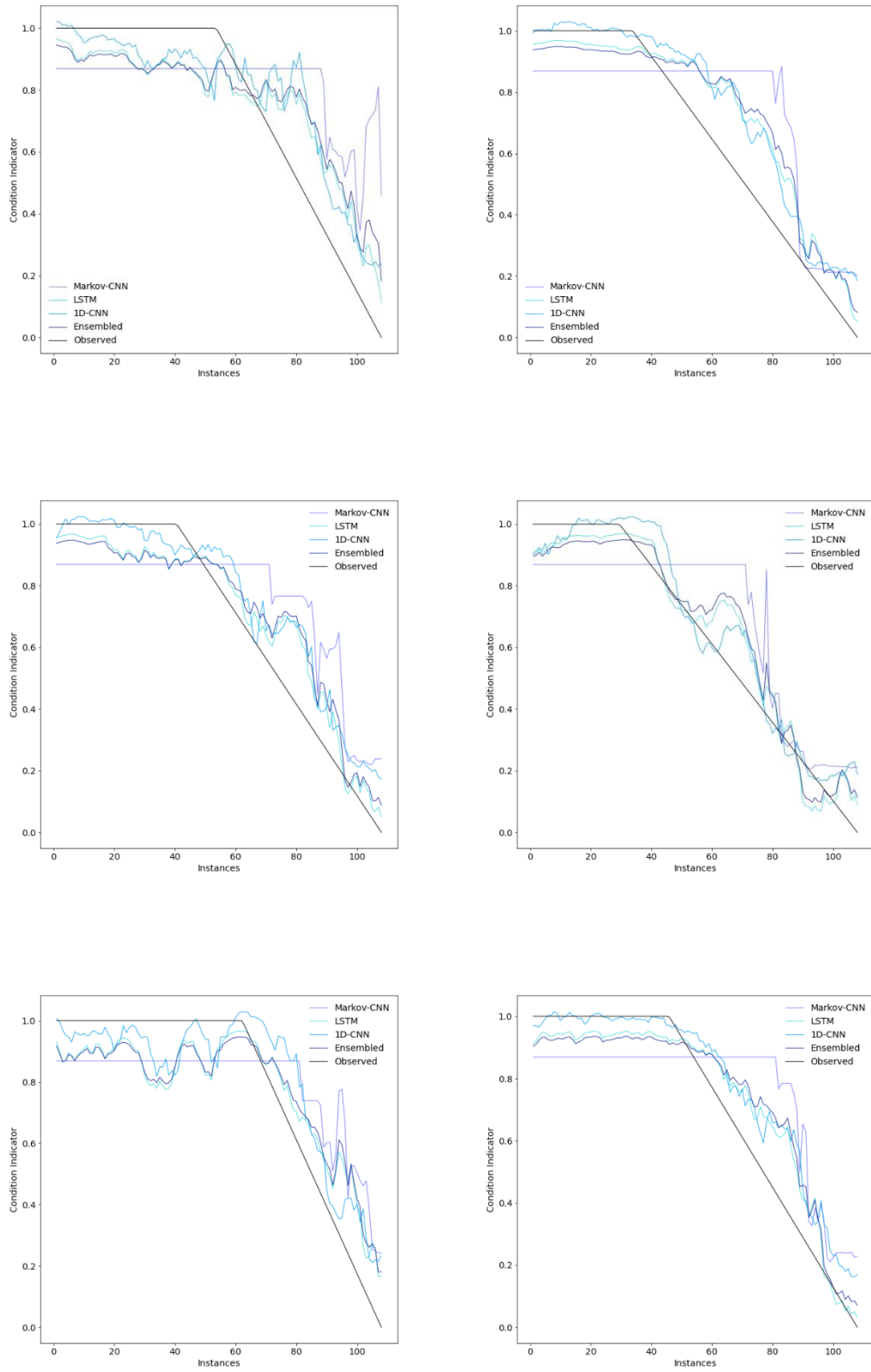


Fig. D.8.14. Examples of condition indicator.

Table D.8.2. RMSE and Maintenance Score between each simulated RUL and their respective predictions for the different analysed models

Sequences	RMSE (Minutes)				Maintenance Score			
	Markov-CNN	LSTM	1D-CNN	Ensembled	Markov-CNN	LSTM	1D-CNN	Ensembled
Sequence 1	49.83	30.47	32.69	33.38	3.02	1.63	1.62	1.88
Sequence 2	38.57	34.79	41.40	35.08	1.93	1.52	1.62	1.59
Sequence 3	33.19	31.27	30.44	30.62	2.23	2.08	1.76	2.11
Sequence 4	22.26	25.22	20.59	23.92	1.57	1.54	0.97	1.54
Sequence 5	30.86	23.64	26.44	23.92	1.80	1.28	1.33	1.36
Sequence 6	31.40	22.18	23.06	23.08	2.00	1.35	0.91	1.48
Sequence 7	31.97	19.78	19.34	21.90	1.62	0.97	0.87	1.10
Sequence 8	25.23	22.11	21.19	22.33	1.55	1.25	1.07	1.30
Sequence 9	39.06	24.75	24.77	27.11	2.02	1.30	1.17	1.44
Sequence 10	29.17	15.06	16.62	17.60	1.61	0.78	0.70	0.93
Sequence 11	25.66	13.29	12.81	15.14	1.44	0.76	0.65	0.89
Sequence 12	29.60	24.55	20.88	24.44	1.78	1.54	1.06	1.57
Sequence 13	24.19	25.21	26.69	23.70	1.46	1.31	1.40	1.29
Sequence 14	98.75	87.34	94.43	89.28	5.68	4.71	5.15	4.89
Sequence 15	66.93	59.78	60.86	59.87	3.20	2.95	2.90	2.93
Sequence 16	43.37	33.27	38.54	35.17	2.20	1.53	1.52	1.66
Sequence 17	29.09	27.52	28.58	26.85	1.46	1.21	1.33	1.24
Sequence 18	66.21	32.66	38.49	38.02	4.02	1.98	2.12	2.37
Sequence 19	54.55	44.79	49.21	46.05	3.52	2.87	2.84	2.99
Sequence 20	37.80	25.54	22.90	27.83	1.98	1.25	0.98	1.39
Sequence 21	25.09	24.49	24.69	23.54	1.74	1.43	1.16	1.48
Sequence 22	26.54	19.46	17.76	20.51	1.57	1.15	0.93	1.23
Sequence 23	48.62	43.09	37.00	42.80	3.22	2.67	2.20	2.77

Sequence 24	28.13	18.42	18.61	20.03	1.59	1.02	0.79	1.13
Sequence 25	34.93	34.47	31.40	33.69	2.32	2.32	1.91	2.32
Sequence 26	25.99	25.98	23.42	25.45	1.59	1.48	1.03	1.50
Sequence 27	49.64	43.12	52.22	43.89	2.42	2.05	2.39	2.11
Sequence 28	29.07	32.33	35.91	30.80	1.94	1.94	1.92	1.94
Sequence 29	47.49	28.92	27.45	32.16	2.33	1.51	1.32	1.67
Sequence 30	41.03	33.14	31.51	33.89	2.77	2.22	1.93	2.32
Sequence 31	29.89	31.73	32.77	30.51	2.02	2.14	1.94	2.11
Sequence 32	31.64	23.10	21.63	24.00	2.05	1.31	1.03	1.46
Sequence 33	69.74	42.71	44.85	46.17	4.23	2.01	2.15	2.39
Sequence 34	48.30	58.47	54.72	56.05	3.04	3.66	3.51	3.48
Sequence 35	58.75	59.34	53.08	58.67	4.14	4.06	3.48	4.07
Sequence 36	69.77	56.05	56.03	57.19	4.32	3.12	3.20	3.36
Sequence 37	29.10	14.96	16.54	17.37	1.62	0.80	0.76	0.96
Sequence 38	43.36	31.95	30.24	33.25	2.94	1.94	1.61	2.13
Sequence 39	28.26	16.80	19.47	17.99	1.51	0.94	0.94	1.03
Sequence 40	32.35	26.92	28.24	26.91	2.07	1.44	1.45	1.53
Sequence 41	20.68	21.65	20.85	20.85	1.39	1.26	0.92	1.28
Sequence 42	30.82	41.76	40.31	38.06	1.96	2.85	2.53	2.56
Sequence 43	38.90	26.94	28.13	27.46	2.39	1.59	1.45	1.74
Sequence 44	86.16	81.15	84.82	81.57	4.75	4.04	4.15	4.13
Sequence 45	23.60	19.81	19.58	20.11	1.50	1.16	0.99	1.22
Sequence 46	30.81	17.93	15.72	20.04	1.73	1.02	0.79	1.16
Sequence 47	38.02	21.64	21.38	24.37	1.97	1.27	1.06	1.40
Sequence 48	32.66	22.72	21.85	23.50	2.15	1.36	1.04	1.50
Sequence 49	32.76	37.74	37.11	35.87	2.14	2.14	1.98	2.14

Sequence 50	44.82	25.65	26.33	27.95	2.91	1.36	1.35	1.62
Sequence 51	26.34	18.73	25.45	19.39	1.41	0.85	1.19	0.95
Sequence 52	51.25	53.73	55.20	51.23	2.61	2.97	2.99	2.76
Sequence 53	20.14	17.78	21.16	16.92	1.18	0.89	1.02	0.92
Sequence 54	36.88	23.21	24.33	25.69	1.92	1.12	1.08	1.28
Sequence 55	23.38	23.04	21.92	22.74	1.48	1.37	1.08	1.39
Sequence 56	19.16	17.92	18.65	17.39	1.19	1.05	1.02	1.04
Sequence 57	47.75	63.74	66.30	60.28	3.32	4.62	4.66	4.35
Sequence 58	25.96	14.52	15.50	16.55	1.56	0.82	0.69	0.96
Sequence 59	18.15	31.32	27.88	28.06	1.31	2.13	1.73	1.94
Sequence 60	24.63	12.29	12.30	13.97	1.28	0.63	0.57	0.71
Sequence 61	58.32	35.59	36.06	39.80	2.86	1.72	1.82	1.93
Sequence 62	27.66	29.22	33.20	28.14	1.45	1.54	1.71	1.51
Sequence 63	45.88	31.51	30.40	34.04	2.36	1.58	1.56	1.72
Sequence 64	36.06	48.21	50.91	44.31	2.17	2.94	2.70	2.78
Sequence 65	59.57	30.23	31.28	33.99	3.71	1.92	1.69	2.27
Sequence 66	40.08	45.67	47.14	44.13	2.65	3.14	2.99	3.03
Sequence 67	40.02	51.07	51.96	47.91	2.61	3.18	3.23	3.04
Sequence 68	24.38	28.20	29.68	26.66	1.55	1.65	1.53	1.62
Sequence 69	28.37	25.13	25.89	25.07	1.83	1.46	1.23	1.53
Sequence 70	35.40	22.46	22.82	24.75	1.90	1.13	1.06	1.27
Sequence 71	31.50	45.78	46.48	42.42	2.19	3.24	3.11	3.02
Sequence 72	41.81	32.18	31.63	30.56	2.75	1.68	1.60	1.79
Sequence 73	58.31	58.65	54.68	58.21	3.56	3.70	3.16	3.65
Sequence 74	30.26	19.24	16.55	21.17	1.73	1.10	0.76	1.23
Sequence 75	30.62	25.23	24.57	25.88	1.81	1.42	1.24	1.50

Sequence 76	26.17	39.40	41.71	36.20	1.29	1.85	1.86	1.69
Sequence 77	64.57	46.69	52.18	49.25	3.86	2.66	3.07	2.82
Sequence 78	40.89	25.30	20.78	28.00	2.14	1.29	1.08	1.46
Sequence 79	25.45	19.57	23.77	17.97	1.28	1.00	1.05	0.85
Sequence 80	36.51	25.22	25.94	27.33	1.92	1.25	1.05	1.38
Sequence 81	36.08	31.56	31.27	31.55	2.32	1.85	1.49	1.93
Sequence 82	23.01	16.88	20.77	15.71	1.33	1.02	1.18	0.93
Sequence 83	23.43	21.33	29.76	21.24	1.39	1.10	1.33	1.15
Sequence 84	68.73	46.63	53.48	48.74	4.40	2.88	3.25	3.16
Sequence 85	34.59	30.50	32.42	29.11	2.17	1.61	1.73	1.65
Sequence 86	48.09	44.86	47.58	44.59	3.06	3.13	3.13	3.11
Sequence 87	36.72	17.18	16.97	20.63	2.12	0.91	0.81	1.15
Sequence 88	35.16	38.26	42.48	36.54	2.37	2.35	2.25	2.35
Sequence 89	43.24	42.14	44.71	41.41	2.36	2.30	2.44	2.29
Sequence 90	38.96	38.34	45.01	37.76	1.80	1.93	2.32	1.87
Sequence 91	58.69	61.68	69.51	60.90	2.78	3.03	3.55	2.97
Sequence 92	41.90	27.39	25.16	30.07	2.13	1.37	1.29	1.51
Sequence 93	24.57	24.79	25.61	24.43	1.53	1.41	1.04	1.43
Sequence 94	30.41	20.85	23.34	21.99	1.47	1.12	1.16	1.15
Sequence 95	22.24	14.77	19.76	15.35	1.13	0.70	1.06	0.74
Sequence 96	44.56	35.02	40.01	36.52	2.22	1.54	1.60	1.67
Sequence 97	21.70	28.60	32.77	26.23	1.46	1.67	1.73	1.61
Sequence 98	26.69	32.96	32.63	31.11	1.85	2.16	1.95	2.09
Sequence 99	51.49	53.83	52.48	52.47	2.94	3.03	2.92	2.91
Sequence 100	39.81	23.39	20.62	26.18	2.08	1.17	0.96	1.35
Median	36.08	29.22	30.40	30.02	2.05	1.53	1.48	1.59

