DEVELOPING A NOVEL HYPERPARAMETER OPTIMISATION METHOD USING LEARNING CURVE PREDICTION TO ENHANCE DECISION-MAKING

By Craig Smith

A Doctor of Philosophy Thesis

Submitted to the Department of Design, Manufacturing and Engineering

Management, University of Strathclyde, 75 Montrose St, Glasgow G1 1XJ.

06/2024

Declaration of Authenticity and Author's Rights

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.

Signed:

Date: 14/06/2024

List of Publications

Journal Paper

Advancements in Artificial Intelligence-Based Decision Support Systems for Improving Construction Project Sustainability: A Systematic Literature Review

Informatics **2022**, *9*(2), 43; **https://doi.org/10.3390/informatics9020043**

Conference Paper and Presentation

Decision Support Systems to Improve Project Sustainability in the Construction Sector

EWG-DSS 7th International Conference on Decision Support System Technology - Loughborough, United Kingdom;

https://icdsst2021.files.wordpress.com/2021/05/icdsst-2021-proceedings-1.pdf
Duration: 26 May 2021 → 28 May 2021

Papers Under Review

Title

Creating Learning Curves for Hyperparameter Optimisation: a case using a Convolutional Gated Recurrent Neural Network

Journal: *Applied Soft Computing*

Working papers

Titled

Objective and Optimised Feature Importance Analysis for Improving Construction Project Performance

Abstract

In response to the slow adoption of artificial intelligence (AI) trends in the construction sector, this study is one of the pioneers to tackle the challenges of subjective decision-making practice in construction project management using machine learning (ML) techniques. A neural network-based Decision Support System (DSS) is developed to model significant correlations among various decision factors as well as identifying critical success factors (CSFs) such that decision outcomes can be enhanced with greater accuracy and reduced subjectivity.

Existing methods for HPO using learning curve prediction are limited in their ability to predict unseen learning curves on the same dataset. The current gap in predicting full learning curves without running all configurations limits the efficiency of these approaches and constrains their application. A key contribution of this study is the development of a novel hyperparameter optimisation (HPO) algorithm, namely SEquential Learning Curve Training (SELECT), grounded in learning curve prediction which can help to improve both modelling efficiency and effectiveness. Leveraging a Convolutional Gated Recurrent Neural Network (CGRNN), the SELECT method predicts learning curves for unseen hyperparameter configurations without the need to train them. Comparative validation of SELECT against existing HPO methods such as Tree Parzen's Estimator, Bayesian Optimisation with Gaussian Process, Hyperband and

Random Search were conducted, with prediction accuracies ranging between 7%-68% better than the benchmarks in the experiments. Further to this, the computational expense for the SELECT method is less than that of the benchmarks, with the closest benchmark requiring 25% more time to find optimum hyperparameters, averaging over all datasets. The consistency of allocated computational resources is also another benefit with the standard deviation between experiments being 81s for the SELECT method, while the closest benchmark had a standard deviation of 427s averaged over 5 datasets and 5-fold splits of each. This underscores its superiority in prediction accuracy and computational efficiency. The SELECT algorithm exhibits the capability to find high performing hyperparameter configurations across different well-known datasets, including synthetic and real-world scenarios, and demonstrates a high capability for identifying CSFs through feature importance analysis.

The validation of the DSS, involving feedback from senior industry experts, reflects positive performance evaluations, with an average score of 3.93 out of 5 on a Likert scale over all questions with a standard deviation of 0.84. These experts, intrigued by the system's potential, express strong interest in collaborative efforts for future development. This research, adeptly navigating industry challenges, provides not only objective decision support in construction project management but also introduces a novel HPO approach that transcends the confines of the construction sector, with applicability in the greater field of AI.

Acknowledgements

I would like to express my sincere gratitude to the sponsors of this research, Galliford Try and the National Manufacturing Institute Scotland (NMIS). Their invaluable support, resources, and commitment to fostering innovation have significantly contributed to the success of my research.

I am particularly indebted to my academic mentor, Andy Wong, whose guidance, expertise, and encouragement have been crucial throughout this journey. His insightful advice and unwavering support have greatly enriched my academic and professional growth.

I would also like to extend my thanks to Angus Liu, my industrial supervisor, whose support and practical insights have been immensely helpful. His experience and advice have been invaluable in navigating the complexities of this research.

Dedication

To my family, Serina, Ren, and Louis. We have gone through this journey together. You have been my inspiration to succeed and my crutch in times of difficulty. I am grateful and lucky to have you all in my life.

Table of Contents

List of Abbreviations

Abbreviation	Description
1DCNN	One dimensional convolutional neural network
AI	Artificial intelligence
ANN	Artificial neural network
BI	Bayesian inference
block	A collection of information related to learning curve data over a set interval
$block_{size}$	The number of epochs in each block of learning curve information
C	Training cycle length in blocks
CBR	Case based reasoning
CGRNN	Convolutional gated recurrent neural network
CNN	Convolutional neural network
CP	Commercial Performance
CSF	Critical success factors
DPP	Dynamic performance prediction
D_{scale}	Percentage drop from starting loss to loss threshold when setting the base learning rate
D_{set}	Percentage drop from starting loss to loss threshold when setting the increase rate for the learning rate
DSS	Decision support system
DT	Decision tree
EEP	Early-stage project prediction
EFHNN	Evolutionary fuzzy hybrid neural network
F	Adjustment factor for setting the threshold loss while tuning the learning rate
FDA	Forecast Duration Accuracy at PO
FL	Fuzzy logic
FS	Fibonacci sequence
GA	Genetic algorithm
GD	Gradient descent
grad	The final gradient of the learning curve in a training cycle
$grad_{limit}$	The range of acceptable gradient of convergence while setting the loss threshold
GRU	Gated recurrent unit (recurrent neural network)
HPO	Hyperparameter optimisation
I	Feature Importance
$I_{increase}$	Increase rate of the learning rate
$I_{relative}$	Relative feature importance of a feature

$L_{current}$	The current recorded loss at each stage of training
LIME	Local Interpretable Model-agnostic Explanations
lr_{base}	Base learning rate for negligible difference, set for each configuration
lr_{scale}	Initial learning rate while setting the base learning rate
lr_{set}	The learning rate achieved at the loss threshold
L_{scale}	Loss threshold when setting the base learning rate
L_{set}	Loss threshold when setting the increase rate for the learning rate
L_{start}	Initial loss at the start of training
LSTM	Long short-term memory (recurrent neural network)
$L_{threshold}$	Loss threshold for setting the learning rate of each configuration
MDP	Markov decision process
ML	Machine learning
MLP	Multi-layer perceptron
MLR	Multivariate linear regression
$n_{increase}$	The number of epochs to the loss threshold when setting the increase rate for the learning rate
NLP	Natural language processing
n_{run}	Set epoch limit after the learning rate has been set
OSFDA	On-site Forecast Duration Accuracy
P	The prediction length in blocks
PFI	Permutation feature importance
RF	Random forest
RL	Reinforcement learning
RNN	Recurrent neural network
Sc	Scaling factor for increasing the learning rate exponentially while setting the base learning rate
S	Starting step length in blocks
SHAP	SHapley Additive exPlanations
SRA	Safety risk assessments
SVM	Support vector machine
$train_{seq}$	The sequential training set for training the CGRNN
$trial_{quantity}$	Quantity of trialled configurations from the ranked learning curve predictions
$tune_{quantity}$	The number of iterations while setting the loss threshold
W	The length of the window in blocks
X	Input range for sequential prediction
Y	Output range for sequential prediction

List of Figures

List of Tables

1 Introduction

1.1 Construction Sustainability and Intelligent Decision Support Systems

Throughout history, construction has been one of the key industries which has contributed to both the function and evolution of society, broadly covering the development of housing and infrastructure as well as commercial and industrial sectors (Carty Gerard, 1995). It is a field of engineering which is crucially important throughout all areas of the planet but one which comes with its own unique challenges. Each construction project covers new ground, with new work locations, variations in labour costs and expertise, material and equipment requirements, logistics, safety, and regulatory requirements. This leads to a significant amount of uncertainty which can result in project overspending, delays for project timelines, disputes with customers, contractors, and employees if decision-makers make the incorrect choices (Ortiz-Gonzalez et al., 2022).

The difficulties in decision making do not just relate to the project nature in the construction industry, the task of making the correct decision can be further impacted by the traditional methods of construction project management. The traditional approach relies primarily on the knowledge of experts to make decisions with tacit knowledge and limited data availability(You and Wu, 2019). The methods of collecting and utilising data have been through manual means, with decentralised storage which comes with its own disadvantages. This can result in data acquisition and use being slow, with flaws and missing information, resulting in the potential for the wrong conclusions being made with substantial consequences (Andújar-Montoya et al., 2015).

In recent years, even the critical success factors (CSF) of projects have been evolving to consider not only the economic success of construction projects but also the environment and social impact as well (Kiani Mavi and Standing, 2018). The change in CSF has become evident through the evolution of societal needs towards sustainable practices at the beginning of this century. The focus is no longer prioritising just project scheduling, cost, and quality, but also other tangible and intangible factors as well (Salma Ahmed, 2021). Environmental success in construction requires further considerations regarding the material usage, energy efficiency, waste management and recycling, adhering to environmental standards, while the social factors relate to how a company structure and each project impacts both the employee's safety and satisfaction and the same for the surrounding community. All these factors combine to create enormous challenges in relation to decision making in the construction industry with so many components to consider and there being significant financial, environmental, and social impacts of making the wrong choices.

Further to this, it can be difficult for decision makers to fully understand and control the projects when there are so many decision variables. Even with years of experience and collaboration, decision makers may struggle to uncover useful observations from the data having the 5V's - velocity, volume, value, variety, and veracity. Intelligent tools that can identify and visualise CSFs would help to overcome the inherent limitation of human decision makers (Nikmehr et al., 2021).

In the realm of construction project management, the need for effective decision making is more critical than ever, given the intrinsic complexities involved and the impact on construction sustainability. To address the limitations of traditional methods, the evolution of integrated decision support systems (DSSs) has emerged as a practical solution, aiming to streamline decision-making processes within the industry (Galjanić et al., 2022). By leveraging technology and data driven insights, these systems facilitate informed decision-making for projects of varying scales and complexities. They are also tailored to manage the intricacies of construction projects and provide real-time analyses and predictive models, enabling project managers and key stakeholders to make well informed decisions, allocate resources efficiently, and manage risks effectively. With user-friendly interfaces and customisable features, these systems promote adaptability and efficiency in construction project management, enhancing overall project sustainability.

Complementing the advancements of DSSs, the evolution and utilisation of AI and ML has emerged as a key driver in improving decision making within the construction industry. By harnessing the power of AI driven analytics and ML algorithms, project managers gain access to predictive insights and trend analyses, empowering them to make strategic decisions that mitigate risks and optimise project performance (Pan and Zhang, 2021). Through the integration of AI and ML into DSSs, project managers can navigate the complexities of the field with better precision, enabling them to adapt to evolving project CSF, detect and prepare for potential challenges, and ensure optimal resource allocation. As AI continues to advance, its role in improving decision making processes at all stages of the construction industry is set to redefine how the sector operates, enabling more efficient and adaptable approaches to project management with increased sustainability in the future.

1.2 The Research Problem

The inspiration for this research began with the industrial sponsor's motivation to digitalise their methods of project management and to move away from the traditional approaches to project management in the construction sector. Because of this, the opted for a collaboration with the University of Strathclyde to carry out research into intelligent DSSs for construction project sustainability. This study is believed to be the first step in a direction of employing new digital technologies for improving their project management methods. Given the advancements of intelligent DSSs in construction and the evolution of the CSF in construction projects with a focus on sustainability, this research journey began with an initial investigation to address two questions:

(i) What are the current trends of DSS technologies in relation to the use of AI throughout the construction project lifecycle?

(ii) What are the current trends of DSSs in relation to construction project sustainability?

Hence, a systematic literature review was conducted to resolve these two questions which provided insight over knowledge gaps.

1.3 Research Focus

In the pursuit of enhancing construction project sustainability through an intelligent DSS, various approaches were explored. The most significant was focused on optimising the DSS performance. This not only contributes to the efficacy of the DSS for construction but also contributes to the field of hyperparameter optimisation (HPO) in AI as a whole. The literature review uncovered that the most used ML algorithm for prediction was the neural network. Nevertheless, it became apparent that the methods employed to establish the hyperparameters of the neural network models generally lacked a reliable approach to optimise the network architecture hyperparameters, resulting in suboptimal outcomes(Mohammad Kabir Yaqubi, 2019, Bala et al., 2014). It was also established that HPO is a crucial component of implementing ML in the construction industry (Bilal and Oyedele, 2020). With this, a newfound emphasis on HPO was determined, resulting in the potential impact of the findings and developments of this research extending beyond the original goal. HPO can be applied across a broader spectrum of applications, serving as a fundamental addition to the comprehensive body of knowledge surrounding AI.

It was established through the literature review that there is a significant gap in the field of learning curve prediction for HPO. Previous studies have primarily focused on using meta-learning to predict learning curves on new datasets by studying previous ones (Wistuba and Pedapati, 2020, Klein et al., 2017). Moreover, existing approaches to learning curve prediction have concentrated on halting poorly performing learning curves (Domhan et al., 2015). This leaves a gap for an approach that can incorporate both the training and prediction of learning curves on the same dataset, allowing for the prediction of the performance of fully unseen learning curves based on training a subset of the hyperparameter search space.

1.4 Research Aim and Objectives

This study aims to advance the field of hyperparameter optimisation (HPO) and learning curve prediction by developing an innovative approach that overcomes existing limitations in current methodologies. The specific objectives of this aim include:

1. **Overcoming Limitations in Learning Curve Prediction:** Address the constraints of existing learning curve prediction methods with a more integrated framework that utilises both training and prediction within the same dataset.
2. **Creating a New HPO Approach:** Introduce a novel hyperparameter optimisation technique that leverages the newly developed learning curve prediction model, enhancing predictive accuracy and efficiency.
3. **Validating Against Existing Benchmarks:** Conduct comprehensive validation of the new HPO method against established benchmarks to demonstrate its effectiveness and reliability in practical applications.
4. **Integrating the HPO Method into a Feature Importance Analysis Tool:** Develop a tool that combines the HPO method with feature importance analysis techniques,

enabling users to gain insights into the critical factors affecting model performance.

5. **Demonstrating Industrial Significance in a Decision Support System (DSS):** Showcase the applicability and relevance of the developed methodologies within a practical DSS, highlighting their potential to contribute to informed decision-making in real-world scenarios.

By addressing these objectives, this study aims to offer a valuable academic contribution in the form of a novel HPO approach which harnesses learning curve prediction and deep learning. A mechanism, titled SEquential LEarning Curve Training (SELECT), which can generate top performing neural network hyperparameters efficiently when compared to existing HPO methods based on the same computational effort. Additionally, the integration of the SELECT method into a decision-making tool can guide project managers in identifying the CSF related to project success with a capability to adapt to sustainability criteria in the future. The implementation of the proposed HPO method is anticipated to not only enhance current practices but also pave the way for more effective and sustainable construction project management strategies.

1.5 Research Novelty

A key novelty of the SELECT HPO mechanism is a method of transforming observed learning curve data into a sequenced and windowed training set, tailored for training a Convolutional Gated Recurrent Neural Network (CGRNN). This method, for the first time, addresses the challenges related to the significant variations in learning curves for new datasets and hyperparameter configurations. This novel approach is combined into a machine learning pipeline which culminates in a new HPO method. This novel HPO method can achieve high predictive accuracy, computational efficiency, as well as computational consistency. Further to this, this method is able to select high performing models which find the important relationships with data, verified through thorough analysis. This novel method contributes significantly to the field of HPO by enabling a more effective and efficient application of learning curve prediction. The impact of this novel method is explicitly demonstrated through all experiments in this study.

1.6 Thesis Structure

The structure of the thesis is described below.

- **Chapter 2 Literature Review**: The thesis will begin with a thorough review of relevant literature, providing context for the study and leading to the chosen gaps related to the optimisation of decision support systems and objective feature importance analysis for sustainable CSFs.
- **Chapter 3 Hyperparameter Optimisation**: The next chapter will delve into the development of the SELECT HPO method, outlining the method, the validation against multiple benchmarks, and the findings of high accuracy, computational efficiency and consistency.
- **Chapter 4 Feature Importance Analysis:** The combination of the SELECT HPO method and feature importance tools will be explored in detail, starting with a

comparison between existing HPO methods for feature importance, then analysis of the SELECT method combined with multiple feature importance tools. This will be conducted on both synthetic and real-world datasets, shedding light on its role and implications for detecting the relationships for CSFs.

- **Chapter 5 Decision Support System Development:** Insights into the construction of the DSS is provided, emphasising the seamless integration of the SELECT HPO and feature importance aspects, while presenting the additional functionalities of project trend prediction and the analysis of past project performance.
- **Chapter 6 Validating the Decision Support System:** The validation of the DSS will be presented and discussed. This will be in the form of a survey distributed to industry experts for a qualitative analysis of the DSS in its current level of development.
- **Chapter 7 Conclusion:** The conclusion chapter will summarise the key findings and implications derived from the study, providing a comprehensive wrap-up of the thesis.

1.7 Summary

The introduction of the thesis highlights the research aims and objectives. It starts by addressing the significant challenges in decision-making within the construction industry, particularly related to the nature of construction projects and the traditional approaches taken for decision making. With the evolving inclusion of sustainability criteria in the CSF for construction projects, there is a growing need for more intelligent decision-making techniques that can manage additional complexities and uncertainties. Intelligent DSSs have emerged as a viable solution, using advanced AI to manage complex data for practical interpretation. The introduction also discusses the research problem, leading to the primary goal of developing an innovative approach of HPO. This method aims to advance the field of hyperparameter optimisation with learning curve prediction. It will then be used to enhance the performance of a DSS in construction project management with the aid of feature importance analysis.

2 Literature Review

2.1 Introduction

The key focus of this chapter is to investigate the current literature on DSS technologies in construction project management with the use of AI for improving sustainability. More specifically, the two initial research questions to investigate were:

RQ1: What are the trends in research for using AI in DSS during the construction project lifecycle?

RQ2: What are the trends in relation to DSS and construction project sustainability?

This systematic literature review involved a comprehensive search within specific databases, using relevant keywords and strict inclusion and exclusion criteria to ensure the selection of relevant literature. Categorisation of the literature was done based on its application in the field of construction, the advancements for DSSs in relation to AI, and the trends in relation to sustainability, emphasising the gaps and limitations in the existing methods used. The literature findings highlighted the need of incorporating feature importance analysis within the context of sustainability goals. This also suggests the critical need for advanced HPO techniques to improve predictive accuracy of DSSs.

Building upon the established rationale, the review then delved into a thorough analysis of various HPO methods, emphasising the shortcomings and inefficiencies in existing practices, as well as a focus on the significance of learning curve prediction within the context of HPO, particularly on the limitations. These points will be addressed in Chapter 3 of this thesis.

2.2 Chapter Structure

This chapter is structured by first explaining background information related to DSSs, sustainability, AI, and the construction project lifecycle. The method of the literature review will then be explained, followed by the findings of the review. Specifically, the chosen gap related to the optimisation of AI for improving sustainability in decision-making will be justified. This helps to create the basis for further investigation into the current tools for HPO, especially the crucial gap in relation to learning curve prediction for HPO.

2.3 Background

To address RQ1 and RQ2, it is important to first discuss key concepts and terminology. The background information will cover what DSSs are, sustainability, key aspects in relation to AI and this study, and the construction project lifecycle.

2.3.1 Decision Support Systems

DSSs are a computer-based tool designed to aid project managers in complex decision-making tasks (Rao et al., 1994, Keen, 1980). Initially, DSSs were more passive, operating strictly based on user input and with limited decision-making capabilities (Rao et al., 1994). However, recent advancements in AI have significantly enhanced the capabilities of DSSs in the construction domain. Figure 2-1 illustrates the three fundamental components of a typical DSS. The system includes a user interface facilitating human-computer interaction, allowing users to input data for analysis and receive recommendations in a comprehensible format. At the core of the system lies the inference engine, employing mathematics, logic, and AI algorithms to perform complex reasoning and computations. Utilising data from the knowledge base and user inputs, the inference engine generates decisions or solutions to the presented problems. The knowledge base serves as a repository for decision-making logics and historical data, continuously updated with new knowledge from user interactions and real-world problem-solving scenarios. This constant updating process contributes to the advancement of the knowledge base and the overall intelligence of the DSS.

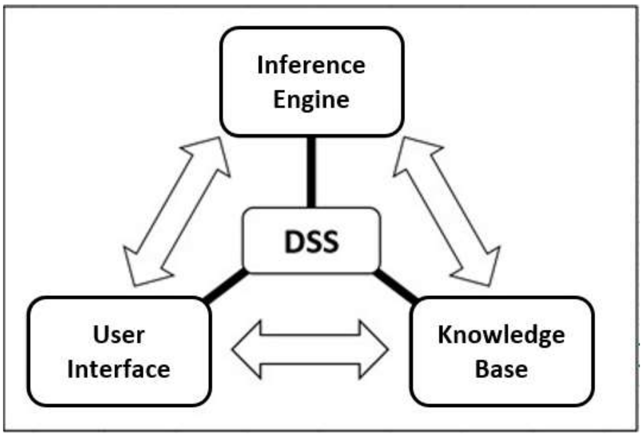


Figure 2-1 The three key components of a DSS

2.3.2 Sustainability

Project sustainability is dependent on the principle that the project should fulfil current requirements while safeguarding the concerns of the future. A commonly embraced perspective on sustainability revolves around the concept of the three pillars of sustainability: economic, environmental, and social objectives (Ranjbari et al., 2021). These are more informally referred to as the "three Ps": profit, planet, and people (Böcker and Meelen, 2017). The specific objectives of each are explained in the following sub-sections.

Economic Sustainability

The focus of economic sustainability is to ensure that there is a positive financial outcome in relation to the resources invested. This involves generating value and profits while also saving through careful cost reduction (Azapagic and Perdan, 2000). Common practices in relation to economic sustainability encompass efficient project management, adherence to established standards and regulations, as well as adept risk management and mitigation strategies.

Environmental Sustainability

Looking further than business considerations, the objective of environmental sustainability is to minimise the adverse effects of operations on the natural surroundings and to preserve and enhance the environment. This involves curbing energy consumption, restricting material usage, and adopting eco-friendly materials (Hong et al., 2021). The pursuits of environmental sustainability are frequently intertwined with economic sustainability, as the reduction of waste and the reducing the impact of inefficient practices can lead to financial savings and enhanced profits.

Social Sustainability

The aim of social sustainability is to uphold and enhance the standard of human life, encompassing customers, employees, contractors, and all other stakeholders affected by project activities. This is achieved through the enhancement of health and well-being, robust training and development initiatives, the promotion of workplace diversity, and active contributions to societal betterment (Fatourehchi and Zarghami, 2020). The advantages of prioritising social sustainability include boosted morale and well-being among company personnel, improved relationships with suppliers, customers, and involved parties, and the enhancement of local and global reputations.

2.3.3 Artificial Intelligence

AI refers to an area of science where systems are able to perform tasks normally requiring human intelligence, such as visual perception, reasoning, learning and decision-making(Nath et al., 2024). This research will primarily concentrate on the utilisation of AI for decision-making purposes. The subsequent section will outline different types of AI and HPO as well as feature engineering and feature importance.

Machine Learning

Machine learning (ML) is the process of developing computer programs that learn from past data to make predictions without being explicitly programmed to do so, with data driving the operation, rather than the programmers (Ethem, 2021). The learning methods include supervised learning; for labelled datasets where both the inputs and desired results are known in the training set, unsupervised learning; for unlabelled data where the desired result is unknown, and the dataset is analysed to recognise patterns and relationships between groups of data. There is also reinforcement learning (RL) for mapping from situations to actions to maximise rewards (Abioye et al., 2021). Examples of machine learning algorithms encompass multivariate-linear regression (MLR), logistic regression (LR), support vector machine (SVM), decision tree (DT), random forest (RF), K-means, Bayesian inference (BI), and artificial neural network (ANN).

Fuzzy Logic

In the real world, particularly in project management, there arise instances where human judgment is necessary for decision-making, often in the presence of uncertainties regarding the optimal choice. Fuzzy logic (FL) serves as a tool to address these situations, initially proposed in 1965 by Lotfi Zadeh (Bělohlávek and Klir, 2011). It is a technique to gauge the degree of accuracy of uncertain data, finding widespread applications in real-world systems to tackle intricate and ambiguous problems characterised by incomplete or imprecise information (Chen and Pan, 2021). Rather than measure something to be true or false, fuzzy logic enables the quantification the level of truth. Within the scope of the articles examined in this study, FL is predominantly employed to quantify expert knowledge derived from ranked questionnaires (Awad and Fayek, 2012), effectively capturing human reasoning for diverse decision-making applications.

Natural Language Processing

Human communication primarily relies on natural language such as English or Mandarin, in contrast to computer programming languages. To enable computers to interpret natural language, the application of natural language processing (NLP) becomes crucial (Hapke et al., 2019). NLP focuses on developing computational models that emulate human linguistic capabilities, encompassing reading, writing, listening, and speaking functions (Bilal et al., 2016). It serves to convert natural language into a machine-readable format, finding diverse applications in social media, customer service, e-commerce, education, entertainment, finance, and healthcare sectors (Hagiwara, 2021). Within the domain of construction project management, NLP facilitates the analysis of typed documentation and reports, enabling knowledge extraction for various purposes. For instance, NLP can aid in evaluating accident reports in the construction sector to identify precursors for potential accidents (Baker et al., 2020).

Evolutionary Algorithms

Evolutionary algorithms represent an interdisciplinary tool bridging biology, AI, numerical optimisation, and decision support, finding widespread applications across various engineering domains. These algorithms utilise organic evolution models to achieve intelligent optimisation (Back, 1996). Intelligent optimisation tasks typically involve the exploration for the most optimal outcome, either to minimise or maximise an objective function within specified constraints (Pan and Zhang, 2021). An illustrative example of such an algorithm is the genetic algorithm (GA). The GA initialises with a population of potential solutions to a problem, with each solution then evaluated against its fitness to solving the problem and the best of these solutions then selected for a new population. Selected individual solutions are paired together, combining their characteristics for the next generation of solutions and the addition of random changes, or mutations. This process is repeated until termination, with the GA converging through the evolution of generations of paired solutions (Kuptametee et al., 2024). This can be harnessed to optimise system outcomes, thereby enhancing model performance for decision-making in construction project management contexts (Cheng et al., 2010).

Hyperparameter Optimisation

During the training of ML models, parameters represent the changing variables that adapt to the training data, optimising their values to achieve the best performance. On the other hand, hyperparameters, established outside of training, serve as control values that govern the functioning of the ML model. The configuration of these hyperparameters significantly impacts model performance, emphasising the importance of attaining the optimal hyperparameter settings during training to yield the best results. Consequently, HPO has emerged as a critical area of research in recent years (Yang and Shami, 2020). With the increasing complexity of models, the number and nature of hyperparameters can significantly influence performance. HPO is the process of determining the best possible hyperparameter configuration to optimise model performance. This usually involves exploring a variety of hyperparameter configurations and determining how they impact the performance of the model (Shaziya and Zaheer, 2021), the better performing hyperparameters are then selected for use in prediction.

Feature Engineering and Feature Importance

Feature engineering is a process in ML that selects the most relevant features from the raw data to improve the performance of predictive models (Wei et al., 2019). Feature importance, a key aspect of feature engineering, refers to the technique used to determine the significance of each feature in predicting the target variable. It helps identify the most influential features that contribute the most to the model's predictive power (Musolf et al., 2022), allowing only the most important features to be used for predictive models, which can lead to increased accuracy and efficiency. By understanding feature importance, better informed decisions can be made about which features to prioritise and how to optimise the overall performance and interpretability of the ML models.

2.3.4 Construction Project Lifecycle

The construction industry encompasses a wide range of activities, including the construction, extension, installation, repair, and maintenance of various structures and infrastructures, such as buildings, transport routes, and water services (Ofori, 1990). This encompasses both commercial and residential applications, as well as the development of essential infrastructure components like water service stations and pipelines. Projects in the construction industry typically progress through five key stages: initiation, planning, execution, controlling, and closing (Vargas, 2001). A brief explanation of each stage is bulleted below.

- **Initiation:** The initial project approval marks the stage where the project's primary scope is established, and key stakeholders are identified.
- **Planning:** During this phase, the project plan is formulated, outlining the deliverables and requirements, the selection of human, machine, and material resources, and the documentation of a project delivery schedule.
- **Execution:** This phase signifies the practical implementation of the documented plan established in the previous stage, representing a crucial period during which a sizeable portion of the project's allocated time and resources are dedicated to the execution of planned activities and tasks.
- **Controlling:** This phase involves the thorough assessment of project execution outcomes in comparison to the initially documented plan, aiming to identify any disparities and subsequently address and rectify these discrepancies.
- **Closing:** The concluding stage of the project lifecycle typically involves the preparation of a comprehensive report detailing the project's outcomes, alongside the handover of deliverables to the client. Simultaneously, all services

and contracts are finalised, ensuring the relevant stakeholders are duly informed of the project's closure.

The studies discussed in this chapter of the thesis will be viewed through the five stated construction project lifecycle categories, with the research application of DSS in one or more of these stages.

2.4 Literature Review Methodology

The aim of a systematic approach to reviewing literature is to identify all the empirical evidence within a pre-specified inclusion criteria to answer a particular research hypothesis (Snyder, 2019). The nature of this method reduces subjectivity in the research, leading to a reduction in bias. This method also allows for a quantitative analysis of papers to determine overall trends and relationships within a study. This literature review was conducted using a systematic approach with three key stages, identification, screening, and assessment as shown in Figure 2-2.

2.4.1 Paper Identification

This review is to investigate the current state of AI-based DSS techniques in the construction sector to improve project sustainability. The key words, thus, were decision support system, construction, project sustainability, AI, and ML for literature searching within the three well-known databases: Scopus, Science Direct, and ProQuest. The search was conducted on the article titles, keywords, and abstracts. Only papers written in

English between 2010 to present and peer-reviewed articles were included. The relevant disciplines were Engineering; Computer Science; Mathematics; Business, Management, and Accounting; and Decision Sciences. This resulted in a total of 624, 1494, and 570 papers from Scopus, Science Direct, and ProQuest, respectively, for a grand total of 2688 papers to screen.

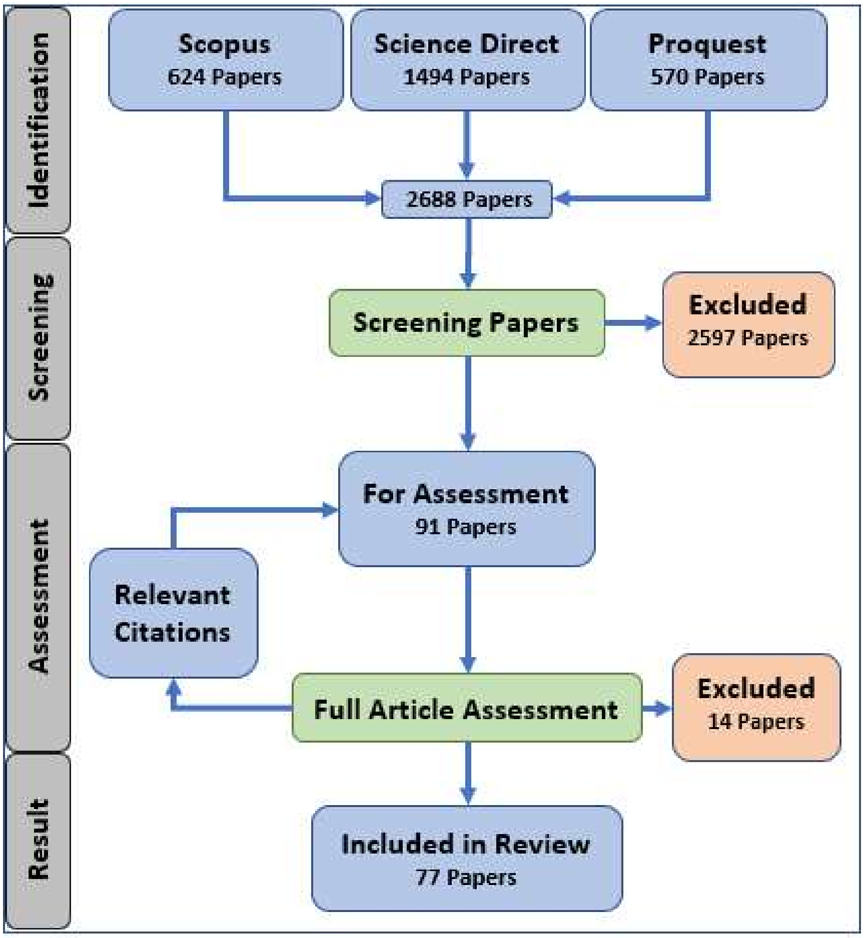


Figure 2-2 The systematic approach of literature review.

2.4.2 Screening

For screening, the abstracts were read for each of the identified papers. As there is a variety of frameworks that can be defined as a DSS (Kersten, 2000), papers were deemed as relevant only if DSSs or decision making were examined. Among all relevant papers, two levels were defined to differentiate relevance among them. The top level of relevance focused on papers that included research adopting AI with sustainability goals of construction projects. The second level of relevance included papers investigating any two of the three of adopting AI, sustainability goals, and construction projects for decision making and DSSs. These two levels of significance were used for the screening the papers. If papers achieved either of the two stated levels of relevance in the abstract, they would be included for full-article assessment. If these levels of relevance were not achieved, then the papers would be eliminated from the study. This resulted in ninety-one papers selected for the next stage of the review, which is the full-article assessment.

2.4.3 Assessment

The full content of each of the 91 remaining papers was assessed with respect to the criteria shown in Table 2-1, using a similar approach to (Zhang et al., 2019). A further fourteen papers were eliminated from the review at this stage; hence, only 77 papers remained, which included 9 literature reviews and 68 research papers for in-depth analysis.

Table 2-1 The assessment criteria of the literature review.

Data to Collect	Description
Task	Intended task of the DSS
Title	Title of the paper
Author (s)	List of authors
Contribution	Contribution to literature
Limitations	Potential improvements
Year	Year of publication
AI	Type of AI algorithm used
Sustainability	Economic, environmental, or social goal considerations
Stage of Construction	Lifecycle stage of operation
Institution	Location of the institution which carried out the study
Case study	Where is the case study located

2.5 Review Findings

This section describes the findings of this research. This will start with the categorisation of the papers by the task of the DSSs in all assessed papers and followed by the findings related to AI, sustainability, and the project lifecycle.

2.5.1 Categorising the Task of the DSSs

The areas in which a DSS may be applied in the construction project lifecycle varies in the forms of data being used, the tasks of the inference system, and when in the cycle these tools may be applied. The sixty-eight papers listed in this study have been organised into six distinct categories based on the task of the DSS. There is early-stage project prediction (EPP), which takes up 50% of all studies, with sub-categories focusing on various metrics for performance measurement; there is dynamic performance prediction (DPP), which takes up 17% of all studies; and then, there are papers focused on contractor and supplier evaluation, site logistics, design optimisation, and safety risk assessments (SRA). Looking at Figure 2-3, there has been an increase in studies over the second half of the last decade, which would suggest an increase in interest in this field. It can also be noticed that the EPP has a near-consistent level of interest throughout the decade with other areas such as SRA, site logistics, DPP, design optimisation, and supplier evaluation having more studies from 2016 onward. This shows a growth in the quantity of studies over this period but also a growth through increased variety of application.

Each of the chosen categories of DSS application were analysed against the type of AI used, the considerations for sustainability, and what stage in the construction lifecycle the system operates. The findings can be seen in Table 2-2. The following sub-sections will discuss the categories in more detail, followed by discussion on the use of AI, sustainability, and the project lifecycle.

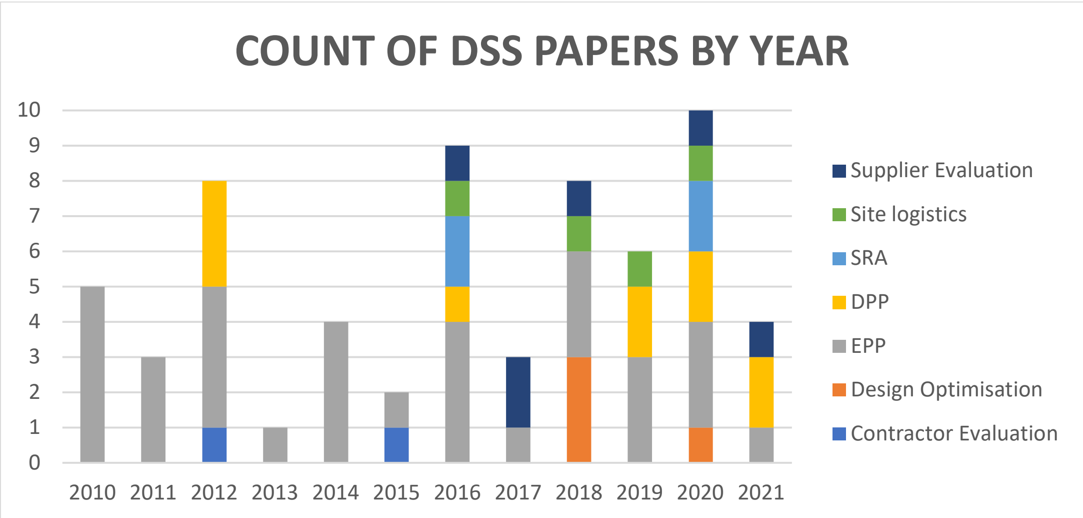


Figure 2-3 DSS Applications count by year of publication.

Table 2-2 DSS Application by AI type, sustainability criteria, and the stage of the construction project lifecycle

DSS Category		Type of AI Used												Sustainability Crit.			Stage of Lifecycle					
Category Name	**Example References**	**ANN**	**BI**	**CBR**	**CNN**	**DT**	**FL**	**GA**	**GD**	**MLR**	**RF**	**RL**	**SVM**	**Eco**	**Env**	**Soc**	**Init.**	**Plan.**	**Exec.**	**Cont.**	**Close**	**All**
Contractor and Supplier Evaluation	**(Omar et al., 2016, Khan et al., 2018, Awad and Fayek, 2012)**						6							7	5	6	1	1				
Design Optimisation	**(Minhas et al., 2018, Ferreiro-Cabello et al., 2018)**	1										1		4	4	2		4				
Early-stage Project Prediction	**(Ghazimoradi et al., 2016, Son et al., 2012, Wang et al., 2012, Soto and Adey, 2016, Shi et al., 2016)**	19		6		1	11	6		6	1		4	33	7	8	11	22				
Dynamic Performance Prediction	**(Choi et al., 2021, Marzoughi and Arthanari, 2016, Assaad et al., 2020)**	3	1	1		2	2	1			1		2	10	2	3			1			8
Safety Risk Assessment	**(2020, Tixier et al., 2016a)**	2			1		1		1		1		1			4			4			
Site logistics	**(Papadaki and Chassiakos, 2016, Greif et al., 2020, Jeong and Ramírez-Gómez, 2018)**						1	1						4	3	3		2	2			
	Total	**25**	**1**	**7**	**1**	**3**	**21**	**8**	**1**	**6**	**3**	**1**	**7**	**58**	**21**	**26**	**12**	**29**	**7**			**8**

Contractor and Supplier Evaluation

An area of research where DSSs have been applied with sustainability criteria using AI is for evaluating contractors and material suppliers. (Fallahpour et al., 2017) refined sustainability criteria for the selection of suppliers with the assistance of academics and industry experts for defining the importance of applicability of criteria taken from the literature. Fuzzy preference programming was then used to allocate weights to each of the sustainability criteria, resulting in graded levels of importance reducing from economic to environmental to social criteria. Fuzzy Technique for Order of Preference by Similarity to Ideal Solution (FTOPSIS) was then used for supplier selection. Others have also used a similar approach, with (Kannan et al., 2020) also using fuzzy logic for weight definition but using VIKOR for the selection of the projects. Another example is (Luthra et al., 2017), who used VIKOR for project selection but with analytical hierarchy processing (AHP) for the weight definition. The sustainable selection criteria in all these studies are great examples of a drive towards sustainability, and these are just some examples of the few studies into supplier selection in manufacturing (Kannan et al., 2020). It can be seen in these studies that all the evaluation criteria are defined through subjective opinions of experts related to the work, and there is a lack of quantitative data. Combining these two data types may prove beneficial for supplier selections. These are all focused on the manufacturing industry, which does have a different format from the construction industry for supplier selection and would have differences in the selection criteria based of the unique aspects of construction projects when compared to manufacturing.

A couple of studies were found that applied a similar approach to the supplier selection, for example, (Ulubeyli and Kazaz, 2015) created a framework software platform for the selection of construction project sub-contractors using fuzzy logic algorithms. This is done through providing linguistic ranking and marking questionnaires to experts at multiple stages in the selection process; then, fuzzy set theory is used to group, quantify, and rank sub-contractors. This is focused on international construction projects and is limited in that it does not factor in the built working relationships between contractors and sub-contractors. Additionally, all the data are provided based off expert opinion, so there is a level of subjectivity in the process of selection. Some factors that have not been studied in the recorded papers is the evaluation of contractors throughout the execution stage of the project lifecycle and the use of more complex AI models for tasks in addition to quantifying linguistic data.

Design Optimisation

This category focuses on the use of DSSs for improving design in construction. All papers related to this category were published from 2018 onwards, and sustainable design is the main driver for all of them. All papers related to design optimisation considered the economic and environmental goals of sustainability, while only half of them considered social sustainability.

(Minhas et al., 2018) highlighted ongoing research into the use of DSSs for sustainable building material selection in the design stages, with a key focus on incorporating criteria for the environmental goals of sustainability. (Santos et al., 2019) developed a DSS for helping design engineers to choose sustainable materials during the planning stage of construction for pavement design. This method not only considers economic, environmental, and social goals during the project lifecycle but also for the maintenance of the materials during the lifecycle of the product. An example of AI being used for DSSs for design optimisation would be (BuHamdan et al., 2020), who developed a DSS for concept-design decision making in the construction industry. They adopted a Markov decision process (MDP) and RL for this DSS. The aim of this model was to implement value engineering from the manufacturing section into the construction design phase. The focus was to achieve optimisation against environmental, economic, and social criteria. Using the MDP approach was especially useful, as the structure of this approach has similarities to the decision-making system that engineers manually carry out in the concept-design stage of construction projects. The method was tested using the concept design of a house, and the design was optimised, which showed a positive result; however, there is area for improvement by adding feedback complexity and representing the interdependencies between different decisions at different stages of design.

Early-Stage Project Predictions

The most popular application for a DSS from the last ten years is for making predictions of project performance at the initiation and planning stages of the project lifecycle. This can be for project cost prediction, project delays, and for risk in project selection. These areas of study all follow the same approach of utilising historical project performance data and key parameters to train an algorithm for predicting the resultant performance given the same input parameters for a new test project. This is an especially useful tool, as it provides the project manager with a quantifiable method for selecting which projects to choose during the initiation stage of the lifecycle or how best to plan for a project prior to execution. The most popular algorithms to be used are ANNs and more recent models, which include hybrids with FL for quantifying qualitative data and genetic algorithms (GA) for optimising the weights of the parameters (Tang et al., 2010, Bilal and Oyedele, 2020, Elmousalami, 2019, Cheng et al., 2010). Case-based reasoning (CBR) has also been studied, utilising previous similar cases of projects to make predictions (Marzouk and Ahmed, 2011, Zima, 2015, Kim, 2013, Koo et al., 2011, Car-Pusic et al., 2020). It can be observed in Table 2-2 that most of the research into project predictions examine the economic pillar of sustainability with 76% of all EPP research solely focusing on the economic sustainability goals. Research considering environmental and social goals is the minority, equating to approximately 25% of studies.

The ability to predict the cost of a project accurately has a significant impact on the economic sustainability of a construction project. This could help to ensure project success for choosing which project, equipment, or contractors to use or for determining the number of resources to provide. In construction engineering management, cost estimation at the start of a project is key to preventing cost overruns and ensuring project success (Martínez-Rojas et al., 2016, Doloi, 2013).

For improving the accuracy of predictions, (Alex et al., 2010) employed the use of an ANN to improve the prediction accuracy of water and sewer service project, as there were discrepancies of 60% error in predictions from standard practice in project cost estimation. Using this ANN, they managed to reduce the error down to 20%. This is a clear improvement although this level of inaccuracy is still high when compared to other studies in the construction industry, an example being (Bala et al., 2014), who created a model for predicting the cost of building construction projects in Nigeria, also using an ANN model for cost prediction. Based off refined input parameters from 243 questionnaires given to experts in the field and an ANN with two hidden layers and sigmoid transfer functions, a high prediction accuracy was achieved having the mean absolute percentage error of only 5%. However, such high accuracy might be a result of high similarity over different building projects. In other words, the robustness of this model had not been evaluated on other building types, and hence, the generalisation of the cost model is deemed low.

Most of the recent research into EPP has used hybrid AI models (58% of all EPP studies). (Yu and Skibniewski, 2010) combined an ANN with FL to create an adaptive neuro-fuzzy inference system (ANFIS) for making cost predictions alongside principal items ration estimation method (PIREM) for keeping accuracy with fluctuating market prices. This method managed to achieve a mean prediction accuracy of within 10% of the actual cost when evaluated on residential building constructions in China. Another hybrid ANN model is defined as the evolutionary fuzzy hybrid neural network (EFHNN), which is a high-order neural network hybrid that used fuzzy inference for dealing with project uncertainties and a GA for optimising the prediction accuracy. This model was assessed on 28 building projects and compared to a singular linear ANN with increased accuracy in predicting the overall cost of projects and cost per internal categories of expenditure.

Another method for cost prediction is the use of case-based reasoning (CBR). This is an experience-based solution relating previously successful solutions to similar problems that occur in the future. (Zima, 2015) presented a case-based method for predicting construction costs using sports field installations as a case study. This did prove to have a mean absolute percentage error of 5%, and the method is not computationally intensive; however, it is limited by the number and type of previous cases as well as the similarity of the new projects. The model is validated with the construction of sports fields, which has highly similar tasks. Applying this model independently on more complex construction projects would better measure its robustness. Another CBR-based prediction model found in (Marzouk and Ahmed, 2011) compared the CBR method with hybrid models of CBR+ANN and CBR+FL, with the CBR+FL model proving most accurate with an average prediction error of 9% for predicting the cost of pump station projects. Leading further evidence towards the benefits of using hybrid AI solutions.

As well as determining the project cost at initial stages, there is benefit from predicting performance against other metrics. Project delays can have a substantial impact on success; (Yaseen et al., 2020) created a method of categorising project delays in the construction sector by use of a random forest classifier with a genetic algorithm for result optimisation. This method split projects into three categories of delay: less than 50% overrun, 50–100% overrun, and greater than 100% overrun. This model proved to have a classification accuracy of 91.67% and was deemed better the random forest model on its own, again highlighting the advantage of hybrid models for prediction. Additionally, the range of classification is substantial, and an improvement on the classification metric may be of larger benefit, as not all sectors of construction will find it acceptable to have ranges of 50% of the total specified time for classification. Although the construction industry is known for delays in projects, there is surprisingly little research into the use of AI technologies for predicting project delay likelihood at least not to the same level of depth as project cost prediction.

From all the previously stated studies into early prediction, environmental and social parameters and goals were not considered in the estimations. When it comes to predicting project risk for project selection, sustainability criteria have been a topic of research. The research presented by (Fallahpour et al., 2020) introduced a method for selecting sustainability criteria for project selection and then used fuzzy preference programming for attribute weight selection and FL for aiding in the selection of projects. Fifteen different attributes, with five for each pillar, were selected from studying the literature and evaluation by three experts in construction engineering. Each of these attributes were given local (per attribute) and global (per category) weightings of importance and developed into 25 fuzzy rules within the system for defining the best alternative project to select. The system was evaluated using six projects from a construction company in Iran and compared with five other defuzzification methods and checked with a consistency index. Another study by (Akbari et al., 2018) pulls a larger area of expertise with input from fifty-three experts in the form of a questionnaire. The weights and rules are built through AHP and the novel rough set theory, respectively. This was assessed on classifying 26 projects against sustainability criteria with a prediction accuracy ranging from 84–95%. These studies show real promise for the use of sustainability criteria in project selection although the weighting criteria is based off the subjective opinions of the experts, and the testing is based off a small quantity of projects. Considering quantitative data alongside the qualitative data may be something that could improve the robustness of the predictions, it would also be advantageous to apply the selection criteria to a larger program of projects for evaluation.

For all the reviewed studies, accuracy is as much dependent on the area of application and available data as it is for the algorithms used. The benefits of hybrid AI models are clear for improving prediction accuracies and for improving the robustness of models. Although the primary goal for project prediction is on cost estimation, other areas such as project risk are being investigated, which consider the social and environmental goals of sustainability as well as the economic.

Dynamic Performance Prediction

The main limitations in the EPP papers are that once a project begins the execution stage, there is usually a great deal of uncertainty, which can affect the predictive capability regardless of how powerful the AI algorithm is or the completeness of the pre-execution data. This is due to the fluctuating nature of the time dependent variables in construction project management, such as internal factors related to human resources during project execution or external factors, such as the impact of the weather on progress. Over the last 6 years only a minority of studies have dynamically predicted performance throughout the execution and all other stages of the life cycle. This allows for project managers to make educated decisions at the planning stages and then proactively improve project performance at the execution stage through to the controls and closure.

The authors of (Cheng et al., 2010), who proposed the evolutionary fuzzy hybrid neural network (EFHNN) for project cost prediction at early stages, clearly understood the benefits of creating a dynamic performance-prediction tool. This hybrid is a combination of FL for dealing with uncertain data, a high-order ANN for making predictions, and GA for optimising the results. The same authors published a paper on their dynamic prediction performance method (Cheng et al., 2012), which used the same hybrid AI algorithms to classify the performance of projects throughout the lifecycle. This classified project performance into four levels ranging from successful to disastrous, with inputs related to 10-time dependent variables, including change order data, weather impact, owner commitments, contractor commitments, recorded incidents, and overtime work. This model is classified with a high accuracy; however, the method was only validated against the highly similar evolutionary fuzzy neural inference model (EFNIM) and with only twelve projects for training and 3 projects for testing. This work could be taken further by comparing the model with a larger pool of AI models and with a much larger dataset.

A DSS framework presented in (You and Wu, 2019) combines the use of a manufacturing enterprise resource planning (ERP) system with building information modelling (BIM) for the purpose of guidance on project management, materials management, financial management, and human resource management, which will optimise project processes with the use of machine learning algorithms in the execution and control stages of a project. This is just a framework now, but this has the potential for real value in the future. Further study that includes the application of this system and the evaluation of AI models for project optimisation is needed to gauge the overall effectiveness. When considering DPPs that consider sustainability criteria, (Dong et al., 2019) presented a framework for a sustainable construction project management index for evaluating construction projects. Six dimensions are defined: financial, scheduling, quality, safety, as well as informatisation and “greenization”. It is positive to see that research into dynamic construction performance measurement is being considered through the lens of sustainability. It is also key to note that from all studies into the DPP category, there are studies that have utilised AI and hybrids for improving the economic sustainability of projects (Choi et al., 2021), and there have been DPP studies that have considered all three goals of sustainability, but the use of AI models has not yet been seen to improve all three goals of sustainability in a single study of DPP.

Safety Risk Assessment

In the application of improving safety through the project lifecycle, there have been studies into the use of natural language processing (NLP) for analysing injury reports. (Tixier et al., 2016a) used NLP to structure data from accident reports into attributes of incidents and the safety outcomes and then used random forest and stochastic gradient tree boosting for prediction. The models were able to have better predictive capability for defining the injury type, energy type involved, and the injured body part with a higher likelihood than at random, which gives evidence to the use of quantitative and empirical methods for evaluating safety compared to that of expert opinion and subjective judgment. One of the authors then took the study further, such as (Baker et al., 2020), which introduced a method for automatically determining valid accident precursors for accidents in the oil and gas sector. Three different ML techniques were used and compared. These are convolutional neural network (CNN), hierarchical attention network (HAN), and term frequency-inverse document frequency representation with support vector machine TF-IDF-SVM, which were used for NLP. All predictions of precursors performed better than random selection, and the TF-IDF + SVM method proved to be the most accurate. The data collected for these reports were quantitative in nature, and circumstantial and environmental information that contribute to hazards in the workplace were not considered.

Site Logistics

Site logistics can be defined as the control of the movement of people, equipment, and materials related to a work site. In this paper, the category for site logistics covers all the DSSs, which focus on improving site logistics with the use of AI and sustainability criteria. (Greif et al., 2020) developed a digital twin and DSS, which applied heuristic optimisation and clustering for the purposes of silo replenishment on various construction sites during project execution. The purpose of this software tool is to predict the best routes for resupply vehicles to optimise vehicle usage and minimise work site stoppage times. Over a 3-year period, this reduced logistic costs by up to 25% and with every kilometre of transport saved having a positive impact on the development of CO2 emissions. The complexity of the digital twin and refill truck cost had a large effect on the cost reduction, which has left area for improvement.

Another study into improving logistics is (Guerlain et al., 2019), which covered the material transport routes, the emission levels and size of vehicles as well as the use of a construction consolidation centre for minimising the economic, social, and environmental impact of projects in Luxembourg. The multiple DSS were tested on one large project, producing 47 alternative combinations of the above variables, with 5 reducing emissions and cost. This would be especially useful for projects in densely populated areas. Although this system considers sustainability in construction, it is limited in that it relies on experience of experts and mathematics. Using AI instead for optimisation would provide a much larger pool of alternatives to consider for optimisation. Most site logistics studies have considered all three pillars of sustainability, but there is room for further study into the benefits of AI for optimisation considering hybrids to improve model performance.

2.5.2 Observations and Trends Related to AI

Table 2-2 shows a wide range of AI algorithms used in the literature and these various algorithms which are listed in Table 2-3 will be examined in more details.

Table 2-3 Artificial intelligence algorithms identified from the literature review.

Abbreviation	Algorithm Title
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
BI	Bayesian Inference
FL	Fuzzy Logic
CBR	Case Based Reasoning
DT	Decision Tree
RF	Random Forest
GA	Genetic Algorithm
GD	Gradient Descent
MLR	Multivariate Linear Regression
RL	Reinforced Learning
SVM	Support Vector Machine

The overall trend in AI shows that complex prediction in the form of ANNs and quantifying expert opinion using FL have had the most focus, which covered 37% and 31% of papers, respectively. GA is also popular for the optimisation of performance metrics, with 12% of studies considering this. CBR, which focuses on the use of previous cases to advise project managers on how to progress in future projects, and two other algorithms, namely MLR and SVM, have been involved in approximately 10% of all studies. In total, 46% of the studies used hybrids of multiple AI algorithms; this was not an increasing trend, though. As the quantity of papers increased over the decade, the ratio of hybrid models decreased. This reduction in the ratio of hybrid models does coincide with the increase in studies for other applications of DSSs than EPP. In all, 50% of all studies are focused on EPP, which can lead to a bias of the overall results towards the EPP research.

Contractor and supplier evaluation papers have used only FL for quantifying expert opinion (Fallahpour et al., 2017, Ulubeyli and Kazaz, 2015) while overlooking the potential of using ML techniques to examine empirical data alongside the opinion of experts. Design optimisation and site logistics have only a few studies that use AI, and there is no obvious trend, but these two categories were published within the last six years, suggesting an increase in interest; hence, the potential of AI has not been fully explored in these areas.

EPP is the most popular field of study and has been investigated with a wide variety of AI models. The most popular are the ANN and FL, with CBR, MLR, and GA also used in a considerable number of studies. DPP has also been involved a wide variety of AI models, the most common being ANN, followed by DT, FL, and SVM. The similarity between AI used for the EPP and the DPP makes sense, as the tasks required for both are highly similar. They need project data, which can be empirical or linguistic from previous experience, and a value or values need to be predicted from the data. The main difference between those two categories is the stage of the construction lifecycle in which they operate. EPP operates only in the initiation and planning stages of the project lifecycle, while the DPP studies operate in the execution and all areas of the project lifecycle.

The SRA papers focus on the use of NLP for analysing accident reports, with one paper (Baker et al., 2020) comparing multiple AI algorithms for interpreting the data from risk assessments. These studies base the analysis purely off the wording in the incident reports but fail to address the circumstantial data that accompanies them. All the SRA papers were published from 2016, which suggests an increase in interest.

These findings show that there is a wide variety of AI models used for construction DSS research. The type of AI used in research depends on the application of the DSS; however, some areas have been explored thoroughly, such as EPP, while site logistics, design optimisation, and contractor and supplier evaluation have very few studies that explore the benefits of AI.

The ANN has shown to be popular due to a prominent level of success, but this approach is highly dependent on the architecture and set parameters for training. These values, or hyperparameters, can significantly impact the accuracy of the predictions being made (Tayefeh Hashemi et al., 2020). This merits the need for the optimisation of the hyperparameter as a key stage to include in the use of machine learning in the construction sector for decision making. (Bilal and Oyedele, 2020) explains in their guide for machine learning in construction that the lack of consideration towards this can lead to less-than-optimal results, while including HPO as a key stage in their process of implementing ML algorithms in construction. (Tapeh and Naser, 2023) also highlights the need for HPO when models are performing poorly and to achieve satisfactory results while mentioning that future studies should investigate HPO. This point is agreed upon by (Akinosho et al., 2020) who concludes that HPO is critical for optimal model performance, stating that the omission of such a step could result in models which do not meet expectations.

The lack of consideration for HPO for ANNs in DSSs from the reviewed studies is evident. The approach taken by (Alex et al., 2010, Wang et al., 2012, Ferreiro-Cabello et al., 2018, Yousefi et al., 2016, Ebrat and Ghodsi, 2014, Patel et al., 2023) was of trial an error, manually selecting the best model. With experience this can be effective in achieving a proficient level of performance, but this can be time consuming and there is no guarantee that the optimum performance has been achieved. (Mohammad Kabir Yaqubi, 2019) used a simple equation to determine the architecture of the ANN while (Wen, 2010) employed a similar approach. This method may be quick but cannot encompass the complexity of the combination of hyperparameters which contribute to the optimum performance in the ANN algorithm. Another approach taken was simply the selection of the hyperparameters without the inclusion of HPO by (Baker et al., 2020, 2020, Car-Pusic et al., 2020), with a number of studies failing to address the approach taken for HPO at all (Chaovalitwongse et al., 2012, Williams and Gong, 2014).

In the review of the literature, only a single study included an automated HPO method for the ANN. (Cheng et al., 2012), employed the GA for the optimisation of the architecture of their neural network alongside other components of their evolutionary fuzzy hybrid neural network (EFHNN). The GA algorithm is an effective tool for HPO although it requires its own hyperparameters to be set and can have a slow convergence rate on finding the optimum parameters (Bischl et al., 2021, Del Buono et al., 2020).

2.5.3 Observations and Trends Related to Sustainability

Table 2-2 reports that 87% of studies considered the economic goal of sustainability while only 30% and 38% of studies included criteria from the environmental and social goals, respectively. As the societal switch to the consideration of environmental and social goals is recent, this is a reasonable result. A key factor to consider is the timeline for papers published over the last decade. From Figure 2-4, there is not only an increase in the number of papers published, but there is also a significant increase in the ratio of social and environmental goals being considered.

It must be noted that this coincides with the increase in papers focused on design, site logistics, safety, and both supplier and contractor evaluation as shown in Figure 2-3. These studies have been noted to have a high percentage of consideration for the environmental and social goals of sustainability.

EPP primarily focuses on the economic goal of sustainability when viewing bidding, claims, and cost prediction; however, there has been an area of EPP focused on project risk, of which most studies consider the three sustainability goals (Taylan et al., 2014, Hatefi and Tamošaitienė, 2019). This field of study has increased in regularity in the second half of the decade. DPP has the smallest ratio of consideration for goals other than the economic goal of sustainability; however, there were a couple of studies in 2019 and 2020 that adopted an approach towards all goals of sustainability (Dong et al., 2019, Lee and Yu-Lan, 2020).

These findings show that there is a shift towards research into project sustainability, with an increase in studies specifically aimed at improving sustainability in the latter half of the decade.

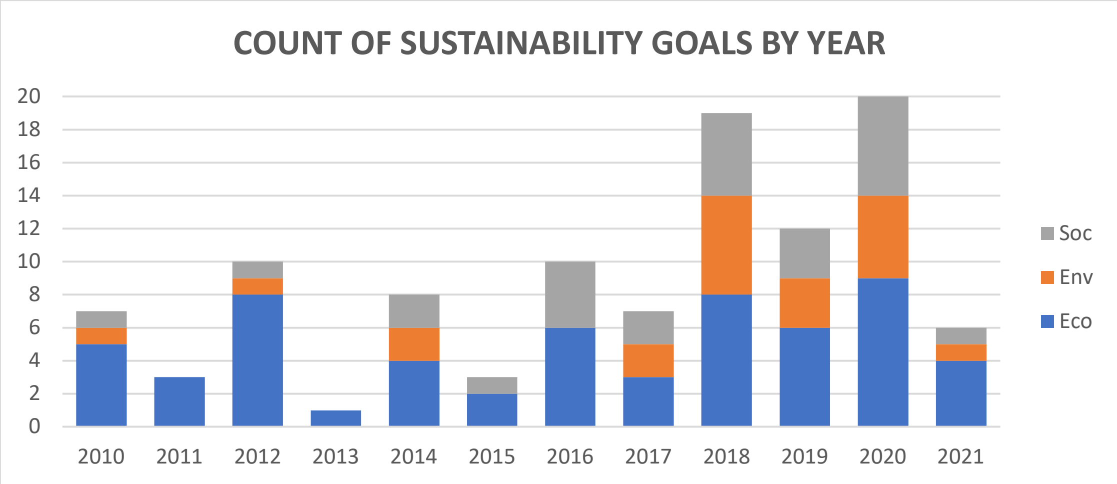


Figure 2-4 The three goals of sustainability by year of publication.

When evaluating sustainability criteria across studies, a prevalent trend emerges in the selection and prioritisation of key sustainability factors. Commonly, researchers gather relevant criteria from existing literature and determine their significance through expert opinions (Fallahpour et al., 2017, Alavi et al., 2021). Some studies, such as (Kannan et al., 2020) and (Luthra et al., 2017), employ FL and AHP to further refine subjective expert judgments. While this method is used across various studies assessing project sustainability, it can introduce biases and yield imperfect results.

Despite the incorporation of mathematical tools to refine expert opinions, inherent subjectivity remains a limitation compared to an objective approach which draws knowledge from quantified data and feature analysis. Only one study, as identified among the reviewed literature (Santos et al., 2019), integrates both objective and subjective data in the assessment of sustainability criteria. This combined approach uses a mixture of subjective and objective weighting methods, providing a comprehensive evaluation of various sustainability indicators in the context of road pavement decision-making. However, a potential drawback of this approach lies in its limited capacity to capture intricate interdependencies between different indicators, oversimplifying the complex relationships within sustainable pavement construction. Additionally, the reliance on predetermined weights may not account for dynamic changes in indicator importance over time, potentially leading to inaccuracies in prioritising sustainability factors.

2.5.4 Observations and Trends Related to Project Lifecycle

Each of the papers were evaluated for the stage of project in which the operations of the decision tools were focused. The stages, as stated in Table 2-2, are initiation (init.), planning (plan.), execution (exec.), controls (cont.), and close. There are also studies that focus on the whole project lifecycle rather than any single stage. No study was recorded as solely operating in the controls or closing stages.

About half (43%) of all papers under study were focused on decision support in the planning section, and a further 18% were focused on the initiation stage of construction. Figure 2-5 highlights that there has been a consistent production of publications dedicated to the planning and initiation stages of the project lifecycle. Research focused on the execution stage of the lifecycle has gained attention from 2016 onwards. The same trend is seen with studies that cover all five stages of the lifecycle although there was a spike of four papers in 2012 that covered all stages.

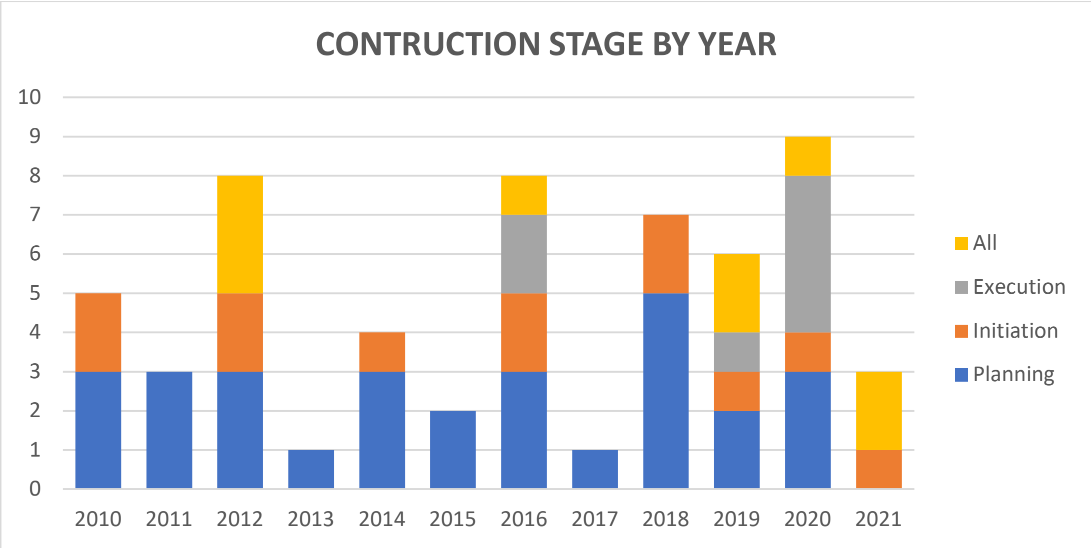


Figure 2-5 The count of projects focused on the stages of the project lifecycle against the year.

Analysing by category, EPP focuses on making predictions at the initial stages of the construction project lifecycle prior to the execution of the project plan. This is separate from DPP, which has similar characteristics but operates through the whole project lifecycle. Contractor evaluation is also only researched at the beginning of the project lifecycle in all selected papers (Awad and Fayek, 2012, Ulubeyli and Kazaz, 2015) despite contractors operating through the project execution. The design-optimisation papers also only focus on the planning stage, but this is understandable; excluding reworks, all design is completed in the planning stage of the construction project lifecycle. The SRA studies understandably focus on the execution stage of the project lifecycle, as this is where the largest risks to health lie. Although the studies that focus on supplier evaluation are relevant to this research, as they include the use of AI for decision making and consider sustainability criteria, the supplier evaluation papers were focused on manufacturing projects, so these studies do not fit into the construction project lifecycle and were not considered for this stage of the analysis.

2.6 Knowledge Gaps

Regarding RQ1, the observations made in this study suggest that there will be an increase in studies into DSS technology in the future, with AI/ML being used for applications that cover all stages of the project lifecycle and for applications in management, logistics, and design. Regarding RQ2, although the economic goal of sustainability has been the focus of most studies, there is a clear rise in research that investigate the social and environmental goals through all applications of DSS technology. This suggests that there will be further studies considering all three pillars of sustainability in the future over the whole project lifecycle. Looking at more specific examples of potential study, the following sections will highlight gaps in the literature and future avenues of research.

2.6.1 Contractor Evaluation

Contractor evaluation was only ever considered at the point of selection during the planning stage of the project lifecycle and using only subjective data from experts via questionnaires. Contractors can have a long-lasting effect on a single project. An investigation of contractor performance throughout the project lifecycle against sustainability success criteria is a good avenue for future research. For data collection, a sustainability performance questionnaire could be created and distributed at regular intervals and stages of the project lifecycle combined with available empirical measurements of contractor performance. For inference, a hybrid of FL for handling uncertain data and a machine learning model, such as the ANN, for making predictions and determining the trends in performance throughout the project lifecycle. This could lead to improved project and contract work efficiency and a potential metric for ongoing sustainable contractor evaluation throughout the whole project lifecycle.

2.6.2 Design Optimisation

AI-based DSSs for design optimisation in construction appears to be a new area of study with all research being published from 2018. There has been some work investigating sustainability goals; however, the benefits of different and hybrid AI models have not been fully explored yet. The primary stream for DSSs in design optimisation is the optimisation of material selection choices for sustainability. This has had minimal AI use for inference engine design until now, as there is only a single study recorded to have used AI for sustainable material selection. For future research, studying the benefits of multiple models of AI or combinations for material selection would be a fruitful avenue to pursue. In addition, consider other steps of the decision-making process during design. (BuHamdan et al., 2020) uses RL for decision making in concept design; this is only effective in this study due to the highly similar nature of the designs being produced. With a dataset covering multiple design projects, a supervised learning approach, such as the use of an ANN (Ferreiro-Cabello et al., 2018), could be used for optimising concept design for sustainability.

2.6.3 Dynamic Performance Prediction

There is an increasing trend towards continuous performance prediction throughout the project lifecycle. At present, there have been studies that consider AI models and others with sustainability criteria, but a study focused on the continuous measurement of performance against sustainability criteria using an AI inference engine is an avenue to be pursued. This can be investigated with the intention of determining the readiness for this transition in the construction sector or the development of a framework to achieve this. An example of an approach that could be taken for this would be (Bilal and Oyedele, 2020), who developed a six-stage guideline for applied machine learning in construction. It starts with problem definition and data selection and then data preparation and pre-processing, training the baseline estimator, creating interpretable machine learning, training the final estimator, and deployment and scoring. This is a comprehensive guide that could be applied for predicting a variety of performance metrics. A challenge related to this avenue of research would be for the collection and verification of newly defined data at regular intervals related to sustainability criteria over multiple construction projects. Furthermore, the complexity of the inference engine would need to accommodate data for prediction, which can change throughout all stages of the project lifecycle and work with incomplete and both linguistic and empirical data.

2.6.4 Safety Risk Assessment

Another future direction would expand on the SRA field of study. The use of near miss reporting as a dataset, considering circumstantial information alongside the quantitative methodology stated in (Tixier et al., 2016a, Baker et al., 2020, Tixier et al., 2016b), may be an avenue to pursue. This may allow for other forms of AI to be used. Using near-miss reporting, according to the Heinrich accident triangle (Marshall et al., 2018), there are approximately 300 near misses for every 30 minor accidents and 1 major accidents. This would be a much larger pool of data for determining trends in accidents at work. This also allows a company to pre-emptively reduce accidents through analysis of near miss reporting. Challenges to this may be in the difficulty in determining an accurate metric for showing the improvement made by the system.

2.6.5 Site Logistics

Improving the efficiency of site logistics has shown to improve the sustainability of projects, as discussed in Section 4.1.6, but there is opportunity for further study in utilising AI models for the decision-making process. (Guerlain et al., 2019) developed a system that looked at the material transport routes, the emission levels from vehicles, and the size of the transport vehicles but relied heavily on the experience of experts. This produced only 47 alternatives to choose from for increasing sustainability. Using this same approach but incorporating the predictive capability of machine learning algorithms such as ANN would optimise the resultant transport routes. The nature of training a neural network would lead to the consideration of a significantly larger pool of alternatives to compare from and objectively identify the optimal course of action to take. Studying the benefits of AI-based DSSs for material supply for improving logistic sustainability can be seen as a good focus for future research.

2.6.6 Hyperparameter Optimisation

The review of the literature in this study has highlighted a notable gap concerning the optimisation of hyperparameters. While the use of ANNs has gained traction due to their successful outcomes, the failure to adequately address the optimisation of these hyperparameters remains a persistent issue (Tayefeh Hashemi et al., 2020, Bilal and Oyedele, 2020, Tapeh and Naser, 2023, Akinosho et al., 2020). The significance of this gap becomes evident as numerous studies, including (Alex et al., 2010, Wang et al., 2012, Ferreiro-Cabello et al., 2018, Yousefi et al., 2016, Ebrat and Ghodsi, 2014, Patel et al., 2023), rely on manual selection or simplistic equations to determine the architecture and hyperparameters of the ANN.

Although these methods might yield acceptable results, they often fall short of achieving the optimum performance and fail to account for the intricate interactions among various hyperparameters. Additionally, a substantial number of studies, such as (Baker et al., 2020, 2020, Car-Pusic et al., 2020, Chaovalitwongse et al., 2012, Williams and Gong, 2014), either select hyperparameters arbitrarily or neglect to highlight the optimisation process entirely, thereby leading to suboptimal model performance. Despite the existence of advanced techniques like Genetic Algorithms (GAs) for HPO, as demonstrated by (Cheng et al., 2012), the slow convergence rate and the need for setting additional hyperparameters can pose challenges to the effective implementation of these methods.

This gap underscores the urgent need for a standardised and automated HPO approach within the context of utilising ANNs for DSS in the construction sector, ensuring the attainment of optimal model performance and accurate decision-making processes.

2.6.7 Sustainability Criteria Importance Evaluation

The literature review findings suggest that the assessment of sustainability criteria frequently involves the insights of industry experts to determine the significance of key sustainability factors. Notably, certain studies, referenced in (Kannan et al., 2020) and (Luthra et al., 2017), utilise advanced methodologies such as FL and AHP to refine the subjective judgments of experts. However, while these approaches integrate mathematical tools to improve the credibility of expert opinions, their inherent subjectivity remains a prominent constraint.

Unlike these studies, which heavily rely on expert perspectives, a solitary work identified in the literature review (Santos et al., 2019) adopts a dual approach, integrating both objective and subjective data in the assessment of sustainability criteria. Despite the merits of this approach, it is crucial to acknowledge its potential limitations in capturing complex interdependencies among different indicators.

It is evident that the existing methodologies lack a framework for the effective quantitative assessment of sustainability criteria in construction. The persistent reliance on subjective assessments and expert opinions often leads to biased and inconsistent evaluations, negatively impacting the reliability of the established sustainability standards. Further to this, the limitations in current methodologies to comprehensively capture the intricate interdependencies between various sustainability indicators result in an oversimplified understanding of the complex decision-making landscape, restricting the ability to effectively prioritise sustainability initiatives and address interconnected challenges.

This creates a need for the development of a robust and objective methodology that minimises subjective biases, empowering decision-makers to make well-informed and reliable sustainability evaluations, leading to more effective and informed decision-making.

2.7 The Selection of the Knowledge Gaps

Based on the literature review findings, two critical gaps have been identified with two reasons: (i) these gaps are believed to have a significant impact on construction project sustainability; and (ii) it is feasible to address these gaps even if sustainability data is limited, as discussed in the introduction section.

The first gap is about the optimisation of ANNs hyperparameters, particularly in the context of DSSs in the construction sector. It must be noted that the literature indicates a persistent reliance on manual selection or simplistic equations for determining ANN hyperparameters (Mohammad Kabir Yaqubi, 2019, Ferreiro-Cabello et al., 2018, Yousefi et al., 2016, Wen, 2010), resulting in suboptimal model performance. While the GA has been explored to optimise ANN hyperparameters, challenges such as slow convergence rates and additional hyperparameter settings can impact the effectiveness of this approach (Bischl et al., 2021, Del Buono et al., 2020). This calls for the development of a standardised and automated HPO approach tailored specifically for ANNs, ensuring optimal model performance and accurate decision-making.

The second gap concerns the evaluation of sustainability criteria, which often relies heavily on subjective assessments and expert opinions. While certain methodologies integrate mathematical tools to refine expert judgments, their inherent subjectivity remains a notable limitation (Fallahpour et al., 2017, Alavi et al., 2021, Kannan et al., 2020). Furthermore, the existing methodologies lack a comprehensive framework for effectively quantifying sustainability criteria in the construction domain, leading to biased and inconsistent evaluations that can undermine the reliability of established sustainability standards. The limitations in current methodologies to capture the intricate interdependencies among various sustainability indicators further restrict the ability to prioritise sustainability factors. Therefore, there is a pressing need to develop an objective methodology that minimises subjective biases, facilitating well informed and reliable sustainability evaluations to enable more effective decision-making in the construction industry.

By concentrating on HPO for ANNs in DSS and the development of an objective methodology for evaluating sustainability criteria, this research endeavours to contribute to the enhancement of decision-making in the construction industry.

The primary objective is to establish a standardised and automated HPO approach tailored specifically for ANNs in the construction sector, ensuring optimal model performance and accurate decision making. Additionally, this study aims to establish an objective framework that can be used to accurately assess the significance of quantified sustainability criteria, hence reducing subjective biases, and leading to more reliable sustainability evaluations. By addressing these two gaps, this research aims to lay the groundwork for enhanced practices and sustainable decision-making strategies in construction project management.

2.8 Summary

In this chapter a systematic literature review was conducted to investigate the trends in research for the use of AI in DSSs in the construction project lifecycle, RQ1, and to investigate the trends in relation to DSS and construction project sustainability, RQ2. This resulted in the full paper investigation of seventy-seven studies after screening which were categorised by the application of the DSS, the AI used, the sustainability criteria considered and the stage of the project lifecycle in which they operated.

The most popular application for DSSs was EPP which has been a topic of research consistently throughout the investigation period, while other applications such as DPP, design optimisation, site logistics and SRA were the focus of an increasing number of studies in the latter half of the last decade.

AI has been used throughout a significant amount of the studies with the ANN and FL being the most popular forms of AI to be employed, there has also been an increasing trend in the number of hybrid approaches taken to overcome the weaknesses in the individual AI models.

Economic sustainability has been the primary pillar of sustainability to be considered in the reviewed literature although the other two pillars, the social and environmental goals, have been gaining more attention as time has passed.

Several gaps in the literature were identified, related to each of the categorised applications of DSS, however the most significant gaps to be discovered were in relation specifically to AI and sustainability. These two gaps are in relation to the lack of optimisation for the popular and effective ANN for DSSs, and the lack of objective evaluation of sustainability criteria, considering the complex interdependencies between contributing factors. Addressing both gaps is the selected focus of the research in this thesis. The aim of this research is now to develop an intelligent HPO method for optimising the performance of ANNs for decision making and create an effective method of objectively evaluating sustainability criteria in the construction sector.

This leads onto the following chapters which cover the development of the novel HPO method, the experimentation and validation of the approach for feature importance analysis, the integration of this novel approach into a DSS in collaboration with the industrial sponsor, and the validation of the DSS from expert evaluation.

3 Hyperparameter Optimisation

3.1 Introduction

Since the rise of industry 4.0 and the age of big data, the ANN has become one of the effective tools for developing predictive models of reality. Due to its versatility, ANNs have been applied through all areas of industry from metallurgy and material science, chemical engineering through to computing and manufacturing (Suzuki, 2011, Mumali, 2022, Nagy et al., 2022). The ANN performance relies on its network architecture which is a function of multiple factors. The effect of these factors, also known to be hyperparameters, vary for different datasets. The process of finding the best setting of hyperparameters can be very time-consuming and yet there is no guarantee that such settings are truly optimal. Therefore, automatic HPO has attracted a lot of attention, and its main objective is to maximise the ANN performance within the shortest time. Established HPO methods from the literature will be discussed, leading on to an introduction of a new HPO method, titled the SEquential LEarning Curve Training (SELECT) method, representing both the novelty and contribution of this research. Some of the well-established HPO Methods are discussed below.

Grid Search and Random Search

Grid search (GS) represents a brute force method of HPO that evaluates the model for all hyperparameter configurations within the defined search space, selecting the best performing model. GS can suffer from the curse of dimensionality, as the number of combinations grows exponentially with the addition of hyperparameters. This can lead to significant computational inefficiency and increased time for convergence. Despite its exhaustive nature, GS may miss optimal configurations that lie between the grid points, which limits its effectiveness in high-dimensional spaces. Also, GS does not incorporate any mechanism for adaptation; it evaluates each combination independently, making it less effective at optimising resource allocation. This method can prove time-consuming and computationally expensive, especially with a larger range of hyperparameters to evaluate (Antal-Vaida, 2021).

On the other hand, Random Search (RS) randomly selects configurations from the pre-defined search space, demonstrating better performance with improved efficiency compared to GS (Bergstra and Bengio, 2012a). While RS is more efficient than GS, it still lacks a directed search mechanism. Its random selection may not effectively sample the hyperparameter space, potentially leading to configurations that do not explore the full potential of the model. The simplicity of RS does allow for easy implementation and parallelisation, making it a popular choice for many practitioners.

Sequential Model Based Algorithms

Sequential Model-Based algorithms, such as Bayesian Optimisation (BO), leverage previously selected configurations to identify the best hyperparameter choices within the search space (Snoek et al., 2012).

These algorithms are efficient because of their ability to model uncertainty and balance exploration and exploitation in the hyperparameter space. However, the performance of these algorithms heavily depends on the choice of surrogate models and acquisition functions, which can introduce bias if not selected appropriately. As a result, user of these approaches must carefully consider these factors to maximise the effectiveness of Bayesian Optimisation.

Although BO has proven to be effective in optimisation within short times as compared to GS and RS, it still needs to spend time trialling multiple combinations of the hyperparameters to begin to converge on the best results. BO is effective for lower-dimensional search spaces, it may struggle with high-dimensional problems where the search space is vast. This challenge often necessitates the use of advanced techniques like dimensionality reduction or hybrid approaches that combine various optimisation methods to ensure thorough exploration of the hyperparameter space.

Furthermore, BO-based approaches have gained great popularity in recent years (Shahriari et al., 2016), with multiple variations such as Tree-Structured Parzen's Estimator (TPE) (Bergstra et al., 2011) and SMAC3(Lindauer et al., 2022) using different surrogate methodologies for better performance with higher dimensionality. However, these methods share a common restriction when being used to fine-tune neural network architectures. They will continue to run an iteration until the set number of epochs is reached even if the model performance is clearly not optimal.

Bandit Based Strategies

Bandit-based strategies such as successive halving can be adopted to improve computational efficiency (Jamieson and Talwalkar). This strategy trains multiple network configurations in parallel, allocating further computational cost to promising training iterations. This approach facilitated the creation of the popular Hyperband (HB) approach to HPO (Li et al., 2017) by combining the RS approach to parameter selection with parallel successive halving. Another hybrid approach advocates the integration of the two powerful algorithms, BO and HB, to create BOHB (Falkner et al., 2018). This approach uses BO to direct the selection of new trial iterations while limiting the wasted computational cost with HB.

These methods offer the advantage of efficiently allocating resources by evaluating multiple configurations simultaneously. This parallel approach allows for quicker identification of promising models, reducing overall computational costs. Also, these strategies can adaptively adjust their resource distribution based on performance, ensuring that more computational power is directed toward configurations that show potential, leading to faster convergence in hyperparameter optimisation.

Although the BOHB and HB approaches carry out directed searches and reduce the wasted time in training, these approaches have the potential to eliminate trial iterations that may converge to the best performance for datasets with large variations in convergence rates with various ranges of hyperparameters, a challenge stated in the original paper for HB (Li et al., 2017).

Evolutionary Algorithms

Another approach for HPO is the use of evolutionary algorithms, which emulate biological evolution by employing techniques such as mutation, crossover, and selection to solve complex problems, an example being the GA. Evolutionary algorithms operate by maintaining a population of potential solutions and continually refining them over iterations, being guided by a predefined fitness function.

Evolutionary algorithms are particularly advantageous in scenarios where the search space is non-convex or poorly understood, as they do not rely on gradient information, making them suitable for a broader range of optimisation problems. The flexibility of these algorithms allows for the incorporation of domain knowledge through custom-designed fitness functions, which can guide the search process toward more relevant regions of the solution space. The ability to handle multi-objective optimisation problems is another strong point, as these algorithms can simultaneously consider multiple performance metrics, providing a more comprehensive evaluation of hyperparameter configurations (Morales-Hernández et al., 2023).

Despite their strengths, the challenge of tuning the various parameters inherent to evolutionary algorithms, such as population size and mutation rates, remains a critical factor that can significantly influence their performance and convergence behaviour. These approaches can have large computational costs for convergence, can potentially converge in local minima and can be difficult to generalise(Wei et al., 2022, G et al., 2022, Liu et al., 2023). A significant limitation of evolutionary algorithms lies in their complex and less straightforward adaptability to handle varying numbers of features, rows, and regression variables. While methods such as GA and particle swarm optimisation (PSO) may require careful and specific adjustments to accommodate mixed input spaces, alternative approaches like BO processes and TPE demonstrate a more robust and flexible performance in handling diverse data types, including discrete, categorical, and numerical variables (Morales-Hernández et al., 2023).

Learning Curve Prediction Methods

To improve the efficiency of HPO, recent studies have begun to investigate the benefit of learning curve prediction for terminating poorly performing hyperparameter configurations. (Domhan et al., 2015) used a model agnostic probabilistic model for early termination of poorly performing models, or (Baker et al., 2017) who combined a support vector regression mode to predict the final accuracy based on extracted features of the learning curves, the network architecture and the gradient of the learning curves. A later approach is (Wistuba and Pedapati, 2020) which utilises pairwise ranking loss and leveraging learning curves from other datasets to improve the effectiveness of early termination so that fewer and shorter learning curves can be used for the early termination. A similar objective was attempted by (Sui and Yu, 2020) who used Bayesian contextual bandits for HPO, terminating trials of poorly performing configurations with intelligent resource allocation from learned trends in performance.

A common drawback in all the above studies is that the poorly performing configurations are terminated only by comparative elimination during the trial for each configuration, meaning that training is still required to evaluate the performance of each configuration.

Previous studies have primarily focused on using meta-learning to predict learning curves on new datasets by studying previous ones (Wistuba and Pedapati, 2020, Klein et al., 2017). Moreover, existing approaches to learning curve prediction have concentrated on halting poorly performing learning curves (Domhan et al., 2015).

There has yet to be an approach which can learn the relationship between hyperparameters and learning curve performance, incorporating both the training and prediction of learning curves on the same dataset, allowing for the prediction of the performance of fully UNSEEN learning curves based on training a subset of the hyperparameter search space.

3.1.1 Foundation for Proposed HPO Methodology

Table 3-1 outlines the strengths and limitations of established HPO methodologies. Sequential HPO methods like BO and TPE follow a sequential convergence process, training multiple configurations to observe the relationship between hyperparameters and guide optimisation toward optimal configurations. However, their sequential nature limits parallelisation; the capability to run different trials in parallel to reduce the time taken for HPO. This is due to the need for the results from previous iteration to guide the convergence to the optimum result. There is also inefficiency in completing the training of poorly performing configurations.

Table 3-1 Pros and cons for existing HPO methods

Method	Pros	Cons
Grid Search	• Exhaustively searches the hyperparameter space. • Simple and easy to understand.	• Computationally expensive. • Does not adapt based on observed performance.
Random Search	• Simplicity and ease of implementation. • Requires minimal tuning. • Can perform well with a low computational cost. • Suitable for parallelisation	• Inefficient in finding optimal hyperparameters. • Does not adapt based on observed performance. • May waste resources on less promising configurations.
Bayesian Optimisation	• Efficient in handling noisy or expensive objective functions. • Adaptive exploration of the hyperparameter space. • Converges to optimal solutions with few evaluations.	• May waste resources on less promising configurations. • Poor capability for parallelisation.
Tree Parzen's Estimator	• Efficient with all kinds of hyperparameters. • Balances exploration and exploitation effectively.	• Performance may depend on the quality of the surrogate model. • Poor capability for parallelisation.
Hyperband	• Efficiently allocates resources to promising configurations. • Successive halving for effective resource utilisation. • Good capability for parallelisation	• Can eliminate slow converging, high performance configurations.
BOHB	• Efficiently allocates resources to promising configurations. • Adaptive exploration of the hyperparameter space. • Good capability for parallelisation	• Computational complexity may be higher. • Can eliminate slow converging, high performance configurations.
Genetic Algorithm	• Efficient with all kinds of hyperparameters. • Can find diverse sets of hyperparameter configurations.	• Computationally expensive with large populations. • Parameter sensitivity and the need for careful tuning.
Particle Swarm Optimisation	• Efficient with all kinds of hyperparameters. • Suitable for parallelisation	• Tendency to converge to local optima. • Parameter sensitivity and the need for careful tuning.
Learning curve prediction	• Early stopping for poorly performing configurations.	• Does not generalise well over different datasets. • Unseen configurations cannot be predicted. • Computationally inefficient • Prediction accuracy can be difficult to achieve

On the other hand, bandit-based approaches such as HB and BOHB exhibit a robust parallelisation capability by training many configurations simultaneously in the initial stages. They proceed with the best-performing configurations, reducing computational time spent on poor performing configurations. However, this may eliminate configurations with slower convergence rates that may eventually achieve superior performance.

Evolutionary-based approaches have seen significant advancements, but their reliance on hyperparameters make them less adaptive to new dataset sizes and types. Their inherent limitations with evolving data availability must be resolved if they are to be implemented in a real industrial environment.

3.1.2 Inspiration and Ambitions: Learning Curve Prediction for Efficient HPO

It is commonly understood that there is a correlation between the hyperparameters of ANNs and the learning curves developed during training, with the hyperparameter choice impacting the performance of the ANN configuration throughout the training run. All approaches of HPO monitor some aspect of multiple trained configurations of hyperparameters to determine which configuration is performing better.

When it comes to learning curve prediction, challenges persist in accurately predicting learning curves for unseen configurations. Traditional approaches leveraging meta-learning, focus on predicting learning curves based on historical data from previous datasets can be beneficial but they often suffer from severe inaccuracies when applied to new scenarios due to their dependence on the specific characteristics of prior datasets. Furthermore, these models can be computationally intensive, requiring extensive training time and optimisation of their own hyperparameters(Choi et al., 2018).

In light of these challenges, it was hypothesised in this study that a more effective approach could be developed by treating the problem of learning curve prediction as a machine learning task. This method focuses on training a model using a subset of learning curves derived from hyperparameter configurations within the same dataset. By doing so, the proposed approach aims to predict the performance of learning curves without the need for extensive prior training on multiple datasets.

The anticipated improvements of this new method over existing algorithms include:

1. **Reduced Computational Cost**: By leveraging a smaller subset of configurations for training, the model can achieve faster convergence, minimising the resources required for learning curve prediction. This efficiency allows for more rapid iterations in the HPO process.

2. **Increased Accuracy**: The proposed method seeks to establish a more accurate prediction model for configurations by directly modelling the relationships between hyperparameters and their corresponding learning curves for the same dataset. This contrasts with traditional methods that may misrepresent these relationships due to their reliance on past data.

3. **Enhanced Adaptability**: The new method is designed to adapt to varying characteristics of the dataset, thereby improving its generalisation capabilities. This adaptability allows for more robust performance across different datasets without necessitating extensive retraining.

This serves as the inspiration for the approach crafted during this research. The learning curve of an ANN configuration offers a wealth of data, surpassing the approach of recording final performance sequentially for HPO, as seen in sequential-based approaches, or the performance at various intervals, as observed in bandit-based methodologies. At each epoch of the training cycle of an ANN, a new variation in performance against the hyperparameter configuration emerges. To effectively harness this wealth of data for efficient HPO, an approach for learning curve prediction is essential. The primary objective of this study is to develop an HPO approach with the following key characteristics:

- **Create a sample set of configuration learning curves:** To train another ML model on the relationship between ANN hyperparameters and the learning curve of all other configurations in the search space.
- **Capability for parallelisation**: The sample set of learning curves will be prepared as a single training set for the ML model to learn from, allowing for a capability for parallelisation. The ML learned correlation between the hyperparameters and learning curves will guide HPO, rather than observed performance of individual ANN configurations.

- **Adaptable for varying datasets:** The approach must be adaptable to varying datasets with minimal adjustment of hyperparameters.
- **Compete in Efficiency and Performance:** The approach must be able to compete with existing HPO methods for computational efficiency and optimisation performance.

3.1.3 Novelty and Contribution of this Research

The most significant contribution from this study comes in the form of a mechanism for the HPO of Multi-Layer Perceptron (MLP) neural networks, the SEquential LEarning Curve Training (SELECT) method. The SELECT HPO method incorporates learning curve prediction to determine the trends in performance of all network architectures in a chosen search space. This is achieved with only 6% of the ANN architectures in the chosen experiments and a one-dimensional CGRNN for prediction. This helps to consistently find better performing neural network architectures with shorter computational time than RS, Bayesian Optimisation with Gaussian Process (GPBO), TPE, and HB in the experiments.

A key novelty of the SELECT mechanism is a method to convert neural network learning curve data from multiple network architectures into a sequenced and windowed dataset. This is used for training the CGRNN to predict the learning curves of all network architectures in the search space together with the use of a single prediction window, meaning the learning curves for all **UNSEEN** network architectures are predicted without any training. The predicted final performance of the learning curves is then ranked and only the top predicted performers are trialled for a result. This approach creates an efficient manner for HPO with learning curve prediction, overcoming challenges such as the drastic variation in learning curve shapes for different hyperparameter configurations, or the additional tuning parameters for the learning curve prediction model, highlighted in previous research (Choi et al., 2018).

3.1.4 Structure of the HPO Development

Before discussing the proposed SELECT HPO mechanism, its background theory will be explained and followed by an overview, explaining the individual stages of the developed algorithm. Finally, the experimental setup and validation results of the proposed method will be presented.

3.2 Background Theory

To establish a foundation understanding of the contributing factors related to the proposed method, an explanation of ANNs, HPO in MLPs, learning curves for ANNs and the CGRNN will be given. Then the theory related to the creation of a "windowed" dataset (Moroney, 2020) for sequential prediction will be explained, followed by the challenges of using a windowed dataset for learning curve prediction in HPO.

3.2.1 Artificial Neural Network

The ANN is a type of supervised machine learning algorithm which utilises interconnected nodes, or neurons, in a layered structure which simulates the learning process of the human brain (Wang, 2003).

The most popular architecture for ANN is the MLP, shown in Figure 3-1. This architecture has an input layer for introducing the dataset to the model, requiring a neuron for each input variable. This is followed by several hidden layers, each with several neurons, before reaching the output layer. The neurons have activation functions for converting the accumulated, weighted inputs into output values. The output layer is the prediction made by the model; in this study, the neural network architecture is used for regression so, the output layer is a single neuron which would contain the predicted value. This architecture functions with the use of back-propagation (Chiang et al., 1996) which begins with a forward pass through the network with randomly initialised weights in each neuron where the total loss in the final prediction is measured. This loss is the difference between the prediction and the actual value; the actual value being taken from a training set of the data. From this loss, an optimisation algorithm such as gradient descent is used to evaluate the gradient of the weights of each neuron through a backwards pass through the model. The weights will then be adjusted with a learning rate to reduce the difference between the prediction and the actual value. This full cycle is known as an epoch. Repeating this process multiple times, or for many epochs, can improve the prediction accuracy over time.

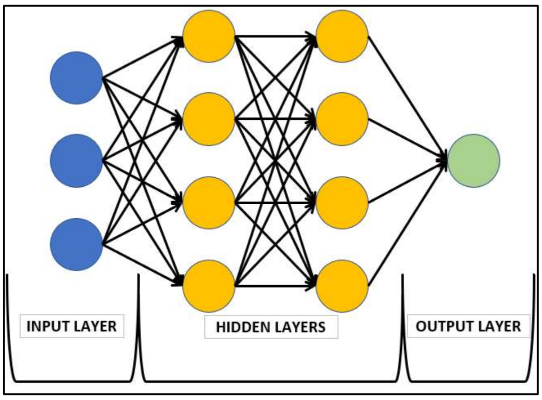


Figure 3-1 The multilayer perceptron neural network architecture.

3.2.2 HPO in a Multilayer Perceptron

HPO is the process of choosing the best selection of hyperparameters to optimise model performance (Hutter et al., 2019). In the case of the MLP ANN, the key hyperparameters are the number of hidden layers, the number of neurons per hidden layer, the learning rate, the activation function, the optimiser, the loss function, and the number of epochs (Zaccone and Karim, 2018). When optimising these hyperparameters, the key is to search over all possible variations to find the best configuration that can help to achieve the best model performance. Given a substantial range of configurations, this process would take a significant amount of computation time that is not always desirable. Hence, several efficient HPO methods have been developed over the years.

3.2.3 Learning Curve for Neural Networks

A learning curve for training a single ANN configuration is the measurement of a performance metric at each epoch of training the model, plotted for all epochs while training. As the ANN trains, the metric will converge towards an optimum. In this study, the performance metric is defined as the loss value showing the difference between the predicted results for the instances in a training set and their actual results. The lower the loss, the better the performance. The training set is the dataset used to optimise the performance of a neural network. A validation set is used to determine if the optimised model is overfitting the training set or not. The validation loss is the same metric as the loss value but on the validation set. Figure 3-2 presents an example of a neural network learning curve. It shows that, after 12 epochs, the validation loss begins to increase, and the neural network in this case is beginning to overfit the training set after 12 epochs.

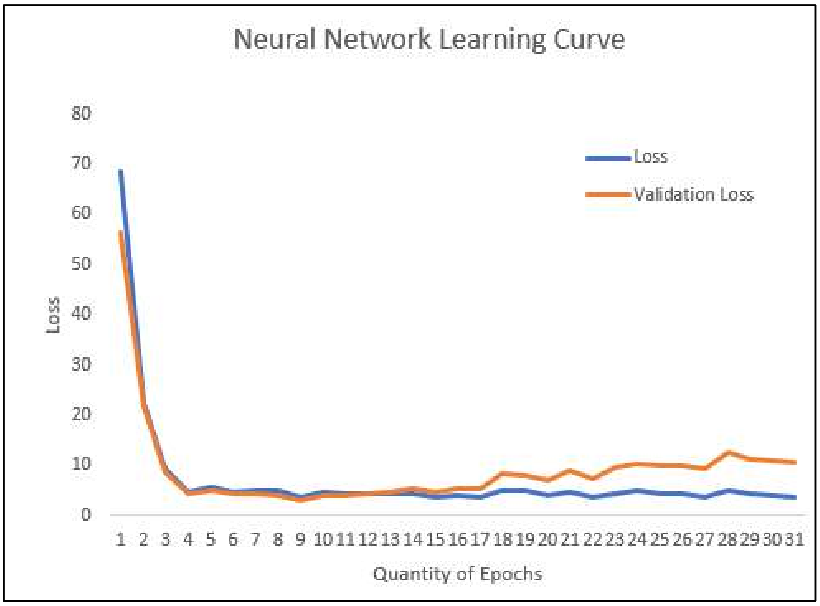


Figure 3-2 A neural network learning curve with loss and validation loss against the number of epochs.

3.2.4 Convolutional Gated Recurrent Neural Network (CGRNN)

The CGRNN is a combination of two distinct types of ANN: a 1-dimensional convolutional neural network(1DCNN) and a type of recurrent neural network known as a gated recurrent unit (GRU).

One Directional Convolutional Neural Network

A convolutional neural network (CNN) is an ANN which is inspired by nature and made up of a similar basic structure of the mammalian visual cortex. It uses convolutions to detect the relationship between features in data. This has shown to be particularly useful for applications such as image classification, object tracking, and text detection and recognition(Zhao et al., 2017, Xie et al., 2020, Kazmi et al., 2021, Gu et al., 2018, Kim et al., 2019). A 1DCNN is a type of CNN which has low computational expense and successful applications in waveform recognition, such as time-series prediction and signal identification*(Hussain et al., 2020, Li et al., 2022, Li et al., 2019)*.

Gated Recurrent Unit

A GRU is a type of recurrent neural network (RNN) which can predict variable length sequences, with hidden state activation for each stage in a sequence relying on the previous stage. RNNs are useful for predicting the steps forward in sequential data, such as time-series data. One limitation of RNNs, however, is that its performance decreases significantly with long-sequence prediction*(Bengio et al., 1994)*. Other RNN architectures were later introduced which helped to overcome this performance issue, such as the long short-term memory (LSTM) unit *(Hochreiter and Schmidhuber, 1997)*, and more recently the GRU was proposed*(Chung et al., 2014)*, which has a simpler architecture with less computation cost. The applications for both of the LSTM and GRU RNNs have focused on sequential data from time-series prediction to natural language processing*(Irie et al., 2016, Kwak and Lim, 2021)*.

Hybrid Methods

There are recent examples of better performing hybrid sequential prediction models, such as *(Wang et al., 2021)* who proposed a 1DCNN-LSTM approach to predict traffic flow, resulting in faster convergence and higher accuracy than the individual sequential models. A 1DCNN-GRU approach was also used by *(Lin and Nuha, 2022)* for the same application with similar improvements in performance over the individual architectures. *(Kanwal et al., 2022)* also produced a hybrid model combining the 1DCNN and the LSTM architectures for stock price prediction, also showing improved accuracy and converge. There is unmistakable evidence of improvement in sequential prediction performance with the hybrid model of 1DCNN and the LSTM and GRU architectures.

3.2.5 Sequential Prediction and the Windowed Dataset

Figure 3-3 shows four key steps of the sequential training process for a using the Fibonacci sequence as an example. To clarify, the Fibonacci sequence (FS) is formed by adding the previous two numbers together to form the next number in the sequence at every step, starting with 1 and 1, leading to the next number, 2. To explain the creation of the windowed dataset, the FS moving along from 1 through to 21 is used as shown at the top of Figure 8. Starting from step 1, if the supplied dataset contains the first 6 numbers (1 to 8) of the FS, the last two numbers will be 13(5+8) and 21(8+13). In step 2, the dataset is converted into a windowed dataset to enable sequential prediction, i.e., predicting the last two numbers of the FS.

The windowed dataset contains a set length of window (as inputs) and a set length of prediction (as output). Using FS as an example, if the window length and prediction length are set to three and one respectively, an instance will be created by covering three numbers in the window and one number in the prediction. For each instance (or row) in the windowed dataset, the window and prediction will move along the sequence in a single step. Hence, the windowed dataset moves three times for the training set in the given example along the FS, forming three instances.

In step 3, the model is trained with the three instances of the windowed dataset. Step 4 uses the trained and a window of 3,5,8 to predict the next step in the sequence which is 13. Similarly, a window of 5,8,13 would predict 21 in the sequence. It is important to note that each value and the order of the values in the windowed dataset contribute to the sequential prediction.

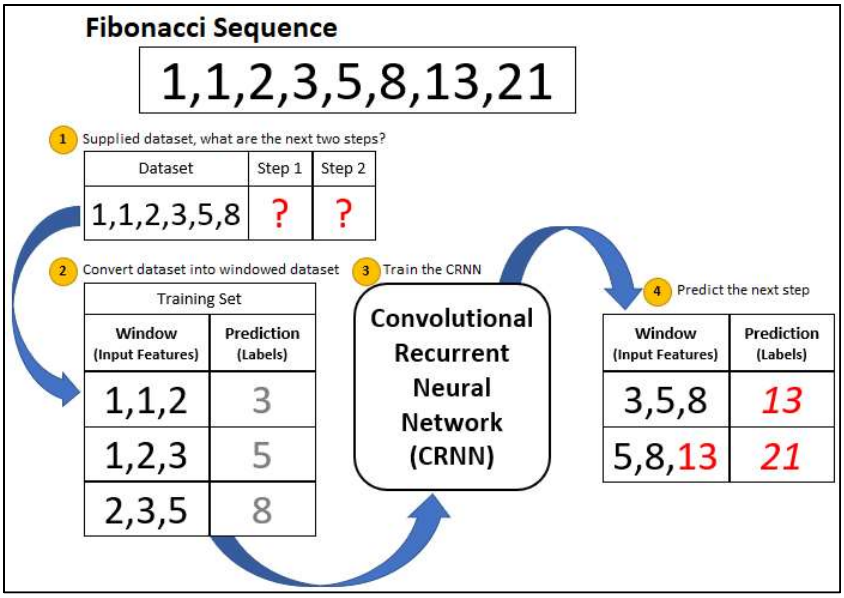


Figure 3-3 Four steps of creating windowed dataset using the Fibonacci sequence.

3.2.6 Challenges for Learning Curve Prediction

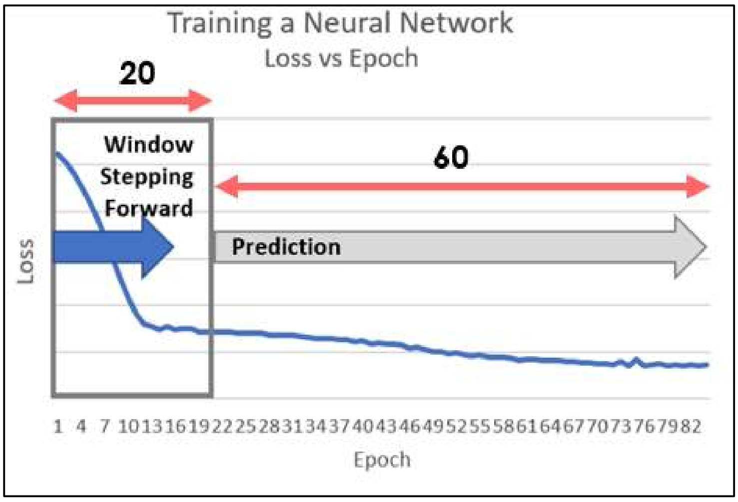


Figure 3-4 A neural network training cycle with the training window and prediction area.

Challenges related to learning curve prediction for HPO come from the variability of the learning curves for different datasets and hyperparameter configurations. When considering windowed datasets for long sequence predictions, the nature of learning curve prediction is especially challenging. Figure 3-4 shows an example of the learning curve of a neural network training cycle. It represents the window covering the initial 20 epochs of the training cycle and the prediction for a further 60 epochs. Given the variations in dataset and hyperparameter configuration a different learning curve would be observed inside the window. It means that a different window would be required to predict the future learning curve of each hyperparameter configuration, meaning that after training on the dataset, each configuration would need to run for the length of the window before a prediction can be made. Also, the accuracy of the prediction reduces with the increasing length of prediction after the window (Preeti et al., 2022).

In Figure 3-4, if there were 500 hyperparameter configurations in the search space, with these variations in learning curve over the training window, each configuration would need to train for 20 epochs for all 500 configurations resulting in 10,000 epochs for predictions alone. This does not include all the processes involved for training the sequence prediction model. Further to this, 20 epochs is used for the window to predict 60 epochs ahead in this example. Increasing the window size would increase the training before making a prediction and reduce the useful prediction time.

3.3 Developing the SELECT HPO Method

This section outlines the progression of ideas that led to the SELECT method for hyperparameter optimisation. The subsequent sections detail the final SELECT algorithm.

The journey began with the goal of using learning curve prediction to optimise neural network hyperparameters. Initially, the aim was to apply early stopping to eliminate poorly performing learning curves, similar to previous studies. However, during testing, a more significant opportunity emerged: the possibility of predicting **entire unseen learning curves** within the search space.

This process began by iteratively training learning curves across various hyperparameter configurations to observe the effects. It became clear that performance was sensitive to hyperparameter choices. For example, learning curves varied significantly with changes to the number of hidden layers, while adjustments to neuron count had a lesser effect. It was also found that learning rates outside a specific range led to universally poor performance across configurations, with this range varying for each dataset.

Through these observations, the idea of tuning learning rates emerged. By slightly setting the learning rate below the convergence range and then incrementally increasing it based on a percentage drop in the curve, the model could adapt the learning rate across configurations. This approach allowed it to respond to different datasets, enhancing adaptability in tuning.

Attention then turned to the challenges of sequence prediction discussed in Section 3.2.6. To mitigate noise and cumulative error, the concept of blocks was introduced. Each block aggregated 10 epochs, smoothing noise and creating stable data intervals that enhanced learning curve prediction accuracy. By training on blocks rather than individual epochs, the model could efficiently capture long-term trends.

Another challenge was the need for training each configuration to cover the window length before making a prediction. To address this, a synthetic starting step was added, achieved by replicating the first instance of each learning curve. This consistency in the starting block allowed the sequential model to begin all configurations from a standard baseline. This also meant that predictions could be made at the start of each training cycle, rather than requiring the cycle to run for the length of the window first.

With these refinements in place, each learning curve featured an adaptive learning rate, block intervals, and a starting step. Figure 3-5 shows a typical learning curve prepared for sequential prediction.

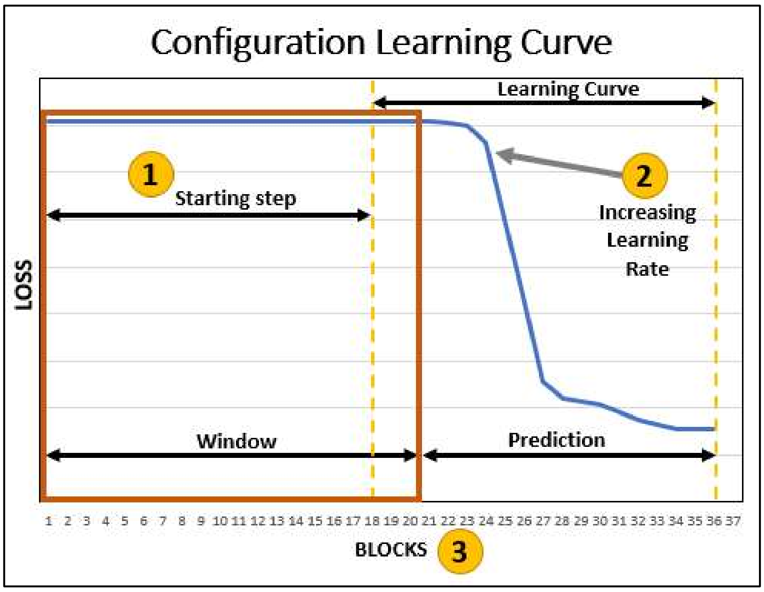


Figure 3-5 An amended neural network learning curve for the sequential prediction model.

Summarising the key points of Figure 3-5:

- A ‘starting step’ created from synthetically attached instances allows for a sizeable training window which will not impact the available prediction length within the learning curve.
- The use of an increasing learning rate, dependent on the learning curve of each hyperparameter configuration, for training all hyperparameter configurations so

that all configurations have negligible difference within the initial window length and converge in an epoch limit.

- The conversion of the learning curve data into 'blocks' of information taken over set epoch intervals reducing sequence length and noise, capturing learning curve trends effectively.

By formatting the learning curves of all trained hyperparameter configurations, as shown in Figure 3-5, and attaching them in sequence, the sequential prediction model could effectively cycle through this structured training set. This setup was intended to enable the sequential prediction model to identify and halt poorly performing learning curves early in the training process of each cycle.

With the code in place, the idea arose to attempt prediction of complete learning curves of unseen configurations using only the initial window of the training cycle, shown in Figure 3-5. This proved successful, allowing the model to extrapolate learning curves across the search space by identifying hyperparameter relationships within a subset of learning curves. This innovation became the foundation for the SEquential LEarning Curve Training (SELECT) method for hyperparameter optimisation.

The SELECT method follows the stages listed below for HPO. Refer to Figure 3-6 for clarification.

- Stage 1: The learning rate is tuned to the dataset learning rate range and parameters for the adaptive learning rate for each hyperparameter configuration are defined.
- Stage 2: A subset of the hyperparameter configurations in the search space are trained, each learning curve is converted into the format shown in Figure 3-5.
- Stage 3: These are then joined in sequence and converted into a windowed dataset for the sequent ML model.
- Stage 4: The sequence ML model (CGRNN) is then trained to learn the relationship between the hyperparameters and the learning curve sequences.
- Stage 5: Using the relationship information, the learning curves for all configurations in the search space would then be predicted with a single instance of the windowed dataset.
- Stage 6: The best predicted performances could then be trialled to select the best performing configuration for optimum performance.

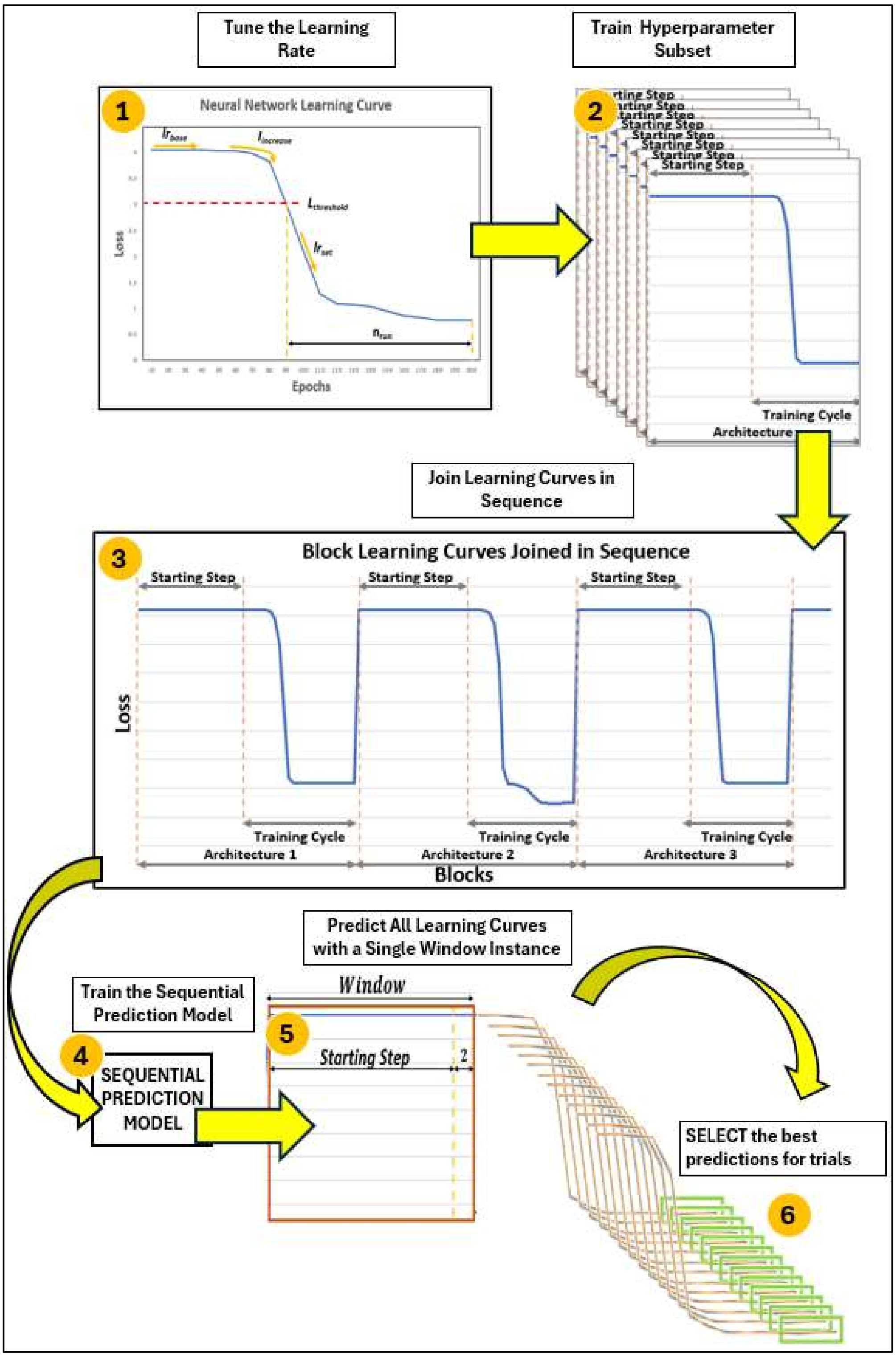


Figure 3-6 Stages of the SELECT algorithm

3.4 SELECT HPO Methodology Abbreviations

Abbreviation	Description
lr_{base}	base learning rate for negligible difference, set for each configuration
$l_{increase}$	Increase rate of the learning rate
L_{start}	Initial loss at the start of training
$L_{threshold}$	Loss threshold for setting the learning rate of each configuration
$L_{current}$	The current recorded loss at each stage of training
lr_{set}	The learning rate achieved at the loss threshold
n_{run}	Set epoch limit after the learning rate has been set
L_{scale}	Loss threshold when setting the base learning rate
D_{scale}	Percentage drop from starting loss to loss threshold when setting the base learning rate
lr_{scale}	Initial learning rate while setting the base learning rate
Sc	Scaling factor for increasing the learning rate exponentially while setting the base learning rate
L_{set}	Loss threshold when setting the increase rate for the learning rate
D_{set}	Percentage drop from starting loss to loss threshold when setting the increase rate for the learning rate
$n_{increase}$	The number of epochs to the loss threshold when setting the increase rate for the learning rate
f	Adjustment factor for setting the threshold loss while tuning the learning rate
$tune_{quantity}$	The number of iterations while setting the loss threshold
$grad_{limit}$	The range of acceptable gradient of convergence while setting the loss threshold
grad	The final gradient of the learning curve in a training cycle
CGRNN	Convolutional gated recurrent neural network
$train_{seq}$	The sequential training set for training the CGRNN
block	A collection of information related to learning curve data over a set interval
$block_{size}$	The number of epochs in each block of learning curve information
X	Input range for sequential prediction
Y	Output range for sequential prediction
W	The length of the window in blocks
P	The prediction length in blocks
S	Starting step length in blocks
C	Training cycle length in blocks
$trial_{quantity}$	Quantity of trialled configurations from the ranked learning curve predictions

3.5 Structure of the Proposed Approach

Incorporating the solution from section 3.3.2, the six key stages of the SELECT method are revised into the pipeline shown in Figure 3-7 with data transfer highlighted between different stages. Each dataset is processed for categorical variables and feature scaling, and split into training, validation, and test sets at stage 1, leading onto the tuning of the learning rate. The purpose of stage 2 is to allow the sequential prediction model to adapt to all datasets and hyperparameter configurations by setting a small learning rate at the start of training which adjusts to each configuration, dependant on the performance of the learning curve. This also makes every trained configuration begin with negligible difference, contributing to the ability of the SELECT method to use a single predictive window at stage 6. The learning curves are converted to 'blocks' of data, the starting step is added, and all learning curves are joined in series, creating the novel training set for the CGRRN at stage 3. The learning curve data is feature scaled and converted into a windowed dataset before training the sequential prediction model at stage 4. Predictions are made with the trained sequential prediction model, and the best results are trialled to select the top performing model at stages 5 and 6. The following sub-sections will cover the key points in relation to all stages and their development.

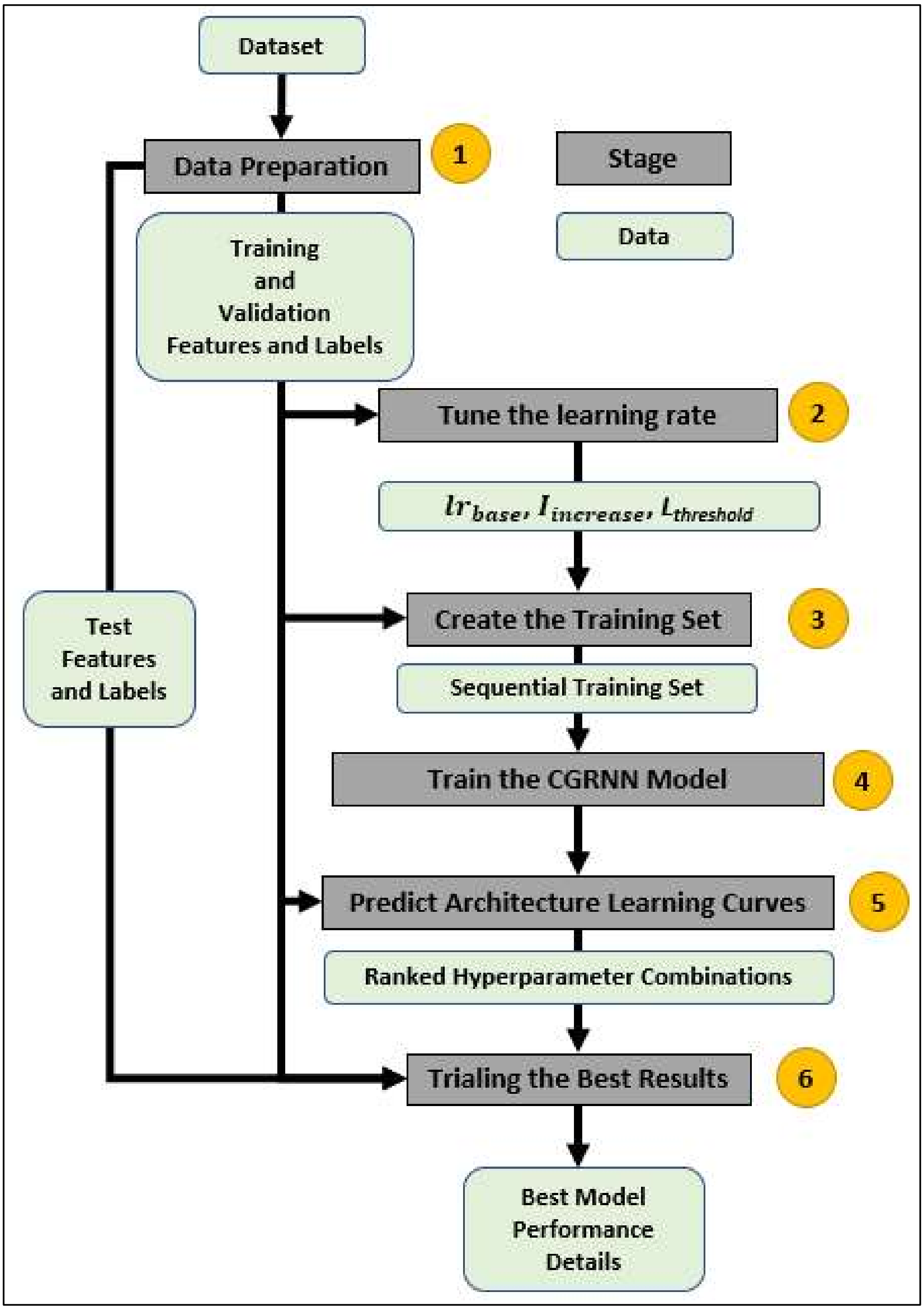


Figure 3-7 The six stages of the proposed HPO approach.

3.5.1 Stage 1: Data Preparation

It is assumed that uploaded datasets have been checked prior to uploading to the algorithm for missing variables or incorrect information so this aspect of preparation is not programmed into the algorithm. The dataset selected for experimentation have all been prepared in advance in this way. The categorical variables are encoded numerically, and the prediction variable and input variables are separated as the labels and features of the data respectively. The application of this algorithm is for regression solutions so there is a single, continuous label for each instance in all datasets in the study. The dataset is then split into the training set and the test set with a split ratio of 20% for the test set. From the training set a 20% subset is created for the validation dataset. The features for all of the split datasets will be scaled using the Scikit-learn MinMaxScaler module (Bisong and Bisong, 2019).

3.5.2 Stage 2: Tuning of the Learning Rate

Rather than treating the learning rate as a set hyperparameter to tune, (Smith, 2017) created an approach for a learning rate optimiser which adjusted the learning rate within set limits based on the performance of the learning curve to assist in the training process by allowing flexibility in learning rate. This method was the loose inspiration for the idea to make the learning rate dependent on the performance of the learning curve. Although the proposed approach applied the same logic in a unique way. Instead of assisting the learning rate converge within set limits, this approach sets a bottom limit for the learning rate, which can the adjust individually for each hyperparameter configuration. For each dataset, it was determined through experimentation that the ideal learning rates of different architectures fitted within a certain range of each other, meaning that if the learning rate could be set to a value slightly below the bottom of the range, dependant on a percentage drop, then it could be increased from this point each time a training cycle began with a new configuration and stop increasing at a relevant change in the learning curve for convergence around the epoch limit. This creates a method of tuning the learning rate to the dataset, and then each configuration depending on each learning curve.

The purpose of tuning the learning rate is to fit the learning curve for every neural network architecture into a set epoch limit to train and predict with the sequential prediction model. For a single window to be used for prediction, each configuration needs to begin at a base learning rate (lr_{base}) with negligible variation in loss and increase at a defined increase rate ($I_{increase}$) until the loss curve hits a threshold loss ($L_{threshold}$). After $L_{threshold}$ is reached, the learning curve will continue for a set epoch limit (n_{run}) with the final learning rate (lr_{set}). Referring to the example of the learning curve in Figure 3-8, the loss has negligible variation in the first thirty epochs with the base learning rate, lr_{base}, before the learning rate increases after each epoch with $I_{increase}$, leading to an increasing reduction rate of loss. As the loss reaches $L_{threshold}$, the defined threshold loss limit at 90 epochs, the learning rate stops increasing. The set learning rate at this point, lr_{set}, is used as the basis for the rest of the learning curve until the loss approaches a minimum after a further 110 epochs, n_{run}. This approach allows the learning rate to adapt to each model configuration and the start of each training cycle to be identical for all configurations. The method for determining each of the parameters in this approach is automatically carried out at the start of the optimisation process of this HPO method. These automatic processes are defined in the following sub-sections.

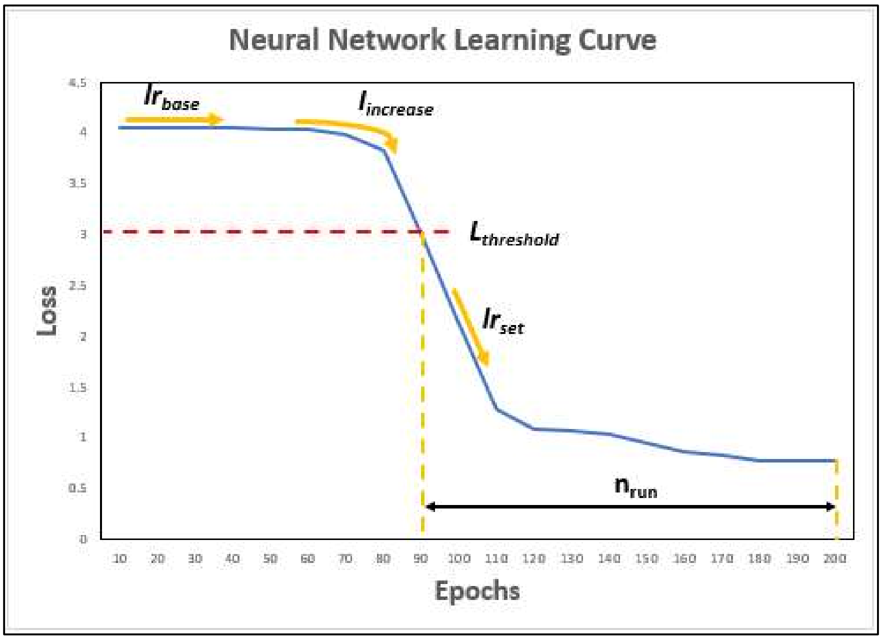


Figure 3-8 The highlighted parameters for tuning the learning rate.

lr_{base}

The purpose of lr_{base} is to have an initial learning rate for each neural network architecture which produces negligible loss differences between all variations but can then be increased to quickly begin to have an impact on the loss. The difficulty with setting lr_{base} is that the ideal learning rate range varies, depending on the dataset being used and the neural network architecture that is being trained. As the variation in the network loss is the guide for lr_{base}, this would require the setting of lr_{base} to be dependent on the loss for this to function on multiple datasets. Regarding this, the following approach is taken to setting lr_{base}, shown in *Figure 3-9*. This algorithm is implemented with the use of a custom callback during the training process. This tool allows for data to be recorded and logical algorithms to be applied during various stages during the training cycle of each configuration, in this case, at the start of training and after each epoch of training of an initial configuration for setting lr_{base}. A description of the steps involved in this algorithm are provided here.

Setting lr_{base}

On train begin

Record L_{start}

$L_{scale} = L_{start} \times (1\text{-}D_{scale})$

On epoch end

Record lr_{scale}

$lr_{scale} = lr_{scale} \times e^{Sc}$

Record $L_{current}$

if $L_{current} < L_{scale}$:

Stop training

$lr_{base} = lr_{scale}$

Figure 3-9 The pseudocode of the algorithm to set the base learning rate(lr_{base})

1. A neural network with a single hidden layer is selected for tuning the learning rate as it has been shown that increasing the number of hidden layers will make the neural network converge more slowly (Uzair and Jamil, 2020).
2. The loss is measured at the start of training the neural network (L_{start}) and an initial threshold scale loss is calculated (L_{scale}) as a fraction of the initial loss with a percentage drop(D_{scale}). In the process of initial development, a loss percentage drop

of 0.02% or a *D_{scale}* value of 0.0002 was determined to be suitable for a negligible variation between configurations.

3. Starting with an initial learning rate (lr_{scale}) which is significantly low enough to have negligible impact on the loss for any dataset (1-e9), the chosen neural network is trained while the learning rate is increased exponentially after each epoch, using a scaling factor (*Sc*).
4. The loss is measured after each epoch($L_{current}$) and compared to L_{scale}. If $L_{current}$ is less than L_{scale}, the training stops and the final lr_{scale} is defined as lr_{base}.

The initial learning rate, lr_{scale}, of 1e-9 was determined through analysis of the equation for stochastic gradient descent. As shown below:

$$\theta_{new} = \theta_{old} - \nabla J(\theta_{old}) \times lr$$

As the weights (θ) will be updated in each epoch with the subtraction of the learning rate (*lr*) multiplied by the gradient of the loss function ($\nabla J(\theta_{old})$) from the old parameters (θ_{old}), defining an *lr* value less than 1e-6 can be considered to produce an extremely small variation in each step, but as it is important for the process to start with a negligible loss for all datasets, an additional factor of 1e-3 was added to this scale, resulting in 1e-9 as the starting learning rate. The *S value was set to 0.1,* allowing the learning rate to increase from 1e-9 to 0.5 over the space of 200 epochs. A length of 200 epochs allows for each dataset to reach L_{scale} without overshooting the learning range for the configurations that were tested. This was set through the process of trial and error at the initial stages of development and has been suitable for all experiments moving forward.

The choice to use 200 epoch limit stemmed from observing the number of epochs for the learning rate to increase to a level so that significant loss is detected. The was a balance set through observation between ensuring that the step increased slowly enough to not overshoot the lower acceptable learning rate limit while also limiting the required computational time required for training.

The approach taken in *Figure 3-9* sets the starting learning rate, lr_{base}, for the training of all neural network architectures moving forward. This is set at the beginning of each optimisation process in the proposed approach for a new dataset to set the bottom limit of the range of learning rates to suit all architectures. As the optimum learning rate will vary depending on each configuration, the learning rate is gradually increased until there is significant loss, accordingly, using the rate of increase, $I_{increase}$. The creation of this variable is discussed next.

$I_{increase}$

The purpose of $I_{increase}$ is to raise the learning rate while training a neural network architecture from lr_{base} to a point where the loss reaches a predefined loss threshold, L_{set}, within a set length of epochs (n_{set}). This increase must also occur after the number of epochs for 2 blocks to pass so that there is minimal difference between all configurations within the length of the window for the sequential prediction model. This means that for the number of epochs of 2 block intervals, the learning rate will be set at lr_{base} allowing for negligible variation for all configurations during this portion of the training cycle. After this stage the learning rate will be made to increase until L_{set} is achieved. $I_{increase}$ will be different for each dataset so it will need to be determined through analysis of an initial trained neural network and then amended to suit the length of n_{set}. The approach for this is described below, expressed in *Figure 3-10.*

Setting $I_{Increase}$

On train begin

Record L_{start}

$L_{set} = L_{start} \times (1 - D_{set})$

$lr_{set} = lr_{base}$

On epoch end

Record lr_{set}

*if epoch>(2*Block):*

$lr_{set} = lr_{set} \times I_{start}$

Record $L_{current}$

if $L_{current} < L_{set}$:

Stop training

epoch $= n_{increase}$

On train end

$I_{increase} = I_{start}^{\frac{n_{increase}}{n_{set}}}$

Figure 3-10 The pseudocode of the algorithm to set the increase rate of the learning rate ($I_{increase}$)

1. The same neural network used for setting lr_{base} is trained.
2. When training begins, the loss is measured at the start of training the neural network (L_{start}) and a threshold loss for setting is calculated $(L_{threshold})$ as a fraction of the initial loss with a percentage drop(D_{set}). lr_{base} must be kept the same for all future learning curves so a new variable, lr_{set}, is used for the increasing learning rate.
3. On the end of each epoch, after the number of epochs required for 2 blocks of data has passed, lr_{set} is increased by a predefined percentage increase I_{start}.

4. The loss is measured after each epoch($L_{current}$) and compared to L_{set}. If $L_{current}$ is less than L_{set}, the training stops, and the final epoch quantity is defined as $n_{increase}$.
5. After training stops, $I_{increase}$ is calculated using I_{start}, $n_{increase}$ and n_{set}. The proof for the equation is provided in Equation 1.

$$lr_{set\,final} = lr_{base} \times I_{start}^{n_{increase}}$$
$$lr_{base} \times I_{increase}^{n_{set}} = lr_{base} \times I_{start}^{n_{increase}}$$
$$I_{increase}^{n_{set}} = I_{start}^{n_{increase}}$$
$$\ln(I_{increase}^{n_{set}}) = \ln(I_{start}^{n_{increase}})$$
$$n_{set} \times \ln(I_{increase}) = n_{increase} \times \ln(I_{start})$$
$$\ln(I_{increase}) = \frac{n_{increase}}{n_{set}} \times \ln(I_{start})$$
$$\ln(I_{increase}) = \ln(I_{start}^{\frac{n_{increase}}{n_{set}}})$$
$$\boldsymbol{I_{increase} = I_{start}^{\frac{n_{increase}}{n_{set}}}}$$

(1)

I_{start} was set through trial and error to be 1.02, this allows for a slow increase and a larger number of epochs to achieve L_{set} compared to n_{set}. The value of n_{set} for all experiments was set to 80 epochs. This means that the desired number of epochs for setting lr_{set} is 80 epochs. With a slow increase rate in the first run, $I_{increase}$ can be set by reducing the number of epochs from $n_{increase}$ to n_{set} for each dataset using the above equation. Implementing a gradual increase in the learning rate offers advantages in terms of training stability. Swift and abrupt increments in the learning rate could result in instability or overshooting during the optimisation process. By adopting a more measured approach, making gradual adjustments to the learning rate, the model is better poised to converge reliably. Additionally, allowing a larger number of epochs to reach L_{set} compared to n_{set} creates an extended training period during which the learning rate is adapted. This approach is conducive to a thorough fine-tuning of the model's parameters, enhancing its overall performance. Even if the learning rate is set at a point where it reaches L_{set} after a predefined number of epochs, n_{set}, this still does not guarantee that the model will converge to a minimum after a predefined set number of epochs after this point, n_{run}. To achieve this, L_{set} needs to be adjusted to a threshold, $L_{threshold}$, which allows for the convergence to fit around n_{run}. The approach to achieve this is discussed next.

$L_{threshold}$

$L_{threshold}$ is a percentage of the initial loss in the learning curve which will stop the increasing learning rate. The purpose of $L_{threshold}$ is to stop $I_{increase}$ at a value of lr_{set} which will allow the learning curve to converge within the set epoch limit n_{run}. This is achieved by iteratively training a single layer neural network and incrementally reducing $L_{threshold}$ with an adjustment factor (*f)* based on the final gradient (*grad*) recorded in n_{run} .

The purpose of tuning the learning rate in this approach is to fit the learning curve for every configuration into a set epoch frame $(n_{increase} + n_{run})$ for the windowed sequential prediction model to train from. Using this trained sequential prediction model, the predictions can be ranked for better performing learning curves, rather than finding the best result at this stage. This means that with slow convergence datasets it is only needed to get as close to convergence as possible. In order to achieve this, the process will repeatedly adjust $L_{threshold}$ for a set quantity of iterations $(tune_{quantity})$, if there is no improvement, the acceptable range of gradient ($grad_{limit}$) will be increased. As $L_{threshold}$ will be changed, $I_{increase}$ will be altered so $n_{increase}$ is kept at n_{set}. This is shown in Figure 3-11 followed by a description of the steps taken.

Setting $L_{threshold}$
Train neural network for $n_{set} + n_{run}$
Record $I_{increase}$
Record $grad$
while $grad < grad_{limit}$:
tune $_{quantity}$ = *tune* $_{quantity}$ *+1*
if tune $_{quantity}$ *>tune* $_{quantity\ limit}$
$increase\ grad_{limit}$
$L_{threshold} = L_{threshold} \times f$
Train neural network for $n_{set} + n_{run}$
Record $grad$
Record $n_{increase}$
$I_{increase} = I_{increase}^{\frac{n_{increase}}{n_{set}}}$

Figure 3-11 The pseudocode of the algorithm to set the threshold loss for setting the learning rate of each configuration ($L_{threshold}$).

1. Train the single layer neural network and record $I_{increase}$ and *grad*.
2. While *grad* is outside of $grad_{limit}$, alter the $L_{threshold}$ with *f*.
3. Train the single layer neural network and record *grad* and $n_{increase}$.
4. Amend $I_{increase}$ to achieve lr_{set} for n_{set} number of epochs.
5. If $grad_{limit}$ is not achieved, repeat until $tune_{quantity\ limit}$ is reached, then increase $grad_{limit}$

6. Repeat steps 2-5 until *grad* achieves $grad_{limit}$

The initial L_{set} is defined as 80% of L_{start} and the adjustment factor has been set to 0.95, meaning that as there is continued iterations with a learning curve not converging, the new $L_{threshold}$ will decrease by 5% each iteration, allowing for a faster convergence each time. These variables were set during development through trial and error.

While tuning the learning rate, multiple parameters were explored through trial and error. It's important to note that the datasets used for development differed from those employed in the validation experiments. The setting of these variables was conducted independently of all validation experiments. The lr_{base}, $l_{increase}$ and $L_{threshold}$ values are automatically determined for each new dataset using the described method, the next step is to generate the sequential training set out of the learning curves.

3.5.3 Stage 3: Creation of the Sequential Training Set

The purpose of the sequential training set ($train_{seq}$) is to provide the sequential prediction model with sufficient information so that the correlation between the learning curve of a neural network, and the network architecture and learning rate can be established, while simultaneously making it possible for a single training window to be used for prediction. The creation of the training set has four stages: (i) running several neural network architectures spanning the hyperparameter search space, (ii) converting the data into blocks of information for long sequence prediction, (iii) attaching the starting step and (iv) finally joining the learning curves from all the trained neural network architectures in series. Each of these stages will be discussed in this section.

Hyperparameter Search Space and Selected Architectures

In this study, the HPO focuses on the learning rate, the number of hidden layers and the number of neurons per hidden layer which are most suited for optimum performance of a neural network for regression applications. With the learning rate dependent on the loss of each neural network architecture configuration, it is no longer an independent hyperparameter and will set automatically and individually to each configuration. Additionally, as the learning rate has been tuned to achieved convergence around the n_{run} limit, the epoch limit is also predefined. This results in the number of neurons per hidden layer and the number of hidden layers being the focus of this HPO approach. For this study, the chosen hidden layer range is between 1-5 hidden layers, while the range of neurons per hidden layer is 1-100. To ensure the effectiveness of the approach, the selected neuron quantities and layers must span the range of each hyperparameter. Hence, the selected architectures are examined through 6 sets of neuron quantities over 5 hidden layers. This means that 6 sets of neuron quantities (1, 20, 40, 60, 80, and 100) per hidden layer are trained for 5 different hidden layers (1,2,3,4 and 5). This results in 30 of the total 500 network architectures being trained, meaning 6% of the potential network architectures are trained for the sequential prediction model to function effectively, incorporating the automatic setting of the learning rate in both $train_{seq}$ and the predictions from the sequential prediction model.

Information Blocks for Long Sequence Prediction

Figure 3-12 illustrates a learning curve of a neural network architecture with 1 hidden layer and 20 neurons incorporating the tuned learning rate method discussed earlier, this is using the 'Behavior of Urban Traffic' dataset(Sassi et al., 2011). The learning curve can be seen to fluctuate rapidly which would limit the ability for the sequential prediction model to interpret the continuous flow of the learning curve, reducing the clarity of the trend in performance. To add to this, a good sequential prediction model would need to foresee up to 180 steps ahead, through this fluctuation, to evaluate the result of the learning curve.

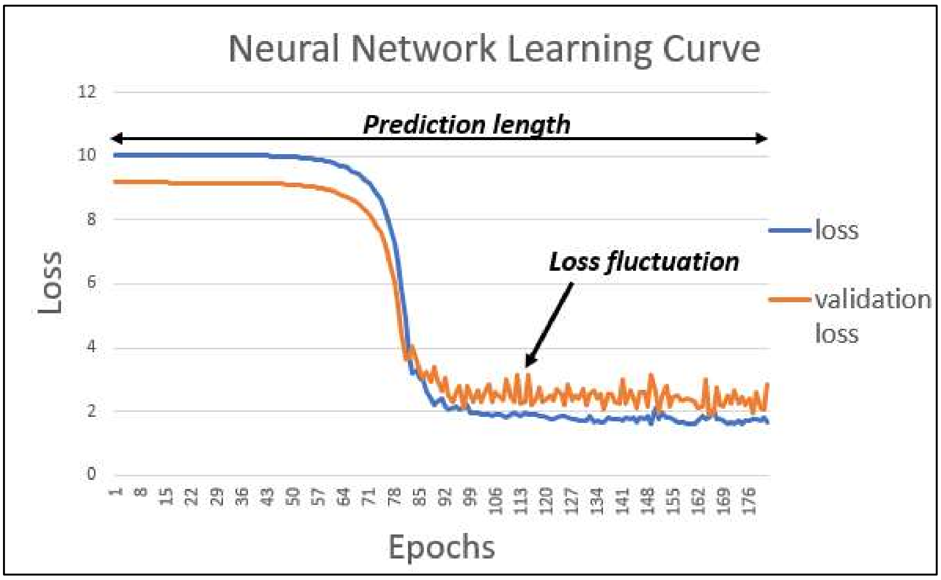


Figure 3-12 The prediction length and loss fluctuation of the neural network learning curve.

To overcome this uncertainty in interpreting the trend and predicting the learning curves, the information in the learning curves is recorded over set intervals, termed as blocks. A block contains the average and range value for each of the loss, the validation loss, and the learning rate over a set interval ($block_{size}$). The equations for the average and range are shown below in Equation *2* and Equation *3* respectively. At the start of each block (*i*), the average and range are calculated before all blocks are appended into a shortened dataset. Table 3-2 shows the learning curve data in Figure 3-12, after being converted into blocks.

$$Average_{i:i+Blo\quad size} = \frac{\sum value}{Block_{size}} \tag{2}$$

$$Range_{i:i+Block_{size}} = maximum\ value - minimum\ value \tag{3}$$

Using $block_{size}$ = 10 epochs, a different learning curve can be produced with the average loss and average validation loss as shown in Figure 3-13. This helps to filter out the fluctuations in loss values and reduces the required length of prediction by a factor of 10, hence it significantly increases the accuracy of predicting the trend of the same learning curve in blocks. Reducing the number of data points from epochs to blocks with the average readings allows for a reduction in the steps forward to be predicted but this also reduces the information that is being used for prediction. This is why the range of values is also recorded. With both the average and range of values taken for each block in parallel, as shown in Table 3-2, the sequential prediction model has an increased number of input features to learn from to make predictions. The average features show the overall trend of the blocks through the learning curve while the range values provide an insight into how the epoch values vary between intervals.

Table 3-2 Learning curve data converted into blocks of information.

Block	Average Loss	Average Valid. Loss	Average Learning Rate	Range Loss	Range Valid. Loss	Range Learning Rate
1	10.05	9.18	3.63E-05	0.00	0.00	0.00E+00
2	10.04	9.18	3.68E-05	0.00	0.00	4.78E-06
3	10.04	9.17	8.22E-05	0.00	0.01	1.00E-04
4	10.03	9.16	2.83E-04	0.02	0.02	3.46E-04
5	10.00	9.13	9.76E-04	0.06	0.06	1.19E-03
6	9.88	9.00	3.36E-03	0.20	0.22	4.11E-03
7	9.46	8.53	1.16E-02	0.76	0.84	1.41E-02
8	7.21	6.15	3.91E-02	5.21	4.38	4.07E-02
9	2.76	3.25	6.06E-02	1.62	1.34	0.00E+00
10	2.05	2.54	6.06E-02	0.31	0.90	0.00E+00
11	1.90	2.57	6.06E-02	0.17	0.84	0.00E+00
12	1.86	2.52	6.06E-02	0.21	0.95	0.00E+00
13	1.78	2.49	6.06E-02	0.19	0.64	0.00E+00
14	1.74	2.46	6.06E-02	0.20	0.96	0.00E+00
15	1.81	2.44	6.06E-02	0.47	1.14	0.00E+00
16	1.71	2.39	6.06E-02	0.35	0.69	0.00E+00
17	1.75	2.33	6.06E-02	0.41	1.08	0.00E+00
18	1.71	2.35	6.06E-02	0.18	0.89	0.00E+00

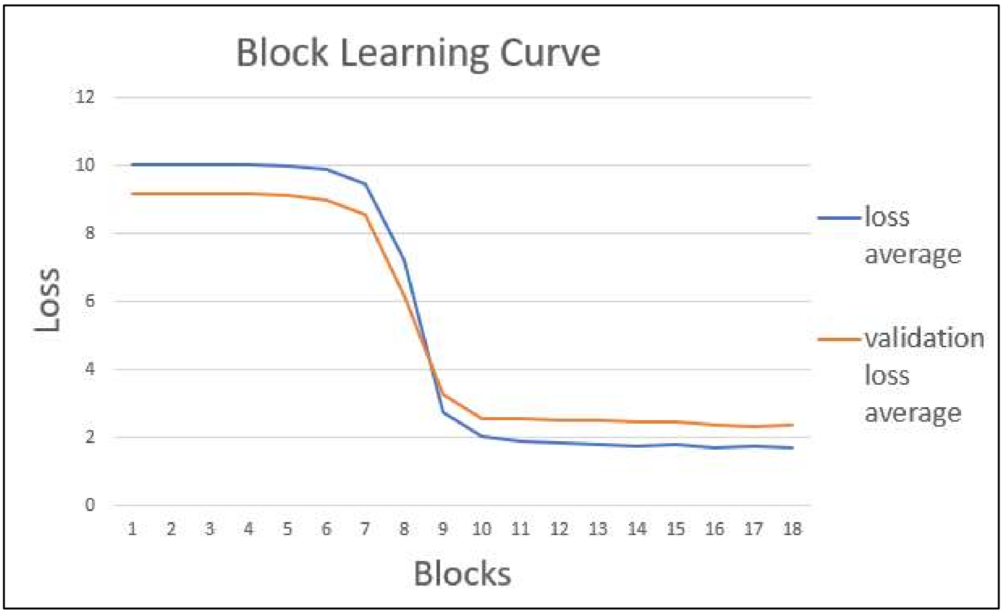


Figure 3-13 The neural network learning curve after being converted into blocks of the average loss.

Starting Step

A windowed dataset must be used to train the sequential prediction model and make predictions. Each window in the dataset requires a set number of blocks in sequence which can be used as a reference (as inputs) for predicting the next blocks in the sequence (as outputs). Before making a prediction, a learning curve must be first created in blocks to cover the window length. With every block being used for the window, the useful prediction length reduces, see Figure 3-14. With a short window length, W1, there is a large prediction length, P1, requiring a narrow window to predict a larger sequence ahead, leading to a model with low accuracy. Increasing the length of the window to W2 would provide further context for the sequential model to learn from, however this would reduce the prediction length to P2, increasing the prediction accuracy, but more blocks are needed for prediction, providing less benefit, and reducing the efficiency of prediction.

To achieve the benefits of having a large window size for prediction context while not impacting the prediction length, a synthetic starting step is added to the block data for each trained configuration. This is done through replicating the first block for every training cycle, see Figure 3-15. The length of the starting step can be increased and decreased as required, depending on the user's selection of window length. It is important for the starting step to be smaller than the window length because replicating the first block means that the data in the starting step is identical. If the window has identical data for all instances, then the sequential prediction model is not able to determine when the starting step begins and ends while training.

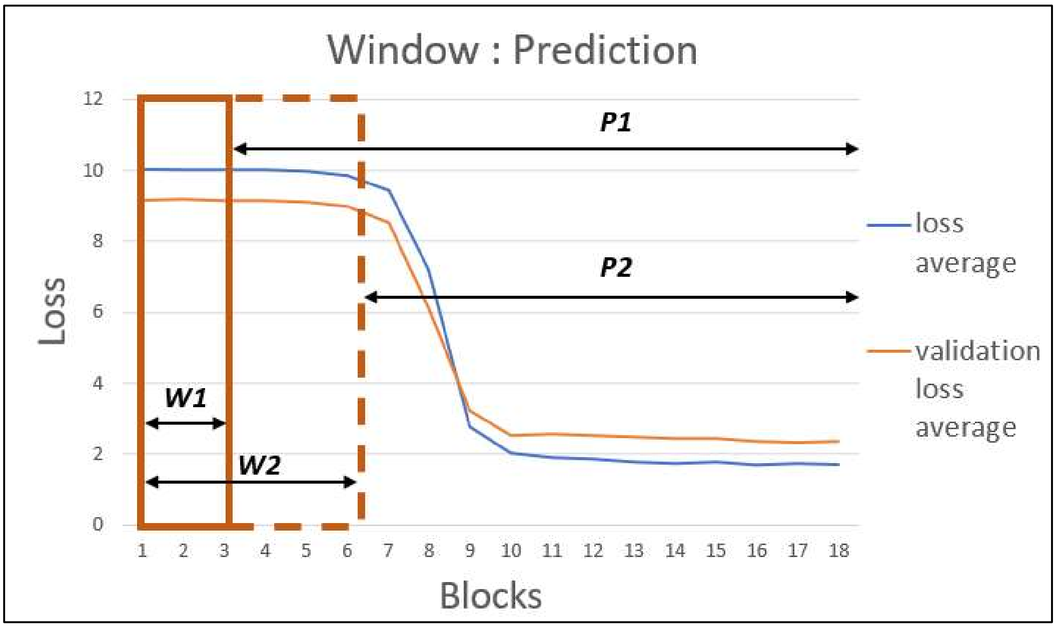


Figure 3-14 The trade-off between the window length and the prediction length in predicting the learning curve.

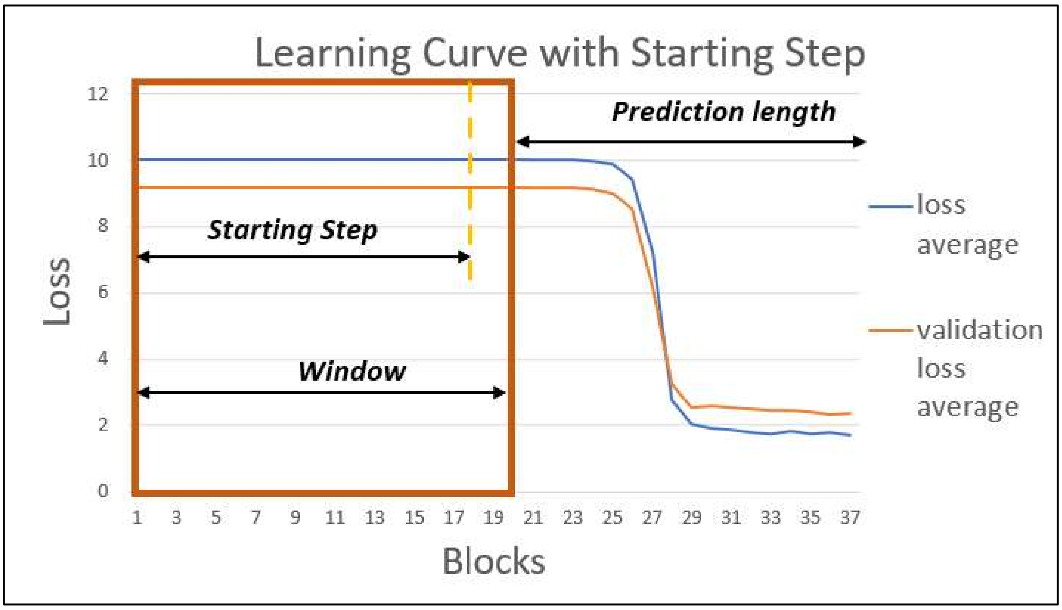
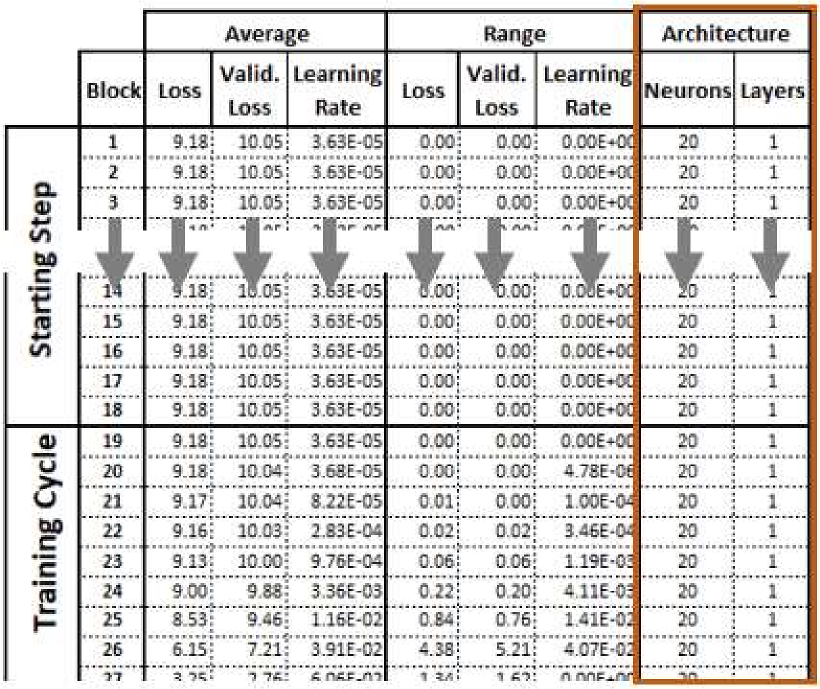


Figure 3-15 A learning curve with the starting step attached to make the prediction length independent of the window size.

Joining the Learning Curves in Series

With the training cycle converted into blocks and the starting step attached, the number of neurons and the number of hidden layers is joined to the learning curve data, see Figure 3-16. This is to allow the sequential prediction model to determine the relationship between the neural network architectures and the corresponding learning curve information. To train the sequential prediction model, a total of 9 features are used in parallel and these are the block number, and the average and range for loss, validation loss, and learning rate in each $block_{size}$ as well as the neurons per hidden layer and number of hidden layers. All the block learning curves for all selected neurons and layer architectures are then joined in series, as represented in a line graph in Figure 3-17, to create a $train_{seq}$.

All the variables are then feature scaled, which normalises the sequential training set for all datasets. This allows the sequential prediction model to work effectively without resetting the internal hyperparameters to adapt to different datasets as the number of features, the number of instances, the range of data and the types of data are always the same, regardless of the initial input dataset. The only parameters which vary for each dataset is the trend in learning curves in relation to the chosen architectures.



		Average			Range			Architecture	
	Block	Loss	Valid. Loss	Learning Rate	Loss	Valid. Loss	Learning Rate	Neurons	Layers
Starting Step	1	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1
	2	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1
	3	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1
	14	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1
	15	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1
	16	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1
	17	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1
	18	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1
Training Cycle	19	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1
	20	9.18	10.04	3.68E-05	0.00	0.00	4.78E-06	20	1
	21	9.17	10.04	8.22E-05	0.01	0.00	1.00E-04	20	1
	22	9.16	10.03	2.83E-04	0.02	0.02	3.46E-04	20	1
	23	9.13	10.00	9.76E-04	0.06	0.06	1.19E-03	20	1
	24	9.00	9.88	3.36E-03	0.22	0.20	4.11E-03	20	1
	25	8.53	9.46	1.16E-02	0.84	0.76	1.41E-02	20	1
	26	6.15	7.21	3.91E-02	4.38	5.21	4.07E-02	20	1
	27	3.25	2.76	6.06E-02	1.34	1.62	0.00E+00	20	1

Figure 3-16 An example of the dataset for training the sequential model.

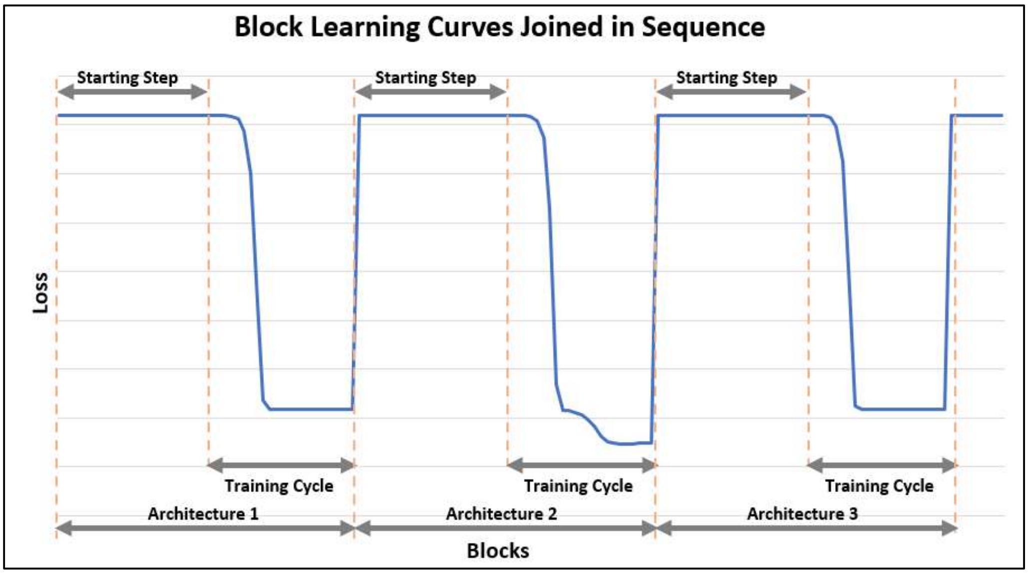


Figure 3-17 Multiple block learning curves attached in series

3.5.4 Stage 4: Training of the Prediction Model

To train the sequential prediction model, $train_{seq}$ must be converted into the format of a windowed dataset as described in Figure 3-3. Each number in the sequence is now replaced by a block which contains 9 input features in parallel, illustrated in Figure 3-18. Figure 3-18 represents the actual format required for sequential training. The window contains inputs for prediction (*X*) and the prediction (*Y*) shows outputs from prediction. Each row accounts for a single step forward in the dataset. Two important factors, the length of the window (*W)* and the length of the prediction (*P)* columns, must be properly defined. The following subsections will describe the justification for the chosen lengths of each factor, followed by the creation of the windowed dataset and finally training the sequential prediction model.

	Average			Range			Architecture	
Block	Loss	Valid. Loss	Learning Rate	Loss	Valid. Loss	Learning Rate	Neurons	Layers
1	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1
2	9.18	10.05	3.63E-05	0.00	0.00	0.00E+00	20	1

Window (X)						Prediction (Y)		
Block 1	Block 2	Block 3	Block 4	Block 5	Block 6	Block 7	Block 8	Block 9
Block 2	Block 3	Block 4	Block 5	Block 6	Block 7	Block 8	Block 9	Block 10
Block 3	Block 4	Block 5	Block 6	Block 7	Block 8	Block 9	Block 10	Block 11
Block 4	Block 5	Block 6	Block 7	Block 8	Block 9	Block 10	Block 11	Block 12
Block 5	Block 6	Block 7	Block 8	Block 9	Block 10	Block 11	Block 12	Block 13
Block 6	Block 7	Block 8	Block 9	Block 10	Block 11	Block 12	Block 13	Block 14
Block 7	Block 8	Block 9	Block 10	Block 11	Block 12	Block 13	Block 14	Block 15

Figure 3-18 An example of the windowed data for the sequential prediction model.

The Window Length

As shown in Fig 3-19, the length of the learning curve for each network architecture used for $train_{seq}$ includes both the length of the starting step (*S*) and the training cycle (*C*).

A constraint with attaching the starting step to each training cycle is that the length of the window (W) increases with S. If W is less than or equal to S, the sequential prediction model will not be able to determine when the starting step begins or ends, as the window will contain the same block information for all blocks while passing each starting step. Additionally, for predicting the performance of each neural network architecture, W should be as short as possible to maximise the prediction length (P). The smallest achievable value of W which fully covers S can be defined by Equation 4, the chosen W is two blocks larger than S. This can encompass the starting step with a single block on either side so that, during training, the window will have both the start and end of the starting step inside the window length.

$$W_{ideal} = S + 2 \tag{4}$$

The two blocks will still need to be trained for the prediction window to work, that is why the initial learning rate is set to lr_{base} and does not increase for two blocks, as expressed in Figure 3-6. With an initial negligible difference between the learning curve for all network architectures, the starting window for all architectures in $train_{seq}$ and the prediction window will be the same.

Increasing *W* raises the quantity of synthetic blocks in *S.* Increasing *W* beyond the length of *C+2* will merit no additional benefit as the start and end of *C* will be encompassed in the window size during training and this will only reduce the ratio of *C*, the useful information, to S as *W* increases further. Previous studies have shown that the size of the training window provides context to the sequence being analysed by the sequential model (Graves, 2012, Jaén-Vargas et al., 2022), for this reason *W* is set to the maximum useful size while accommodating the constraints of increasing *S.* This results in Equation 5, the chosen equation for defining *W*, meaning that *S* will be equal to *C*.

$$W = C + 2 \tag{5}$$

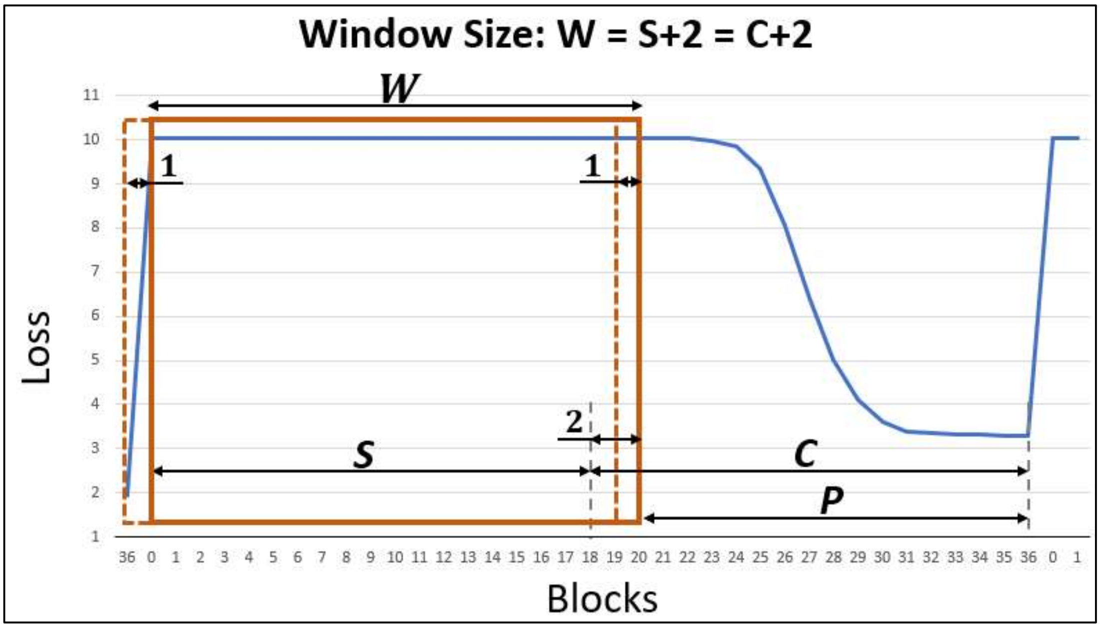


Figure 3-19 The block learning curve with the window size (W), the starting step length (S) and the training cycle length (C)

The Prediction Length

It can be taken from Figure 3-19 that *P* could be calculated with Equation 6. This would mean that the prediction is made after *W* and predicting the remaining learning curve until the end of the training cycle after the initial two blocks have occurred.

$$P = C - 2 \tag{6}$$

As $train_{seq}$ has multiple learning curves attached in series, represented in Figure 3-17, an issue occurs with using (5) is that the model begins to predict the beginning of the next architecture in the series, resulting in the predicted learning curve having a sharp rise at the end. This is fixed by reducing *P* by a further 2 blocks so that the rise is never recorded in the prediction at that stage in the sequence, resulting in Equation 7.

$$P = C - 4 \tag{7}$$

An example of this problem and solution are shown in Figure 3-20, a line graph of the actual learning curve and predicted learning curve with each equation for *P.* With Equation 6, the prediction has a sharp rise in the learning curve in the final blocks (Figure 3-20b). Reducing the prediction length allows for the learning curve to be predicted while eliminating this issue (Figure 3-20a).

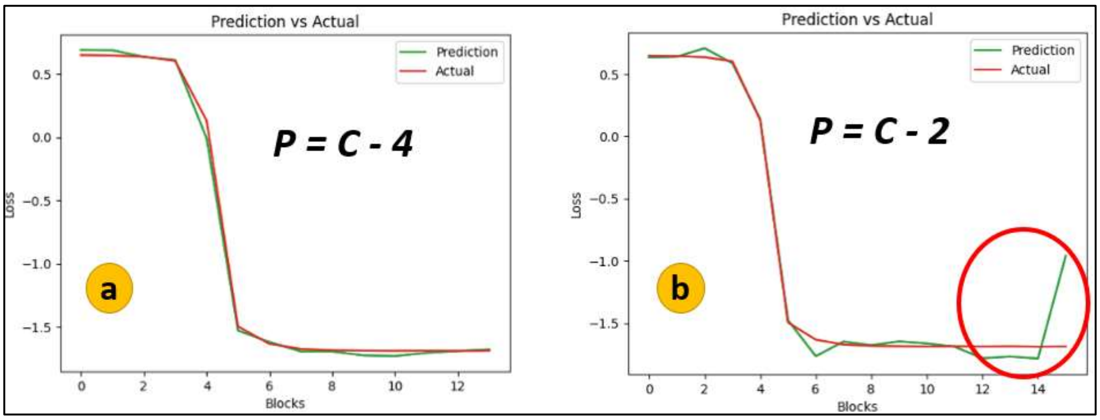


Figure 3-20 Actual vs Predicted learning curves with (a) a prediction which is 2 blocks less than the training cycle length and (b) a prediction which reaches the end of the training cycle.

Creating the Windowed Dataset

With values for both W and P defined, and $train_{seq}$ produced, the windowed training set ($train_{window}$) can be created. This is achieved using *Figure 3-21.* The result of this stage is an input dataset, X, which consists of the 9 input features in parallel being sorted into the window format, shown in Figure 3-18 and a corresponding output set of labels, Y, which only has a single feature. This feature is the average validation loss for the length *P*, the number of steps ahead for prediction. This is the required feature to produce the learning curves for each architecture. Using the predicted average validation loss for the length *P*, the trend in overall performance can be predicted for all architectures.

Creating $\boldsymbol{train_{window}}$

inputs

$train_{seq}$

P

W

for value in $length(train_{seq} - W - P)$:

$row =$ value:(value + W)

$X = X.append(row)$

for value inlength$(train_{seq} - W - P)$:

for step in(0: P, increment = 1):

$label = train_{seq}[value + W + step][Average\ Val\ Loss]$

$label = labels.append(label)$

$Y = Y.append(labels)$

Figure 3-21 The pseudocode of the algorithm to create train$_{window}$.

CGRNN Architecture

For this study, the individual 1DCNN, GRU, Long Short-Term Memory RNN (LSTM)(Hochreiter and Schmidhuber, 1997), as well as the RNN were considered for prediction. These architectures and hybrids of each were compared as an initial assessment through trial and error to determine which would be most suitable for this application. The 1DCNN+GRU and 1DCNN+LSTM architectures produced the best learning curve predictions. However, the 1DCNN+LSTM had a larger computational cost, resulting in the 1DCNN+GRU, termed the CGRNN architecture, being selected for this study.

The chosen CGRNN architecture consists of a 1DCNN layer, multiple GRU layers and a final dense layer. This base architecture has been optimised with Keras Tuner (O'Malley, 2019), using GPBO to get the best sequential prediction model to be used as the CGRNN. Table 3-3 shows the range of hyperparameters which were optimised, as well as the resultant best combination of hyperparameters for the CGRNN. These settings are used for all experiments in this study. The final optimised CGRNN architecture is shown in Figure 3-22. **It must be clarified that the CGRNN hyperparameters have not been changed for any variation in dataset after the initial optimisation and the same parameters have been used for all experiments.**

Table 3-3 CGRNN Hyperparameters: Range and best result

Stage	Hyperparameter	Range	Best Result
Global	Learning rate	1e-9 - 1e-1	0.00213
1DCNN Layer	Filter	10-300	274
	Kernel	2-9	8
	Strides	1-3	1
GRU	Layers	1-3	3
	Units/layer	10-80	32
Dense Layer	Units	5-100	100

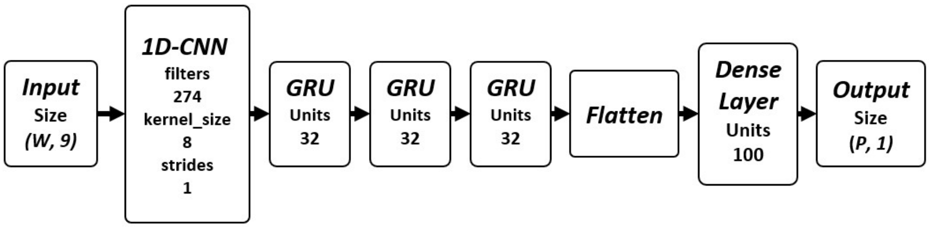


Figure 3-22 The structure of the optimised CGRNN architecture for learning curve prediction.

Training the CGRNN Model

The *X* and *Y* datasets are split into training and validation datasets, and the CGRNN model is optimised with the mean absolute error as a metric (MAE) and an Adam optimiser (Kingma and Ba, 2014). The Adam optimiser is a popular algorithm in machine learning that adaptively adjusts the learning rate for each parameter by combining two techniques: momentum (to smooth gradients) and RMSProp (to scale updates based on recent gradients). This allows Adam to converge faster and more effectively on optimal solutions without extensive tuning, making it ideal for training deep neural networks.

The equation for the MAE is shown in Equation 8, with y_{pred} being the prediction for each instance, y_{actual} being the true label of each instance, and *n* being the number of instances.

$$MAE = \frac{\sum_{i=1}^{n} |y_{pred} - y_{actual}|}{n} \tag{8}$$

The model is trained for 200 epochs, a number selected to be suitable through trial and error, and the MAE for both the training and validation datasets is recorded. Through iterative experimentation, it was deemed that a suitable level of prediction performance was achieved when the MAE of the training set was less than 0.02. The validation MAE is monitored throughout the 200-epoch training cycle and the model parameters for the best validation MAE with the training MAE under 0.02 are selected for prediction. As $train_{seq}$, the source for X and Y, always has the same features, the same number of rows, the same type of features and the same scale, these required training parameters and CGRNN hyperparameters do not need to change when the input dataset for HPO changes, as the only variation is the relationship between the changing learning curve shape and the architecture details. This allows for flexibility with different datasets while not changing the training method and hyperparameters for the CGRNN.

3.5.5 Stage 5: Prediction of Learning Curves

With the trained CGRNN, the next step of the SELECT method is to predict the learning curve of all architectures in the search space from a single prediction window. This is achieved with the following:

1. Train a single neural network architecture for the length of two blocks, stop training and record the learning curve data.
2. Convert the learning curve data into the parallel blocks with the loss, validation loss and learning rate information as before.
3. Attach the starting step to the blocks of data, creating the single prediction window learning curve needed for the CGRNN, a line graph representation is given in Figure 3-23.

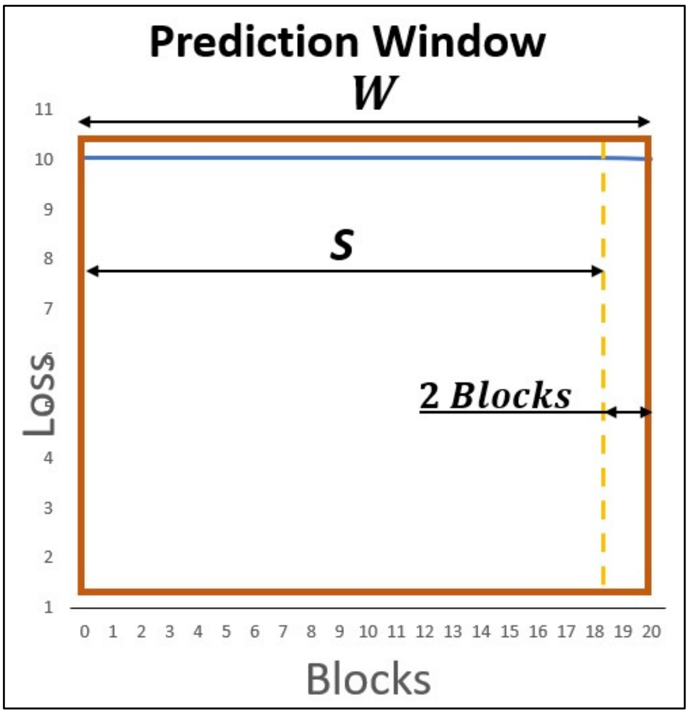


Figure 3-23 The prediction window of the learning curve.

4. For each architecture (neurons per layer and quantity of layers) in the search space, attach the same prediction learning curve data from the single window and predict the learning curve with the CGRNN model for all architectures without running them.

The learning curve of all architectures will be produced from this single prediction window. Fig 3-24 is an example of the predicted learning curve from the CGRNN with 2 hidden layers and 95 neurons against the actual learning curve using the QSAR Fish Toxicity dataset (Cassotti et al., 2015). It shows a learning curve covering 140 epochs, or 14 blocks of 10. The final loss values of this network architecture from the learning curve are highlighted in the green box. Recording the final loss from every predicted architecture provides a predicted performance of these architectures without training them.

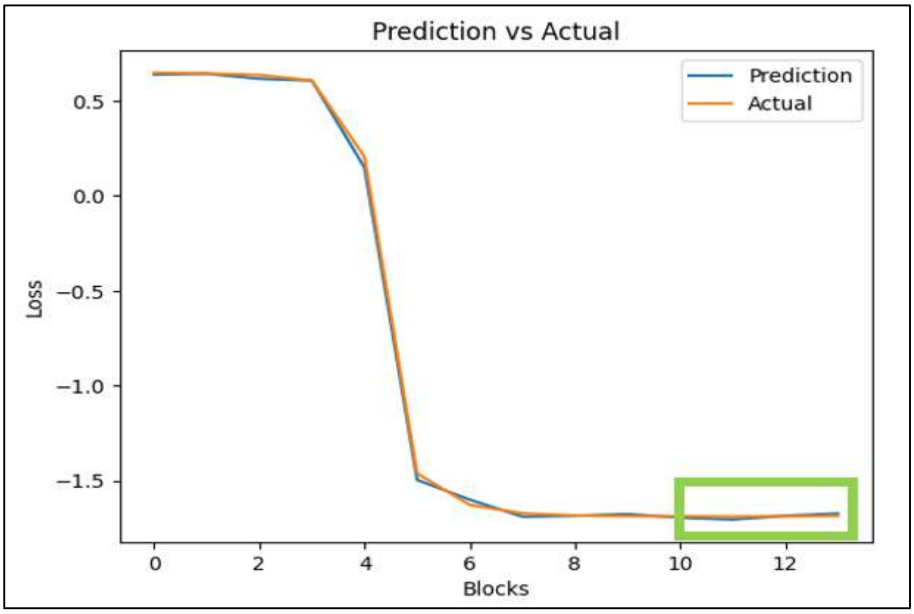


Figure 3-24 Example of the predicted and actual performance from the CGRNN.

If the predicted final loss value within the green box in Figure 3-24 is sorted by hidden layer and then by neurons per layer for all architectures in the search space, the trend in performance can then be visualised as shown in Figure 3-25. The actual results (blue) are the measured final loss values from running all architectures in the search space, while the predicted results (red) are from the SELECT method, using the CGRNN predictions. This is taken from a $train_{seq}$ created from the QSAR Fish Toxicity dataset. The trend in loss shows that the best performance (minimum loss) will be achieved with neural networks which contain 2 hidden layers and 80-100 neurons per hidden layer. The predicted trend in performance is like the actual, also presenting the same location for the best performing neuron and hidden layer combinations. These results show that there is a strong correlation between the number of hidden layers and the number of neurons per hidden layer, and the learning curve for training a neural network. The predicted learning curves can be used to rank the network architectures by performance so that the best performing models can be selected and trained to quickly achieve the best results in HPO.

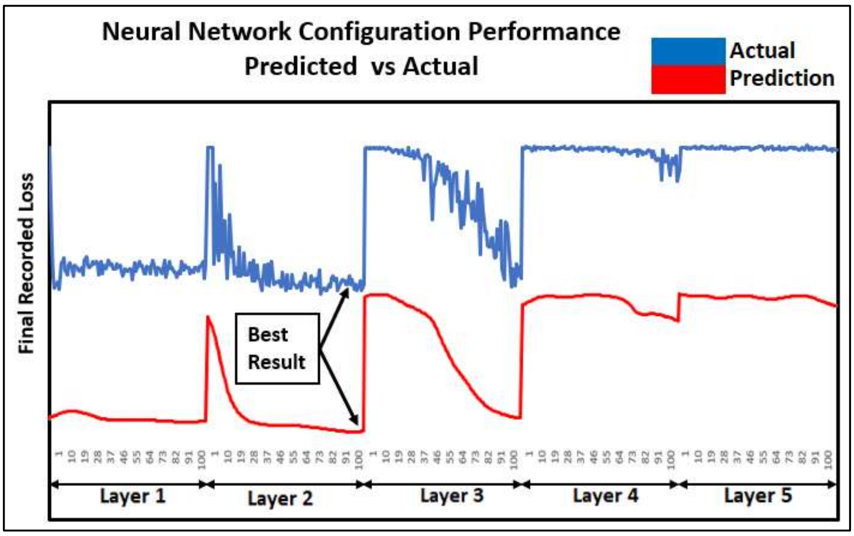


Figure 3-25 The performance trend of the predicted and actual learning curves sorted by hidden layer and neurons per layer.

3.5.6 Stage 6: Trialling the Best Results

Although the predicted trend line is very close to the actual line (Figure 3-25), the variation in loss performance of the actual line cannot be accurately anticipated. This is why the prediction line is used as a guide to look for the best performing models. Ranking the predicted learning curve results by the best final loss, creates an ordered list of architecture combinations which show the top performing models. The best prediction has 2 hidden layers and 95 neurons with minimum loss for the example in Figure 3-25. When comparing this to the actual performance results, the same architecture is the third best performing, i.e., the top 0.6% of the 500 tested combinations. From the top 50 best trained architectures, the prediction model includes 72% of these results in the same range. This implies that this approach can predict good models without running them, hence reducing a significant amount of computational effort.

Taking this advantage, the top 2% (top 10 architectures) of predictions have been selected in this study to train to obtain the actual performance on the test set. This range was chosen because 60% of these predictions fit in the top 6% of all actual performances. This would suggest a high likelihood of finding a good performing model while limiting the number of combinations to train. Increasing the percentage of trained architectures from the ranked predictions would further increase the chance of finding a good model but also raise the required computational effort.

When tuning the learning rate, the aim is to get a close approximation to the best performance within a set cycle time, *C,* to rank architecture performance. The approach taken for predicting performance uses the ranked MAE level of the learning curves to select top models but does not consider the gradient of the learning curves. To overcome this limitation, once the best architectures are selected, the training epoch limit is extended, and the best performing architectures will only stop once the performance stops improving. This is achieved with the use of an early stopping callback. An early stopping callback is a tool used to monitor the validation loss and stop the training cycle when the validation loss does not reduce over a set number of epochs, defined as patience. In this study, the early stopping mechanism starts at the end of the length of C and monitors the validation loss with a patience of 50 epochs. The best recorded weights for the minimum loss are then chosen as the best parameters for the trained configuration. This extended training cycle is given to all the predicted top models from the CGRNN. The best performing of these is selected as the model with the best hyperparameters.

3.6 Validation of the SELECT HPO Method

This section will discuss the experiments that were designed to evaluate the performance of the SELECT method and compare it against the benchmarking algorithms. This will cover the chosen datasets, the benchmark algorithms, the experimental set up and the results with discussion.

3.6.1 Dataset for Analysis

With the lack of available data for the purposes of this study, a method of validation was taken to evaluate the SELECT method over multiple varied datasets which are openly available and used in other research studies to determine the performance of the algorithm with variations in feature types and quantities, the instances in each dataset and for multiple contexts. To achieve this, multiple datasets were selected from the UCI Machine Learning Repository , a repository of machine learning datasets which has become highly popular in all areas of data based research. The datasets used for this study are shown in Table 3-4, with description, assigned code, the prediction output, the number of input features, the number of instances, and sources of references. These datasets cover multiple fields, with variations in the feature quantity and types, and the number of instances in each dataset. The datasets shown in Table 3-4 have been solely selected for the purpose of validation and have not been included in the development process of the SELECT method. The chosen datasets have been verified to have no incorrect or missing data and have been used in previous research.

These datasets were carefully selected with several criteria considered. As the application of this HPO method for industrial impact related to continuous performance metrics, all datasets have been selected to be regression datasets, each with a continuous prediction variable. There was also a desire to find diverse datasets from different fields and those which have been used in previous research. There was also a consideration for variations in the quantity of inputs and instances, but a limit to the number of instances due to limitations in computational capacity. All the datasets were selected to be tabular datasets as well.

Table 3-4 The details of the validation datasets

Description	Code	Output	Input Qty	Instances	Sources
Air foil Self-Noise	D1	Scaled sound press. level	5	1503	(Brooks et al., 1989, González, 2008, Lau et al., 2009)
QSAR Fish Toxicity	D2	Quantitative Response	7	908	(Cassotti et al., 2015)
Concrete Compressive Strength	D3	Concrete Compressive Strength	8	1030	(Yeh, 1998, Yeh, 2006)
Behaviour of Urban Traffic	D4	Slowness in traffic	17	135	(Sassi et al., 2011)
Auto MPG	D5	Mpg	7	398	(Quinlan, 1993)

3.6.2 Benchmark Algorithms

Four well-established benchmark algorithms; RS, GPBO, TPE, HB have been carefully chosen for comparison against the SELECT method, listed in Table 3-5. The selection of these algorithms is grounded in recent studies that have demonstrated their suitability for comparison (Motz et al., 2022, Bischl et al., 2023). RS serves as a computationally efficient baseline, offering simplicity and ease of implementation, making it a relevant point of reference. GPBO leverages Gaussian Processes to model intricate relationships within the hyperparameter space, showcasing a sophisticated approach that has proven effective in capturing complex objective functions. TPE introduces a probabilistic model that adeptly balances exploration and exploitation, providing a middle ground between randomness and guided search. Meanwhile, HB stands out for its adaptive resource allocation strategy, emphasising the importance of efficient resource utilisation in the optimisation process. By incorporating these widely recognised algorithms, the benchmarking process ensures a comprehensive evaluation, drawing upon the strengths and diversity of these algorithms to assess the performance of the SELECT method. The libraries used for these algorithms are Hyperopt (Bergstra et al., 2013) and Keras Tuner (O'Malley, 2019). The pros and cons for each HPO approach are provided in Table 3-5.

Table 3-5 Benchmark HPO Algorithms

Method	Pros	Cons
Random Search	• Simplicity and ease of implementation. • Can perform well with a low computational cost. • Requires minimal tuning. • Suitable for parallelisation	• Inefficient in finding optimal hyperparameters. • Does not adapt based on observed performance. • May waste resources on less promising configurations.
Bayesian Optimisation	• Efficient in handling noisy or expensive objective functions. • Adaptive exploration of the hyperparameter space. • Converges to optimal solutions with few evaluations.	• May waste resources on less promising configurations. • Poor capability for parallelisation.
Tree Parzen's Estimator	• Efficient with all kinds of hyperparameters. • Balances exploration and exploitation effectively.	• Performance may depend on the quality of the surrogate model. • Poor capability for parallelisation.
Hyperband	• Efficiently allocates resources to promising configurations. • Successive halving for effective resource utilisation. • Good capability for parallelisation	• Can eliminate slow converging, high performance configurations.

Random Search

The Random Search (RS) algorithm (Bergstra and Bengio, 2012b) is employed in this study, training randomly sampled configurations from the search space to identify the best performance. Unlike more advanced optimisation algorithms, RS does not rely on specific assumptions, making it a computationally efficient and effective baseline for comparison. Within a set number of epochs, RS selects and samples random configurations from the search space, tests them sequentially, and selects the configuration with the best performance based on a predefined metric. Despite its simplicity, RS has proven to be a preferred optimisation algorithm due to its computational efficiency, requiring fewer resources compared to methods like Grid Search, while still identifying high-performing hyperparameters (Andonie, 2019, Zöller and Huber, 2021). Here is a description of how RS operates:

- **Define the Parameter Space:** RS begins by establishing a defined range or set of discrete values for each hyperparameter. Each parameter range represents an independent dimension in a multi-dimensional search space, within which the optimal configuration is likely to exist.
- **Random Sampling from the Search Space:** RS then samples configurations at random from this multi-dimensional space. Each randomly selected point corresponds to a unique combination of hyperparameter values. For each iteration, RS assigns random values to each hyperparameter independently, resulting in configurations that may vary widely across the search space.
- **Train and Evaluate Each Sampled Configuration:** For each randomly chosen configuration, the model is trained for a specified number of epochs. The performance of each configuration is then evaluated on a validation set using a chosen metric.
- **Repeat Sampling:** RS repeats the process of sampling, training, and evaluating multiple configurations until a specified stopping criterion is met.
- **Select the Best Configuration:** After all sampled configurations have been evaluated, RS selects the configuration with the highest performance metric as the best hyperparameter set.

Bayesian Optimisation with Gaussian Process

GPBO leverages Gaussian Processes to model the objective function during optimisation, providing a nuanced representation of the performance landscape across different hyperparameter configurations. The objective function, in this context, serves as the metric to be either maximised or minimised, reflecting the performance of the machine learning model. GPBO selects configurations through an acquisition function that adeptly balances exploration and exploitation, guided by the insights derived from the Gaussian Process model. This approach aligns with the growing popularity of sequential model-based algorithms in hyperparameter tuning, exemplified by BO (Snoek et al., 2012). BO employs a directed search strategy, leveraging Bayes' Theorem to construct a probability model for selecting the most promising hyperparameters for trial. Diverging from the random sampling approach of RS, BO maintains a record of previous trial performances, using this information to map hyperparameters to a probability score for the objective function. The resulting probability model, referred to as a surrogate, guides the selection of models likely to perform well. This guided approach, rooted in leveraging past performance to inform future trials, has demonstrated its effectiveness in rapidly achieving high accuracy, presenting a highly competitive strategy, particularly in tasks such as neural architecture searching(Liu et al., 2022). Here is a description of the steps in the BO process:

- **Define the Parameter Space:** BO begins by defining a range for each hyperparameter.
- **Initialise with a Small Number of Random Samples:** A few random configurations are selected, and their performance is evaluated.

- **Model the Objective Function with a Gaussian Process (GP):** These initial data points are used to construct a model of the objective function. GP models are a central component of BO, as they provide a probabilistic estimate of the objective function across the search space. For each point in the search space, the GP provides a mean prediction and a variance, creating a distribution over the objective function.
- **Choose an Acquisition Function to Guide Sampling:** The acquisition function uses the GP's mean and variance estimates to decide the next point to evaluate, balancing exploration and exploitation.
- **Evaluate the Selected Configuration:** The hyperparameter configuration chosen by the acquisition function is evaluated on the objective function (the model's performance on a validation set). The result from this evaluation is added to the dataset used to train the GP, improving the model's accuracy and reducing uncertainty in this region of the search space.
- **Iterate Until the Stopping Criterion is Met:** BO iteratively updates the GP model with new observations, refines the acquisition function, and selects new configurations until a stopping criterion is reached.
- **Select the Best Configuration:** Once the search ends, the configuration that achieved the highest performance is selected as the optimal set of hyperparameters.

Tree Parzen's Estimator

Transitioning from GPBO to TPE, TPE is another BO variant that further refines the optimisation process by explicitly separating it into exploration and exploitation phases. In contrast to GPBO's unified probabilistic model, TPE constructs distinct probability density models for configurations that exhibit good and poor performance (Rong et al., 2021). By creating these separate models, TPE more explicitly focuses on distinguishing between promising and less promising regions within the hyperparameter search space. This dual model approach allows TPE to strategically sample hyperparameters, allocating resources more efficiently by dedicating exploration efforts to areas with the potential for improved performance and exploitation efforts to exploit known successful configurations. The success of TPE in achieving high levels of performance has been demonstrated in prior research (Abbas and Myungho, 2023, Motz et al., 2022). This strategy of partitioning the optimisation process into exploration and exploitation, coupled with the construction of separate probability density models, has proven effective in yielding competitive results compared to other HPO methods. This emphasis on targeted exploration exploitation aligns with the broader framework of Bayesian Optimisation, showcasing the adaptability of the approach in various formulations like GPBO and TPE for efficiently navigating and optimising complex hyperparameter spaces. The approach for TPE for HPO is as follows:

- **Define the Parameter Space:** TPE begins by specifying a range of values for each hyperparameter, establishing a multi-dimensional search space where each point corresponds to a specific hyperparameter configuration.
- **Initialise with a Small Number of Random Samples:** A small set of random hyperparameter configurations is generated. Each configuration is evaluated on the objective function, and their corresponding results are recorded.
- **Construct Density Models for Good and Bad Configurations:** TPE utilises two kernel density estimators (KDE) to model the distribution of the objective function values:
    - **Good Configurations:** This model represents hyperparameter configurations that yield low objective function values, meaning better performance.
    - **Bad Configurations:** This model represents configurations that yield higher objective function values, meaning poorer performance.
- **Define the Acquisition Function:** The acquisition function in TPE is set up to balance exploration and exploitation by computing the expected improvement of selecting a new hyperparameter configuration. It identifies the next point in the search space that maximises the likelihood of improving performance based on the density models constructed.
- **Evaluate the Selected Configuration:** The hyperparameter configuration suggested by the acquisition function is evaluated against the objective function.

The outcome of this evaluation is then added to the dataset, informing the density models and refining their accuracy.

- **Iterate Until the Stopping Criterion is Met:** TPE iteratively updates the density models with new observations, recalculates the acquisition function, and selects new hyperparameter configurations until the stopping criterion is achieved.
- **Select the Best Configuration:** Upon completion of the search, the hyperparameter configuration that produced the lowest objective function value is selected as the optimal set of hyperparameters.

Hyperband

HB has emerged as a compelling benchmark for HPO due to its innovative integration of Random Search RS with successive halving, creating a two-phase approach that efficiently navigates and optimises hyperparameter configurations (Li et al., 2017). In the initial phase, a diverse set of random configurations undergoes evaluation, with resource allocation favouring the top performing configurations. This distinctive process quickly identifies promising hyperparameter configurations, directing additional resources toward their further evaluation. The approach has performed well comparatively and is highly suitable for parallelisation (Motz et al., 2022, Vishnu et al., 2022). The effectiveness of Hyperband lies in its strategic resource allocation and its adaptability to the evolving landscape of promising configurations. The second phase involves a resource-intensive selection process, driven by successive halving, where only the best-performing configurations continue to receive increased resources. This iterative method ensures that the most promising configurations are allocated more resources, leading to the eventual identification of a final, best-performing model. The integration of random selection, capability for parallel training, and successive halving makes HB a robust choice, providing a comprehensive and efficient exploration of the hyperparameter search space. The procedure taken by HB is as follows:

- **Define the Parameter Space:** HB begins by specifying a range of values for each hyperparameter, establishing a multi-dimensional search space where each point corresponds to a specific hyperparameter configuration.
- **Initialise with a Small Number of Random Samples:** A small set of random hyperparameter configurations is generated. Each configuration is evaluated using a specified budget of epochs and performance, and their corresponding results are recorded.
- **Set the Resources and Bandwidth:** The total available resource computational time is divided among different configurations. The algorithm determines a set of “bandwidths”, or allocated number of epochs, to allocate resources dynamically as configurations are evaluated.
- **Evaluate Configurations with Early Stopping:** HB uses a multi-armed bandit approach, where configurations are evaluated progressively. After each evaluation phase, the worst-performing configurations are eliminated based on their performance. This process continues, allocating more resources to the remaining configurations.

- **Iterate with Increased Resource Allocation:** The algorithm iteratively adjusts the resource allocation among the surviving configurations, providing more resources to the better performing ones while reducing resources for those that perform poorly.
- **Converge on Optimal Configurations:** As the iterations progress, HB focuses on configurations that show promise based on previous evaluations. The algorithm continues to refine its search until a predetermined stopping criterion is reached.
- **Select the Best Configuration:** Upon completion of the search, the hyperparameter configuration that achieved the best is selected as the optimal set of hyperparameters.

3.6.3 Experimental Setup

To ensure robustness and fairness in this analysis, all algorithms have been fine-tuned to achieve their best performance prior to the comparison. Details are given in the following sub-section on the experimental setup and the fine tuning for all HPO approaches.

Experimental Setup for All HPO Methods

The goal for all HPO methods in this study is to optimise the learning rate, the number of neurons per hidden layer and the number of hidden layers for an MLP neural network architecture to get the best prediction accuracy with the smallest computational cost. The search space is a learning rate range from 1e-9 to 1e-1, 1-5 hidden layers and 1-100 neurons per hidden layer. The prediction accuracy is measured by the MAE, as defined in Equation 8. The computational cost is measured by the number of epochs and clock time it takes to get the best performing neural network architecture. The maximum epoch quantity for each network iteration is defined as 200 epochs. The benchmark approaches were evaluated with 100, 200, and 400 epochs. There was a significant improvement in performance from 100 to 200 epochs, but equivalent results with 400. The 200-epoch limit was selected as a result. Each of the HPO approaches were assessed on the five datasets shown in Table 3-4, with a 5-fold split for each optimisation. The average MAE was calculated on the test set for each, and total epoch quantities and time (in seconds) were recorded.

The SELECT method uses an extended epoch limit beyond the length of C for trialling the predicted top performing learning curves to ensure the top performing models reach a minimum. This is due to the nature of the CGRNN predictions only considering the level of MAE for ranking performance, but the gradient is not used for the ranking, so the extended epoch limit allows the results to reach a minimum based on observed performance. The benchmarks operate on a set epoch limit throughout which is a variation in method. To provide a fair comparison, the chosen best configuration for each of the benchmarks is compared with both the result of the 200-epoch limit and the extended epoch limit using an early stopping callback. This will ensure robustness of the comparison, despite the difference in approach.

As the SELECT method sets the lr_{set} for each model individually, an adaptive learning rate was selected for the benchmark HPO methods to ensure fairness. The benchmarks are given the 'Adam'(Kingma and Ba, 2014) optimiser for the full 200 epoch limit and the proposed approach uses the same optimiser during n_{run}.

Setup of the SELECT Method

The SELECT method requires several parameters to be set as shown in Table 3-6. The $train_{seq}$ dataset is created by training all possible architecture combinations of values in $layer_{train}$ and $neuron_{train}$. The length of n_{set} is 80 epochs, meaning lr_{set} will be determined between 80-100 epochs into each cycle. Then n_{run} is set to 100 epochs, creating a total *C* of 180-200 epochs. $Block_{size}$ is set to 10 epochs. The maximum number of epochs to create the training set for the CGRNN model in the SELECT method with this set up is 6,000 epochs. Additional epochs will be required for tuning the learning rate and testing the $trial_{quantity}$ of best results.

Table 3-6 The control parameters for the proposed approach

Symbol	Description	Value
$layer_{train}$	The layers used to create $train_{seq}$	1,2,3,4,5
$neuron_{train}$	The neurons used to create $train_{seq}$	1,20,40,60,80,100
n_{set}	Learning rate setting epoch quantity	100
n_{run}	Train epoch quantity with lr_{set}	80
$block_{size}$	Interval length for recording training data	10
$trial_{quantity}$	Quantity of trials for best result	10

Random Search, Bayesian Optimisation (GP&TPE)

RS, GPBO and TPE operate sequentially so the setup is the same for all these algorithms. Each algorithm was assessed for 100 iterations at 200 epochs to select the best performing model, i.e., a total of 20,000 epochs. The number of epochs to converge on the best result was recorded as the total epoch quantity.

Hyperband

The HB model utilises parallel training to optimise the parameters of the neural network. To determine a test size, the equation for the number of epochs in a single iteration, as expressed in Equation 9, from(Li et al., 2017). With the HB algorithm, 200 epochs with a successive halving factor of 2, results in 11685 epochs. Covering 2 iterations of the same settings would lead to 23,370 epochs used for the trial. The number of epochs to converge on the best result was recorded as the total epoch quantity.

$$Iteration = max\ epochs * (math.log\ (max\ epochs, factor) ** 2) \tag{9}$$

3.6.4 Results and Discussion

In this section, the experiment results involving the use of the SELECT method, and the four benchmark comparisons are presented in Table 3-6, Table 3-7, Table 3-8, Table 3-9 and Table 3-10 respectively. These are discussed, focusing first on the prediction accuracy then the computational expense.

Prediction Accuracy

The MAE results of all algorithms are shown in Table 3-7 and Table 3-8. The SELECT results are taken from the extended epoch limit (Ext.), while the benchmarks have results from both the set 200 epoch and Ext. limit to allow for both metrics to be compared for the benchmarks. These results are the average and standard deviation of the best performing configurations taken over the 5-fold split of the datasets. The best performances for each row are highlighted in bold.

Table 3-7 MAE at 200 epoch limit and 5-fold split

	SELECT		RS		GPBO		HB		TPE	
Dataset	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**
D1	***2.14***	0.52	3.37(+58%)	1.15	3.6(+68%)	0.56	3.07(+43%)	0.49	3.34(+56%)	1.02
D2	***0.75***	0.07	0.83(+11%)	0.11	0.79(+6%)	0.09	0.84(+13%)	0.10	0.79(+6%)	0.10
D3	***4.08***	0.53	4.87(+19%)	0.80	4.86(+19%)	0.53	4.78(+17%)	0.84	4.83(+18%)	0.83
D4	***2.26***	0.25	2.7(+19%)	0.66	2.75(+22%)	0.79	2.7(+19%)	0.37	3.11(+38%)	1.07
D5	***2.09***	0.28	2.53(+21%)	0.74	2.42(+16%)	0.42	2.29(+9%)	0.27	2.24(+7%)	0.31

*M = Mean, SD = Standard Deviation, Note: '( )' refers to the percentage difference between the benchmark metric and the SELECT result.

Table 3-8 MAE at extended limit and 5-fold split

	SELECT		RS		GPBO		HB		TPE	
Dataset	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**
D1	***2.14***	0.52	3.07(+43%)	1.10	2.4(+12%)	0.73	3.47(+62%)	1.07	3.15(+47%)	0.81
D2	***0.75***	0.07	0.87(+17%)	0.14	0.84(+13%)	0.13	0.82(+9%)	0.10	0.86(+14%)	0.09
D3	***4.08***	0.53	5.29(+29%)	1.35	5.12(+25%)	0.55	5.00(+22%)	0.61	5.00(+22%)	1.45
D4	***2.26***	0.25	2.77(+23%)	0.52	2.74(+21%)	0.52	2.85(+26%)	0.61	3.18(+41%)	0.85
D5	***2.09***	0.28	2.61(+25%)	0.82	2.56(+23%)	0.20	2.68(+28%)	0.54	2.83(+35%)	0.53

*M = Mean, SD = Standard Deviation, note: '( )' refers to the percentage difference between the benchmark metric and the CGRNN result.

The SELECT method has performed better with the smallest MAE and highest consistency in results as compared to all benchmarks for all datasets. This is also for both the 200-epoch limit and the extended epoch limit given to the benchmarks. The extended epoch limit for the benchmarks resulted in worse performance overall, with a reduction in mean MAE for 75% of readings. This makes sense as the algorithms select the learning rate based on observed performance. Extending beyond this limit for the final test would lead to potential overfitting.

By predicting the performance of all hyperparameter configurations at once, the SELECT method establishes a more holistic view of the search space. One of the key advantages of the SELECT method is its ability to avoid local minima. The observation and reaction approach of the benchmark algorithms can lead to suboptimal regions of the search space due to their reliance on direct observations from previous trials. In contrast, SELECT's predictive model facilitates a broader exploration, allowing it to identify high-performing configurations that may otherwise be overlooked. By understanding the overall performance trends rather than focusing solely on past observations, SELECT mitigates the risk of overfitting to noise and enhances the likelihood of discovering optimal configurations. Another advantage to this approach over the benchmarks is the adaptive optimisation of the learning rate for each configuration, rather than using a one-size-fits-all approach as seen in the other methods. The learning rate that performs well with one model, may not necessarily perform well with other hyperparameter configurations. Having an adaptive learning rate can allow flexibility for improved performance.

Ultimately, the SELECT method's unique capability to synthesise information about hyperparameter interactions and performance trends proactively rather than reactively leads to superior results compared to benchmarks like GPBO, TPE, HB, and RS.

The looking at performance on individual datasets, the best performance comparison is in relation to ‘D1’, the Air foil Self-Noise dataset, with nearly all benchmark comparisons having above 40% error, excluding GPBO with the extended epoch limit. The SELECT method still produced a better performing model by 12% MAE error. The dataset with the closest performance between the benchmarks and the SELECT method is ‘D2’, relating to Fish Toxicity, with the SELECT selected configuration achieving a 6-17% better predictive accuracy than the benchmarks.

The better performance of the SELECT method over multiple datasets with variations in feature, instances, feature types and regressive correlation with a 5-fold split shows the robustness of this approach in this application.

Further information can be gained from observing the individual predictions made during the SELECT method. Figure 3-26 shows the comparison between the actual performance trends for the D1, or the Air foil Self-Noise, dataset using a single split of the data, against actual predicted performance using the SELECT method.

The prediction line in Figure 3-26 shows that the best performing model will be found with 4 and 5 hidden layers. These are ranked during optimisation numerically and trialled, with the top results shown in Table 3-9.

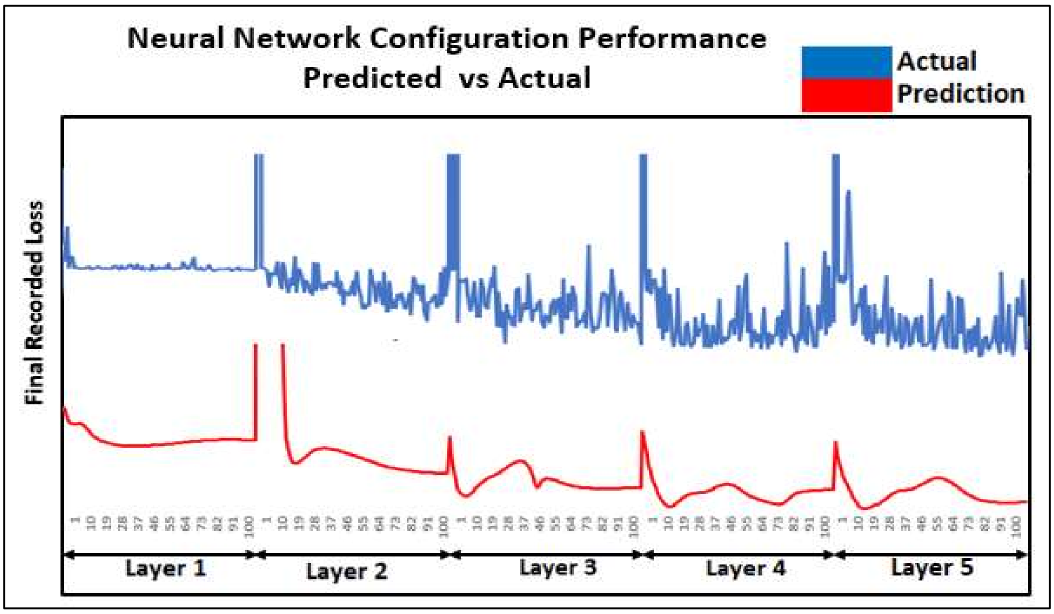


Figure 3-26 The actual and predicted performance trend of neural network configurations.

Table 3-9 Trialled top ranked neural network configurations from a single optimisation with the Air foil Self-Noise dataset

Hidden layer	Neurons	lr_{set}	MAE
5	16	0.00482	1.62
5	15	0.00522	3.35
5	17	0.00661	1.66
5	18	0.00482	2.23
5	19	0.00661	1.67
5	14	0.00564	2.34
4	15	0.00611	2.09
4	**16**	**0.00564**	**1.58**
5	20	0.00661	2.08

The best model configuration has 4 hidden layers, 16 neurons and a lr_{set} of 0.00564. Out of these top models, 7 of them achieved better performance than the closest of the benchmarks, suggesting that the SELECT method is performing effectively in selecting the location of the best network architectures.

A further benefit of this approach is that it can direct the observer to a potentially better search space for better performance. This can be demonstrated from the example in Figure 3-26. The red line has a seasonality to it which suggests that as the hidden layers increase, the performance improves. A further experiment was carried out where the same dataset was trained over 2-6 layers, rather than 1-5, the belief being that the best performance would have 6 hidden layers. The results of this experiment are shown in Table 3-10.

Table 3-10 Trialled top ranked neural network configurations from a single optimisation with the Air foil Self-Noise dataset (D1) over 2-6 hidden layer search space.

Hidden layer	Neurons	Learning Rate	MAE
6	68	0.00433	1.66
6	67	0.00433	1.59
6	69	0.00433	3.28
6	**66**	**0.00403**	**1.43**
6	70	0.00375	1.72
6	65	0.00403	1.91
6	71	0.00375	1.82
6	64	0.00403	2.07
6	72	0.00403	1.52

The new best performing model has a MAE of 1.43, the best recorded in all readings for D1. Table 3-10 shows that all the top ranked performances contain 6 hidden layers, adding further proof to the effectiveness of the SELECT method. It not only predicts the best performing configurations, but it can also generate useful insights to uncover specific search spaces for better performance.

Computational Expense

The average required time and epochs to achieve the best results for the SELECT method and benchmarks are shown in Table 3-11.

The SELECT method took an average of between 6525-7073 epochs to find the best performing configurations over all datasets. When compared to all the benchmarks, the SELECT method achieved the best result with less epochs on average every time.

Table 3-11 Average and standard deviation of the required computational expense in seconds to achieve the best result.

	SELECT		RS		GPBO		HB		TPE	
Dataset	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**
D1	***989 (7073)***	198 (1287)	1460 (11187)	321 (2463)	1042 (8792)	692 (5840)	3267 (15783)	1188 (5738)	1193 (12200)	709 (6571)
D2	***658 (7007)***	85 (443)	1390 (12793)	708 (6514)	1200 (13234)	546 (6018)	1937 (11221)	808 (4679)	663 (8720)	494 (6455)
D3	***730 (6525)***	39 (375)	1203 (11509)	717 (6864)	1362 (12837)	427 (4028)	2632 (13403)	813 (4140)	1143 (13080)	432 (5396)
D4	***518 (7033)***	52 (630)	559 (8502)	382 (5813)	577 (9678)	352 (5905)	2557 (15317)	885 (5303)	683 (12720)	340 (5853)
D5	***665 (6965)***	32 (462)	733 (9669)	427 (5627)	673 (9271)	241 (3318)	1827 (10043)	1314 (7222)	777 (12200)	159 (2445)
Average	***712 (6921)***	81 (640)	1069 (10732)	511 (5456)	971 (10762)	452 (5022)	2444 (13153)	1002 (5416)	891 (11784)	427 (5556)

*M = Mean, SD = Standard Deviation, '( )' refers to the number of epochs required to achieve the best result.

The consistency in results was also better with a standard deviation of 640 epochs over all datasets for the SELECT method, compared to all the benchmarks having an average standard deviation above 5,000 epochs.

This is because the SELECT requires a set time for the training set to be completed for the CGRNN then the predictions are evaluated, which limits the variation in time taken to complete the optimisation procedure. The other approaches find the better performing models through observation of performance, leading to a larger uncertainty of the epochs required for each optimisation.

The SELET method also took less time than the benchmarks in achieving the best results on average for each dataset. Averaging over all datasets, the best result was achieved in 712s. When comparing the convergence of all benchmarks, GPBO and RS performed similarly in both the measurement of time and epochs, and TPE may have taken more epochs to converge over these two approaches, but it had a smaller epoch time, resulting in a better convergence time, averaged over all datasets. The HB algorithm had a significant disadvantage for the epoch quantity due to the nature of this algorithm. Because the trial followed two iterations through the HB algorithm, sometimes the best result occurred in the 2nd iteration, meaning that the total epochs before the best performance was significantly higher than the other algorithms. The HB model took significantly longer than the other benchmarks as the average time per epoch was larger, taking 2,444s as an average over all datasets. The HB approach would achieve a faster time if parallel training were to be included, but as this is not included in the study, HB performed worse than all other algorithms in relation to computational time.

As shown in Table 3-11, for D4: Behaviour of Urban Traffic and D5: Auto MPG respectively, the time taken by proposed HPO method is significantly closer to that of RS, TPE and GPBO. This is due to the variation in time per epoch for training the CGRNN model once the training set is created. Taking D4 as an example, the time taken for the different stages in the proposed approach are shown in Table 3-12.

Table 3-12 The required time in seconds for each stage of the SELECT method for dataset D4

	Time for Each Trial (s)						
SELECT Stages	**1**	**2**	**3**	**4**	**5**	**Mean**	**Range**
Tune the Learning Rate	21	12	12	11	13	14	10
Create the Training Set	279	267	229	246	265	257	50
Train the CGRNN	**132**	**124**	**121**	**123**	**131**	**126**	**11**
Predict Learning Curves	**54**	**53**	**46**	**54**	**53**	**52**	**8**
Best Result Achieved	99	102	58	41	45	69	61
Total Time/Best Result	585	558	466	475	507	518	78

The highlighted SELECT stages in Table 3-12, the training and prediction with the CGRNN, will always take approximately the same amount of time for every dataset. This is because the CGRNN trains with $train_{seq}$ which is created from each dataset and is the same size regardless of what the initial input data is, with 0.63s per epoch for 200 epochs to train the CGRNN with the created with $train_{seq}$.

With a smaller input dataset, such as 'D4', Behaviour of Urban Traffic, which has only 135 instance the CGRNN training time has a larger impact on the total optimisation time of the SELECT method. The TPE, GPBO and RS models have average epoch times of 0.054, 0.06 and 0.066 respectively, for all epochs during optimisation of 'D4'. A shorter time is only achieved with the SELECT method because less training iterations are required to find the best model.

It can also be seen from Table 3-12 that there is a high level of consistency in the time taken for each stage of the SELECT method, with the maximum accumulated range of computational time being 78s over 5 splits of D4. This aspect of the SELECT method can contribute to the overall effectiveness in guiding users to know and plan for allocated resource in HPO. The benchmark methods rely on a termination limit or function to end the HPO process without a guarantee of finding an optimum result without repeat comparisons. The SELECT method has a predefined allocation of resource to get to the solution.

The Scalability of the SELECT HPO Method

As the SELECT method does not require the results of each iteration to guide a surrogate function like BO based approaches, creating the learning curves for the CGRNN training set over the search space can be done completely in parallel. This capability for parallel training is highly significant as all learning curves independently contribute to the of the SELECT algorithm, there are not multiple steps which interact through searching the hyperparameter space. BO relies heavily on studying past configurations to guide the sequential process, and HB may be suitable for parallel training, but it still requires further steps to choose which of the configurations to continue allocating resources to. The mechanism in this study can have all the training and prediction learning curves produced simultaneously, with the only limitation being in computational capability rather than the algorithmic approach. This characteristic lends to high scalability with more hyperparameter dimensions.

3.7 Summary

In this chapter, a research gap related to HPO is highlighted. This gap helps to drive the development of a new HPO mechanism, the SEquential LEarning Curve Training (SELECT) method, which has a capacity for parallelisation, easy adaptability for new datasets, which could avoid the elimination of slow converging learning curves and still have a competitive level of performance compared to existing HPO methods. To the best of researcher's knowledge, there has never been an approach which can predict the full learning curves of unseen neural network hyperparameter configurations in learning curve prediction.

Being inspired by learning curve prediction, a new method of predicting unseen learning curves has been developed to predict complete learning curves without the need to train all the configurations. This SELECT method enables a CGRNN model to predict the best performance of all configurations of MLP neural networks with a single training window, treating the learning rate as a dependent function of the loss during the optimisation of each network architecture. The experimental results have shown that the SELECT method can outperform RS, HB, GPBO and TPE for prediction accuracy in MAE and computational cost in both seconds and epochs on well-known datasets, with a higher consistency in computational expense. The SELECT method has also shown to be able to generate useful insights into the search of better-performing architectures outside of the initial search space. It also lends itself to high scalability potential and the function of predefined resource allocation for finding optimum results.

Taking advantage of the SELECT HPO method, it will be applied to support feature importance analysis with an aim to identify Critical Success Factors (CSFs). Such analysis will be discussed in the following chapter.

4 Feature Importance Analysis

4.1 Introduction

With the help of the SELECT method, a better-tuned or optimised model can be used to identify the contribution of important features, also known as Critical Success Factors (CSFs) within the data and their intricate relationship with the performance metric. This study examines the integration of the optimised model by the SELECT method and three well-known feature importance algorithms to provide a holistic interpretation of feature importance.

This chapter first explains the three feature importance tools employed in this research; Shapley additive explanations (SHAP) (Lundberg and Lee, 2017), local interpretable model-agnostic explanations (LIME) (Ribeiro et al., 2016a) and Permutation Feature Importance (PFI) (Altmann et al., 2010), highlighting their capabilities and nuances. The impact of the SELECT method on the performance of feature importance analysis is then validated. The subsequent goal is to embed the SELECT method, and the three feature importance tools, as an integrated function, into the DSS, which will be discussed in the next chapter.

To affirm the efficacy and reliability of the SELECT method for feature importance, a rigorous evaluation has been undertaken. This involved two stages of analysis:

1. A comparison of the SELECT method against the same established benchmark HPO algorithms from Chapter 3 for feature importance analysis.

2. A comparison of the three selected feature importance algorithms using models optimised by the SELECT method.

The validation process extends to both synthetic and real-world datasets, encompassing scenarios with complex and non-linear relationships between features and performance metrics. Through this comprehensive validation, the robustness and generalisability of the SELECT method is put to the test, paving the way for its integration into the real-world decision-making system.

This chapter will be structured as follows:

- An overview of the three feature importance algorithms will be given.
- Stage 1 and stage 2 of feature importance experiments will be explained.
- The results of the experiments will be presented and discussed.

4.2 Selected Feature Importance Analysis Tools

To obtain an accurate understanding of the interrelationship between important features and key performance metrics, feature importance tools are useful to extract such information from an optimised or fine-tuned neural network model, which is often regarded as a black box.

SHAP, LIME and PFI have been used in previous research in combination for feature importance analysis (Kuzlu et al., 2020). These tools have been deemed useful for interpreting relationships and actioning insights from deep learning applications and areas of construction (De Bock et al., 2023, Love et al., 2023). SHAP and LIME have been frequently utilised owing to their adaptability and the consideration of both local and global feature importance analysis (Chen et al., 2023, Machlev et al., 2022) while PFI has shown to have similar level of capability to SHAP for feature selection (Effrosynidis and Arampatzis, 2021). Each of these approaches to feature importance analysis will be explained next, followed by an explanation of their advantages and disadvantages.

4.2.1 Shapley Additive Explanations (SHAP)

SHAP is a method of providing insights into the contribution of individual features to the predictions of a ML model. It is rooted in cooperative game theory and utilises Shapley values, a concept that originated from fair distribution in cooperative games (Lundberg and Lee, 2017).

In the context of ML, SHAP can generate Shapley values accurately depicting the impact of each feature on a model's predictions among all the features(Marcílio and Eler, 2020). Its primary mechanism is to consider all possible combinations of features and calculate the average contribution of each feature, ensuring that each feature is given a fair share of the credit for the model's output. The calculation of the Shapley values is based on determining what the importance of each feature’s contribution is to the overall prediction through averaging all subset combinations of features within a group.

4.2.2 Local Interpretable Model-agnostic Explanations (LIME)

LIME is a method designed to offer insights into the contribution of individual features to the predictions of ML models, particularly at the local level. Unlike SHAP, LIME focuses on creating local and interpretable explanations for specific instances rather than considering the entire dataset. It can be applied to any ML models without requiring knowledge of its internal structure. LIME values the idea of creating simplified and interpretable "local models" that approximate the behaviour of the complex, black-box model for a specific instance. Its primary mechanism involves generating perturbations or slightly modified versions of the input data and observing how the model's predictions change (Ribeiro et al., 2016b). By fitting a simple, interpretable model to these perturbed instances, LIME can provide an understanding of feature contributions for a particular prediction. The key focus of LIME is to locally approximate the complex model's decision boundary, offering a simplified view that is easier to interpret. This local interpretability is valuable for understanding why a model made a specific prediction for a given instance, even when the overall model may be intricate and challenging to interpret.

4.2.3 Permutation Feature Importance (PFI)

PFI is a method designed to uncover the impact of individual features in influencing the predictions of ML models. Like LIME, it can be applied to any ML algorithms regardless of its internal structure. PFI is particularly valuable for assessing the impact of features on a global scale, offering insights into their overall importance for the model. The method involves systematically permuting or shuffling the values of a single feature and keeping the other features unchanged while making predictions. By doing so, the importance of the shuffled feature is assessed by observing the change in the model's performance metrics. The greater the impact on the model's performance when a feature is randomly shuffled, the more important that feature is (Altmann et al., 2010). PFI provides a straightforward and intuitive measure of feature importance. Features that, when permuted, lead to a significant drop in model performance are deemed highly important, while those with minimal impact are considered to have lesser importance. This method is particularly useful for gaining a global understanding of feature importance, helping to identify key drivers in the overall predictive capacity of the model. The advantage of PFI lies in its simplicity and effectiveness in evaluating feature importance without relying on intricate model-specific details. It offers a practical and widely applicable means of assessing the contribution of individual features to the predictive power of ML models.

4.2.4 Strengths and Weaknesses for the Feature Importance Methods

Each of these feature importance algorithms have their own strengths and weaknesses, as shown in Table 4-1. The combination of the three feature importance methods; SHAP, LIME, and PFI, reflects an effective yet diversified approach of understanding the intricacies within prediction models. Recent research has also shown to use multiple feature importance methods for a holistic understanding of the important factors in decision support (Khanna et al., 2023, Settouti and Saidi, 2024). The use of multiple methods can provide a greater scope of the important factor related to model performance.

SHAP offers both global and local interpretability, underpinned by a solid mathematical foundation, making it effective for exploring feature interactions and unbiased feature evaluation. However, its computational complexity and sensitivity to data shape pose challenges, especially in high-dimensional spaces. LIME excels in local interpretability, simplifying the analysis of complex models and showcasing versatility across varying data types and sizes. Despite its proficiency in capturing non-linear feature relationships, LIME may struggle to evaluate global trends and can be sensitive to data perturbations. On the other hand, PFI, with its simplicity and fast computation, provides robust insights into global trends while exhibiting resilience through duplicate assessments. Although PFI has limitations in uncovering feature interactions and determining correlation direction, its speed and simplicity make it a valuable addition.

Table 4-1 Strengths and weaknesses of the selected feature importance methods; SHAP, LIME and PFI

Method	Abbreviation	Strengths	Weaknesses
Shapley Additive Explanations	SHAP	• Global and local interpretability • Solid mathematical grounding • Thorough examination of feature interaction • Unbiased evaluation of each feature • Visualise correlation direction and magnitude • Effective for non-linear feature relationships	• Computationally complex • Sensitivity to data shape • Difficult interpretability for high dimensionality
Local Interpretable Model-agnostic Explanations	LIME	• Local Interpretability focus • Simplifies complex models in analysis • Visualise correlation direction and magnitude • Versatile over varying data types and sizes • Effective for non-linear feature relationships	• May not evaluate global trends • Sensitive to data perturbations
Permutation Feature Importance	PFI	• Simple computation and fast • Global trend analysis • Robustness through duplicate assessments	• Limited feature interaction insight • Cannot determine correlation direction.

Integrating a neural network optimised with the SELECT method with all three of SHAP, LIME, and PFI, into the DSS leverages the strengths of each method while mitigating their respective weaknesses. This holistic approach ensures a more nuanced and reliable assessment of feature importance, catering to diverse aspects of model interpretability and accommodating various data characteristics.

4.3 Validation Method of Feature Importance Tool

The validation of the SELECT method for feature importance covers two stages of experiments. The two stages can be recalled as follows:

- **The first stage** is the comparison between the feature importance performance of the neural network model optimised by the four benchmarking HPO methods (TPE, HB, RS and GPBO) against that of the model optimised by the SELECT method; and
- **The second stage** is to examine the performance of the SELECT method combined with each of the three selected feature importance tools, SHAP, LIME and PFI.

The validation will be discussed by first explaining the chosen datasets for feature importance and the rationale for their selections. This will be followed by an explanation of the experimental setup for each stage, and then the results and discussion.

4.3.1 Datasets Selection

The challenge in utilising real-world datasets for feature importance analysis lies in the absence of a ground truth level of understanding for feature interactions; a ground truth being the true feature importance, rather than the interpretation of importance which can be skewed by uncertainty and analysis characteristics.

A definitive advantage of using synthetic datasets for feature importance is the ability to control the experiment characteristics such as the quantity of irrelevant features, the instances of data or the noise levels. Additionally, working with the known optimal features allows for a high level in confidence in the results (Bolón-Canedo et al., 2013).

Comparing feature importance solely against the performance and interpretations of previous studies limits the ability to establish a definitive benchmark. As highlighted in Table 4-1, each feature importance algorithm has its own set of strengths and weaknesses, a pattern observed across various areas of model interpretability methods.

Recognising the nuances inherent in different feature importance tools, this study seeks a robust evaluation strategy. To ensure a comprehensive understanding of the SELECT method's performance, the feature importance experiments will encompass both synthetic and real-world datasets, like approaches from previous studies on feature importance (Rudnicki et al., 2015, Zeng et al., 2015). The inclusion of synthetic datasets allows for a controlled exploration of the methodologies, while real-world datasets introduce complexities of those encountered in practical applications. This dual approach aims to provide a nuanced and holistic assessment of the feature importance methods, considering their adaptability across varying data types and scenarios.

The utilisation of synthetically created datasets offers distinct advantages in the context of feature importance analysis. Notably, the relationships between features within synthetic datasets are predefined, providing a clear and known ground truth for feature interactions. This inherent understanding of feature relationships serves as a valuable reference point, enabling a more precise evaluation of how well feature importance methods capture and interpret these known interactions. (Bolón-Canedo et al., 2013).

Synthetic datasets provide a controlled environment for experimentation. The flexibility to manipulate the number of features, instances, and the level of noise allows for systematic exploration of how feature analysis methodologies adapt to variations in dataset characteristics. By intentionally introducing variations in synthetic datasets, a deeper understanding can be gained of the methodologies' performance across a spectrum of dataset complexities.

Real world datasets capture the true complexities of actual data. This includes uneven or non-standard distribution, outliers in the data and natural noise levels. Evaluating the feature importance tools against real world datasets provides an authentic evaluation of performance. Regarding the evaluation, this would be compared to the findings of other research in the results section of this chapter.

The chosen datasets for both Stage 1 and Stage 2 of the experiments are shown in Table 4-2, with the number of features used in each dataset, the instances (rows) and the range of noise levels (uncertainty) generated for testing. The datasets shown in Table 4-2 will be further explained in the sub-sections ahead.

Table 4-2 Datasets for feature importance validation

Stage	Dataset	Reference	Nature	Features	Instances	Noise levels
1st: HPO Method Comparison	make_friedman1	(Pedregosa et al., 2011)	Synthetic	10	500	0%,5%,10%,20%
					1,000	0%,5%,10%,20%
				20	500	0%,5%,10%,20%
	make_friedman2		Synthetic	4	500	0%,5%,10%,20%
2nd: Feature Importance Validation	make_friedman1	(Pedregosa et al., 2011)	Synthetic	10	500	0%,5%,10%,20%
					1000	0%,5%,10%,20%
				20	500	0%,5%,10%,20%
	make_friedman2		Synthetic	4	500	0%,5%,10%,20%
	Concrete Compressive Strength	(Geifman and El-Yaniv, 2019, Muliauwan et al., 2020, Asteris et al., 2021)	Real	8	1030	N/A
	Boston Housing	(Oh, 2019, Adetunji et al., 2022, Calvo-Pardo et al., 2023)	Real	13	506	N/A

4.3.2 Datasets for Stage 1- HPO Method Comparison

The purpose of the stage 1 is to investigate the performance of different HPO approaches in determining the relationships between input features and the target prediction variable in data. Synthetic datasets, namely **make_friedman1** and **make_friedman2** from the scikit-learn python library, were deliberately chosen to create controlled environments with predefined relationships between the features and the target prediction variable. Each dataset is originally created in (Friedman, 1991) and is representative of complex relationships between the prediction and the input features, each specifically used for benchmarking ML model performance, and for feature analysis. Each of the selected datasets will be discussed next.

Make_Friedman1

The make_friedman1 dataset was originally developed for the purpose of avoiding finding relationship structures when they are not there, and to ensure they are found when they are (Friedman, 1991). Implemented as a Python package in the scikit-learn library, it serves as a valuable tool for assessing the capabilities of algorithms in handling non-linear feature relationships (Pedregosa et al., 2011). This dataset is particularly useful for evaluating the robustness of algorithms in handling diverse feature relationships, including non-linear components, and the impact of noise on predictive accuracy. It has been used in many studies as a benchmark for feature importance and selection (Breiman, 1996, Bugata and Drotár, 2023).

The dataset generation involves the calculation of labels (y) for each instance using Equation 10. This equation incorporates five variables ranging between 0 and 1, including co-dependent and cyclical features (x_1, x_2), a non-linear feature (x_3), and two linear features with variations in weighted magnitude (x_4, x_5). Equation 10 also incorporates a standard deviation of the gaussian noise to introduce variability in the data (ε), ranging between 0 and 1. Users have the flexibility to specify the number of instances, features, and the level of noise in the dataset. Notably,

- Increasing the number of features adds unconnected variables, thereby challenging algorithms to discern the true relationships defined by Equation 10.

- The noise level introduced uncertainty in a normally distributed range, controlled by user-defined parameters, provides a dynamic range of complexity for algorithm analysis.

$$y = 10\sin(\pi x_1 x_2) + 20(x_3 - \tfrac{1}{2})^2 + 10x_4 + 5x_5 + \epsilon \tag{10}$$

Make_Friedman2

The make_friedman2 dataset is taken from the same source material as make_friedman1 (Friedman, 1991) and utilises a different complex relationship between the label (y) and input variables. Equation 11 is created from the calculation of impedance (Z) for a simple alternating current series circuit as illustrated in Figure 4-1.

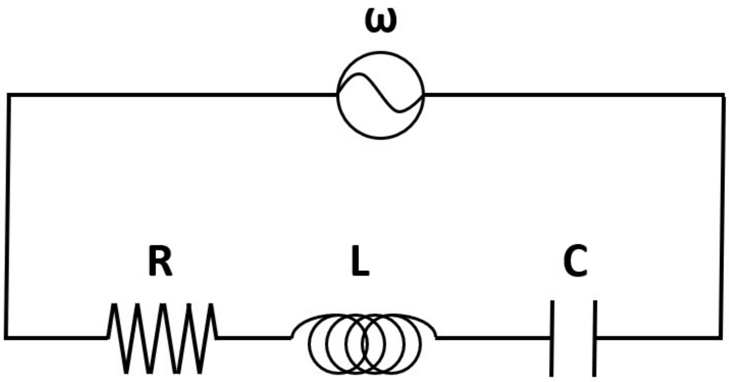


Figure 4-1 Simple alternating current series circuit and equation for impedance (Z)

$$Z(R, \omega, L, C) = \sqrt{(R^2 + (\omega L - \frac{1}{\omega C})^2}$$

(1)

Equation 11 has a highly interconnected relationship between the individual contributions of each of the variables, such that the selected range of each variable can alter the ranked importance significantly. The developed make_friedman2 equation is created from Equation 11, incorporating the variable ranges shown below to produce Equation 12.

$$0 \leq x_1 \leq 100$$

$$40\pi \leq x_2 \leq 560\pi$$

$$0 \leq x_3 \leq 1$$

$$1 \leq x_4 \leq 11$$

$$y = \sqrt{(x_1{}^2 + (x_2 x_3 - \frac{1}{x_2 x_4})^2} + \epsilon$$

(2)

Equation 12 captures the complexity of the impedance calculation while introducing variability as the standard deviation of gaussian noise (ε). It does not contain any unconnected variables for analysis, but the complexity and interconnected relationship is higher than that of make_friedman1 and will challenge the optimised model to detect this sophisticated relationship to achieve high predictive accuracy and evaluate the true feature importance. This is the second of the Friedman benchmark datasets, used in previous feature selection and importance analyses(Tipping, 2001, Kamalov, 2021, Granitto et al., 2005).

Exploration Overview and Dataset Variation

As shown in Table 4-2, both the choice of experiments and dataset variations are determined to offer a thorough assessment of the HPO methods in the context of feature importance analysis.

Beginning with **make_friedman1**, the variations in dataset are explained below.

- The inclusion of both relevant and unconnected features aims to evaluate the models' capacity to differentiate between meaningful and unrelated elements. This is evaluated by testing with both 5 and 15 additional unconnected variables from Equation 10. The 5 additional features are the default number while the 15 is to provide added difficulty.
- Further exploration at varying dataset sizes 500 and 1,000 instances provides insights into the potential influence of data scale on model performance.
- Each of these analyses covers a variation of noise levels ranging from no noise (ε=0) through lower levels of noise (ε=0. 5 and ε= 1) up to high levels of noise (ε=2), as has been defined in previous research (Granitto et al., 2005). This equates to a range of error of approximately 0%, 5%, 10% and 20% of the final y values with no noise when checked on both the 500 and 1,000 instance datasets. The addition of the increasing noise levels examines how well models handle uncertainty, aligning with real-world scenarios where data can be inherently noisy.

Extending the analysis to include **make_friedman2**,

- Characterised by complex interactions among features, adds further depth to the evaluation, tested with 500 instances.
- The same percentage of noise range for uncertainty as friedman1 is tested, 0%, 5%, 10% and 20% of the range of error of the final y values with no noise.

4.3.3 Datasets for Stage 2- Feature Importance Method Validation

The second stage of the experiment is to validate the SELECT method in conjunction with the chosen feature importance algorithms. The key objective is to scrutinise performance variations and interpretations across three chosen feature importance tools, namely SHAP, PFI, and LIME, when coupled with models optimised by the SELECT method.

This assessment seeks to provide comprehensive insights into the combined efficacy and interpretability of these feature importance methodologies. To facilitate these evaluations, both synthetic datasets, **make_friedman1** and **make_friedman2**, will be examined under the same variations in noise levels, covering 0% noise for a ground truth interpretation, increasing the noise to 5%, 10% and 20% for low to high uncertainty. In addition to the synthetic datasets, the validation process also incorporates real-world datasets to authentically represent the potential impact of combining the three feature importance tools. To achieve this purpose, two specific real-world datasets **Boston housing** and **concrete compressive strength**, are chosen and they will be described next.

Boston Housing Dataset

The Boston Housing dataset is a widely utilised dataset in ML and statistics (Oh, 2019, Adetunji et al., 2022, Calvo-Pardo et al., 2023), consisting of 506 instances. Each instance represents various properties of houses in Boston suburbs, encompassing 13 features. These features include *the per capita crime rate, average number of rooms per dwelling, nitric oxides concentration, and others*. The target prediction variable is *the median value of owner-occupied homes (MEDV)*. The dataset is popular for regression tasks, serving as a benchmark for assessing the predictive capabilities of various algorithms in the field.

Concrete Compressive Strength Dataset

The Concrete Compressive Strength dataset is designed to evaluate the compressive strength of concrete and comprises 1,030 instances. Each instance represents a concrete mix with various ingredients, including *cement, blast furnace slag, fly ash, water, superplasticiser, coarse aggregate, and fine aggregate*. The target prediction variable is *the compressive strength of the concrete*. It is frequently employed to assess the performance of predictive models in estimating the strength of concrete based on its composition (Geifman and El-Yaniv, 2019, Muliauwan et al., 2020, Asteris et al., 2021).

Integrating these real-world datasets into the second stage of the experiment helps to validate the SELECT method under conditions that mimic real-world situations and enable fair comparisons against other studies using the same datasets.

4.3.4 Experimental Setup for Stage 1- HPO Method Comparison

The experimental setup for the comparison of the SELECT method against the benchmark algorithms for feature analysis is expressed in Figure 4-2. This process will repeat for all variations in dataset stated in Table 4-2 for the HPO method comparison, and for a 5-fold split of each variation of the synthetic datasets.

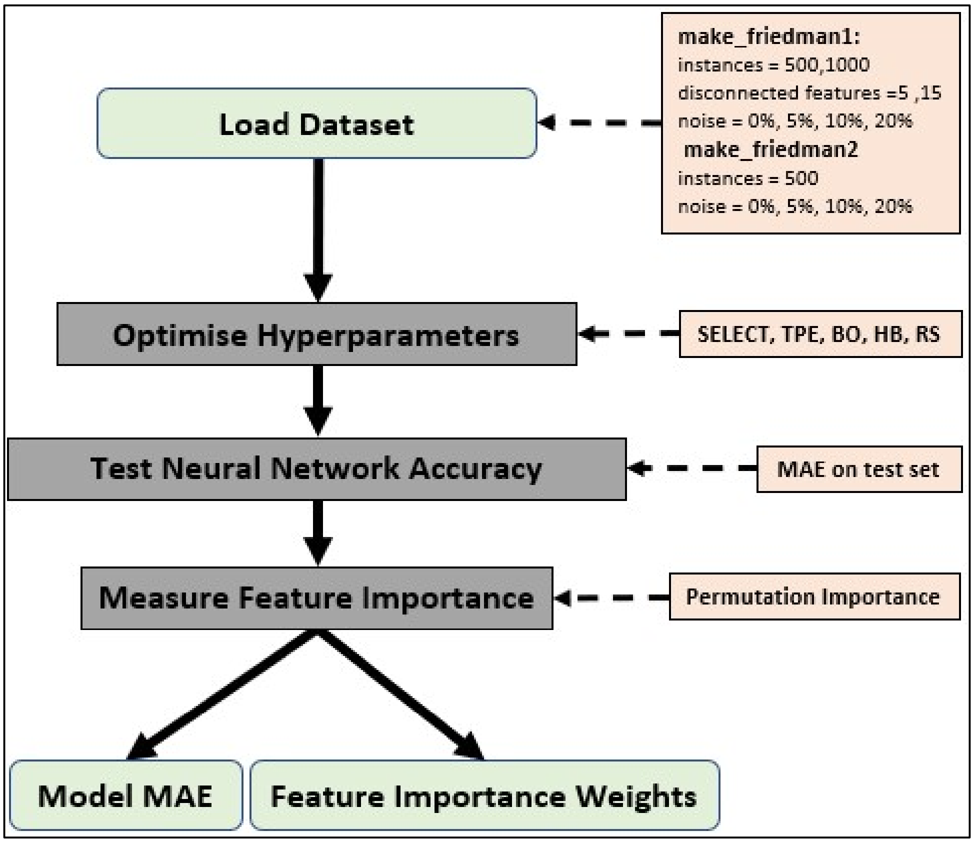


Figure 4-2 Experimental setup of Stage 1

HPO Method Configuration

The evaluation process begins with the selection of the best-performing neural network configuration for each HPO method, following the procedures detailed in section 3.6.3. Optimal configurations are chosen and predictions are made on the test set to assess the model prediction accuracy, utilising the Mean Absolute Error (MAE) as the performance metric. This ensures a common ground for evaluating prediction accuracy across all HPO approaches through all experiments and variations in dataset. As it was determined that the best performance of the benchmark algorithms was achieved with an epoch limit of 200, rather than the extended limit, this epoch limit is used for benchmark algorithms in all experiments in Stage 1.

Choice of Feature Importance Tool

With the selected best performing model, feature importance analysis using PFI is carried out. PFI is chosen as the preferred method for evaluating different HPO approaches primarily due to its robust global evaluation and simplicity, for repeatable experiments with neural networks, with highly similar performance to SHAP(Chen et al., 2024, Mandler and Weigand, 2023).

Feature Importance Assessment Method

Using PFI, features will be systematically analysed and ranked based on their impact on model performance measured by MAE. This is achieved through determining the accuracy of the model in MAE, then shuffling a feature and measuring the difference in accuracy; the larger impact on MAE suggests higher importance. The shuffling is random so the PFI method will repeatedly record the importance weight of the features 10 times and the average weight will be calculated for all features with every optimised model to reduce bias from the randomness in the perturbations in PFI while achieving computational efficiency without parallelisation (Altmann et al., 2010).

To enhance the robustness of the stage 1 evaluation, the entire process, highlighted in Figure 4-2, from HPO for the optimum prediction model to feature importance analysis, is repeated using a 5-fold cross-validation approach (Parr et al., 2024). The MAE values for the optimised model accuracy, and the feature importance ranked results will be recorded for each split of the dataset. The model accuracy and feature rankings will then be averaged over all 5 splits to present the result in both predictive accuracy and the feature rankings.

4.3.5 Experimental Setup for Stage 2- Feature Importance Validation

As highlighted earlier in this chapter, each feature importance tool has its own strengths and weaknesses. The development of the DSS will include all three well-known feature importance tools, SHAP, PFI, and LIME, to offer multiple ways of evaluating the feature importance within the same data. As a further assessment to the validity of SELECT method, the integration of the SELECT optimised ANN model and each of the feature importance tools will be tested and compared to ensure efficacy as well as consistency in performance. Figure 4-3 shows experimental setup of Stage 2 to produce the feature importance weights using each of the three methods.

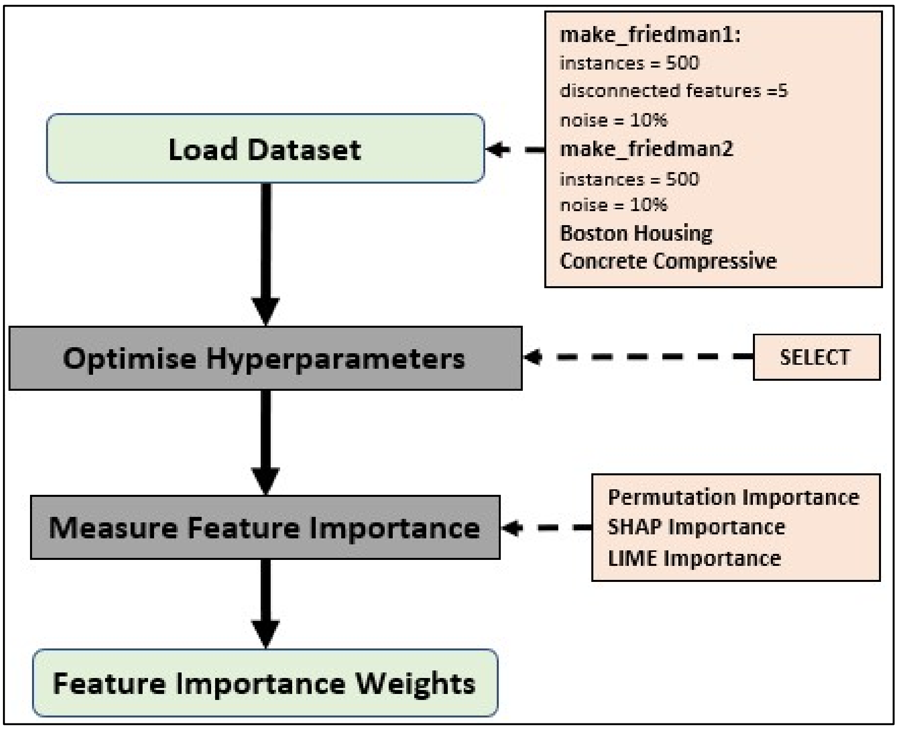


Figure 4-3 Experimental setup of Stage 2.

SELECT Method Configuration

The SELECT method will be set up in the same way as presented in Stage 1 to ensure consistency. The selected configuration will be trained once, and the same optimised model will be used for all feature importance tools to allow fair comparisons.

Feature Importance Configuration and Repetition

As each of the three feature importance methods employs a different mechanism of measuring the importance of features, it is essential to define a common ground. For both SHAP and LIME, the local importance determined for each instance is accumulated to determine a global importance for each feature. The PFI utilises the MAE metric to obtain an absolute global measurement of each feature's impact on model performance. Having the global importance from each method, the relative importance of each feature can then be calculated using Equation 13. The relative importance, as a common ground, can be used as the metric to compare three different feature importance tools. All three feature importance tools have undergone testing using a 5-fold split and the average importance for each feature is calculated across each tool.

$I_{relative}$ = Relative importance of the i-th feature.

I_i = Importance value of the i-th feature.

$\sum_{j=1}^{n} I_j$ = Sum of all feature values

$$I_{relative} = \frac{I_i}{\sum_{j=1}^{n} I_j} \tag{13}$$

4.4 Results and Discussion

To recall, Stage 1 aims to compare the predictive performance and feature importance performance between the SELECT method and four benchmarking HPO algorithms using ONLY synthetic datasets. Such comparison encompasses the MAE to define predictive performance across all experiments, and the use of PFI to define the feature importance performance. Using the same SELECT method, Stage 2 aims to compare feature importance measurement across three well-known methods: SHAP, LIME, and PFI using BOTH synthetic and real-world datasets.

4.4.1 Stage 1- HPO Method Comparison

This section is broken down into the evaluation of each model's predictive accuracy, using MAE as a metric, followed by the comparison of the feature importance using PFI. The evaluation involves the use of two synthetic datasets, **make_friedman1** and **make_friedman2**.

Stage 1 MAE Performance

The average MAE results over the 5-fold cross-validations for all variations of the synthetic datasets for the SELECT method and benchmark algorithms are shown in Table 4-3.

Table 4-3 Average of the MAE for all HPO methods averaged over a 5-fold split

Dataset	Instances	Features	HPO Method	0%	5%	10%	20%
make_friedman 1	500	10	SELECT	0.36	0.76	1.26	2.21
			TPE	0.69 (+94%)	1.3 (+69%)	1.5 (+19%)	2.46 (+11%)
			GPBO	1.37 (+284%)	1.55 (+102%)	1.84 (+46%)	2.47 (+12%)
			HB	1.01 (+184%)	1.18 (+55%)	1.71 (+35%)	2.53 (+15%)
			RS	0.96 (+169%)	1.2 (+57%)	1.65 (+31%)	2.57 (+17%)
		20	SELECT	0.87	1.31	1.73	2.62
			TPE	1.58 (+81%)	1.74 (+33%)	1.97 (+14%)	2.93 (+12%)
			GPBO	1.82 (+109%)	1.94 (+48%)	2.27 (+31%)	3.14 (+20%)
			HB	1.9 (+118%)	1.98 (+51%)	2.12 (+23%)	3.02 (+15%)
			RS	1.85 (+113%)	2.06 (+57%)	2.1 (+22%)	3.17 (+21%)
	1000	10	SELECT	0.25	0.56	1.04	1.96
			TPE	0.3 (+21%)	0.77 (+37%)	1.32 (+28%)	2.2 (+12%)
			GPBO	0.8 (+221%)	0.8 (+44%)	1.21 (+17%)	2.28 (+16%)
			HB	0.46 (+86%)	0.8 (+43%)	1.19 (+15%)	2.19 (+12%)
			RS	0.48 (+94%)	0.79 (+42%)	1.23 (+19%)	2.26 (+15%)
make_friedman 2	500	4	SELECT	8.03	24.59	48.69	92.42
			TPE	13.76 (+71%)	37.1 (+51%)	64.33 (+32%)	100.73 (+9%)
			GPBO	14.37 (+79%)	75.47 (+207%)	49.92 (+3%)	104.87 (+13%)
			HB	11.65 (+45%)	29.97 (+22%)	51.64 (+6%)	96.5 (+4%)
			RS	12.8 (+59%)	31.08 (+26%)	51.44 (+6%)	92.64 (+0.2%)

Note: '()' refers to the percentage difference between the benchmark metric and the CGRNN result.

MAE Performance: Make_friedman1(instances=500, features=10)

Looking at the **make_friedman1**(500,10) with 0% noise level(5 features are disconnected), the model optimised by the SELECT, as the datum model, has a MAE of 0.36 with the nearest accuracy being TPE optimised model of 0.69 (94% more than the SELECT at 0.36). The GPBO optimised model is the worst performing among all benchmarks, achieving an average MAE of 1.3 (284% higher than SELECT or +284%). The performance gaps among all models becomes smaller when the noise level increases. With no noise, the benchmark HPO optimised models achieved an MAE which was 94% and 284% higher than the SELECT method. At 20% noise level the SELECT method outperformed the benchmark HPO methods by only 11% to 17%.

As the noise level increases, the achievable accuracy reduces for all optimisation algorithms, even if one is performing better than the others. The SELECT method still achieves a higher accuracy but the achievable result for all methods gets impacted by the increasing noise. As the SELECT method has the highest accuracy at lower noise levels, it will receive a higher relative impact with more uncertainty than the other algorithms. This can also be seen by the fact that MAE increases with noise level for all experiments, as shown in Figure 4-4.

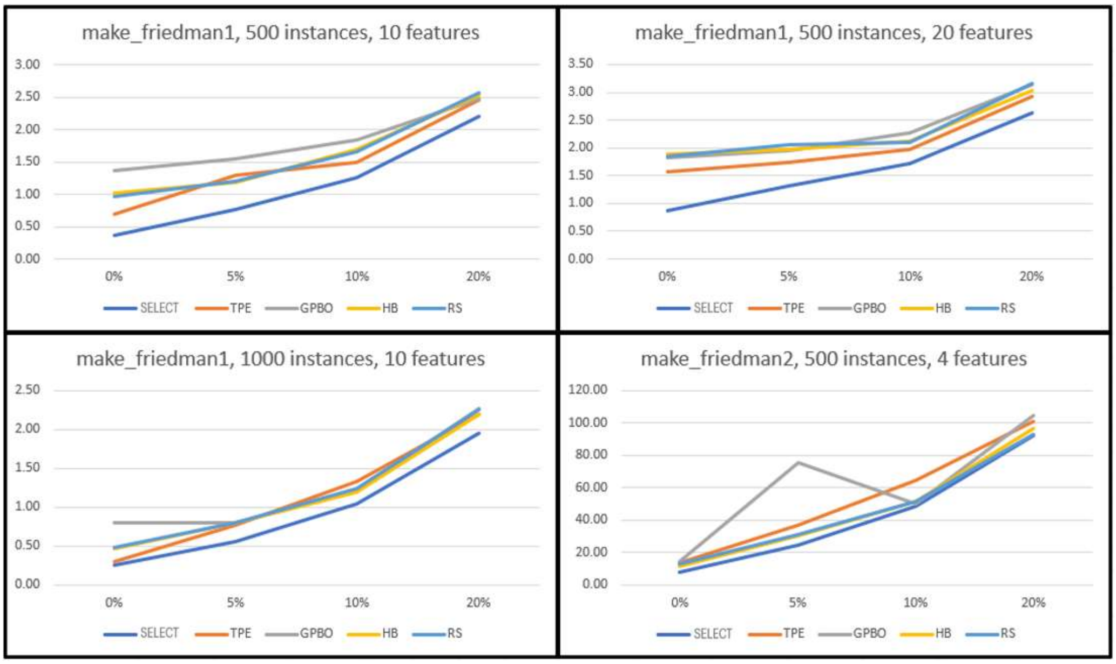


Figure 4-4 Average MAE vs Noise level, for all variations of synthetic dataset

MAE Performance: Make_friedman1(500,20)

When increasing the number of features to 20 (15 are disconnected features), MAE increases for all noise levels and all HPO algorithms. This suggests that the increasing number of disconnected features negatively impacts a model's capability to detect relationships and hence, decrease predictive accuracy. This matches a similar trend seen in previous research with ANN predictive accuracy reducing with increasing disconnected feature quantities (Vecoven et al., 2020, Rengasamy et al., 2021).

MAE Performance: make_friedman1(1000,10)

On the other hand, Table 4-3 shows that increasing the number of instances of the dataset to 1,000 while keeping the same features (10) helps to improve predictive accuracy over all variations in noise as well as all HPO algorithms. One reason is that larger instances enable the models to learn from more data, generating a better understanding of the relationships within the dataset. This matches a trend shown in (Kramer, 2016) where the performance of the prediction mode improves with the increasing number of instances. The same reference also agrees with the earlier observation that performance decreases with increasing feature quantities.

MAE Performance: Make_Friedman2

Once again, with the **make_friedman2** dataset, the SELECT method consistently found better performing models by returning lower MAE results than other benchmark HPO algorithms. At 0% noise level, the models optimised by benchmark HPO methods achieved a MAE that was 45% to 79% higher than the models optimised by the SELECT method. This range of difference between the SELECT method and the benchmarks decreased as the noise level increased to 20%, an observation that can be noted with the **make_friedman1** dataset as shown in Figure 4-4.

As with **make_friedman1**, the MAE increases with all HPO models in a steady trend with the increase of noise from 0% to 20%. There is a noticeably high MAE for GPBO at 5% noise. This is due to a single result from the 5-fold split of GPBO achieving a MAE of 250.3, well above all other results for all other HPO algorithms at 5% noise with the **make_friedman2** dataset. Excluding this single result, the GPBO average MAE at 5% would be 31.8; more in line with the trend seen with all other HPO algorithms.

Absolute Feature Importance: Comparison of HPO Algorithms

After proving the predictive accuracy and robustness of the SELECT method with MAE performance, the next step of Stage 1 is to evaluate how well the model optimised by

SELECT can interpret the importance of key features when compared against the model optimised by the benchmark HPO algorithms.

The absolute feature importance results were obtained using the PFI method over a 5-fold split for both the SELECT method and the benchmark algorithms. The comparisons encompass both the **make_friedman1** and **make_friedman2** datasets, with variations in instances and features. Throughout these experiments, the noise level ranges from 0% to 20%, introducing varying degrees of uncertainty. All experimental results of this section will be presented and discussed with the help of PowerBI software.

Absolute Feature Importance: Make_friedman1(500,10)

Figure 4-5 shows the feature importance from the SELECT model and the benchmark HPO algorithms. All algorithms can distinguish clearly the 5 connected variables, X1-X5, from the disconnected variables. Also, there is a clear and consistent ranking of feature importance descending from X4, X2, X1, X5 and X3, generated by all approaches. This ranking agrees with previous literature on the true ranking of the relevant features, termed as the "Friedman 1 Benchmark" (Greenwell, 2022). This has also been covered thoroughly by (Greenwell et al., 2020), who used the Friedman 1 benchmark with multiple machine learning models, and feature importance methods, including the combination of ANNs and PFI. Nearly all results pointed to the same ranking, with some variation in determining the equivalent ranking of X5 and X3. This shows that all HPO methods, combined with PFI, successfully assign importance in alignment with previous research.

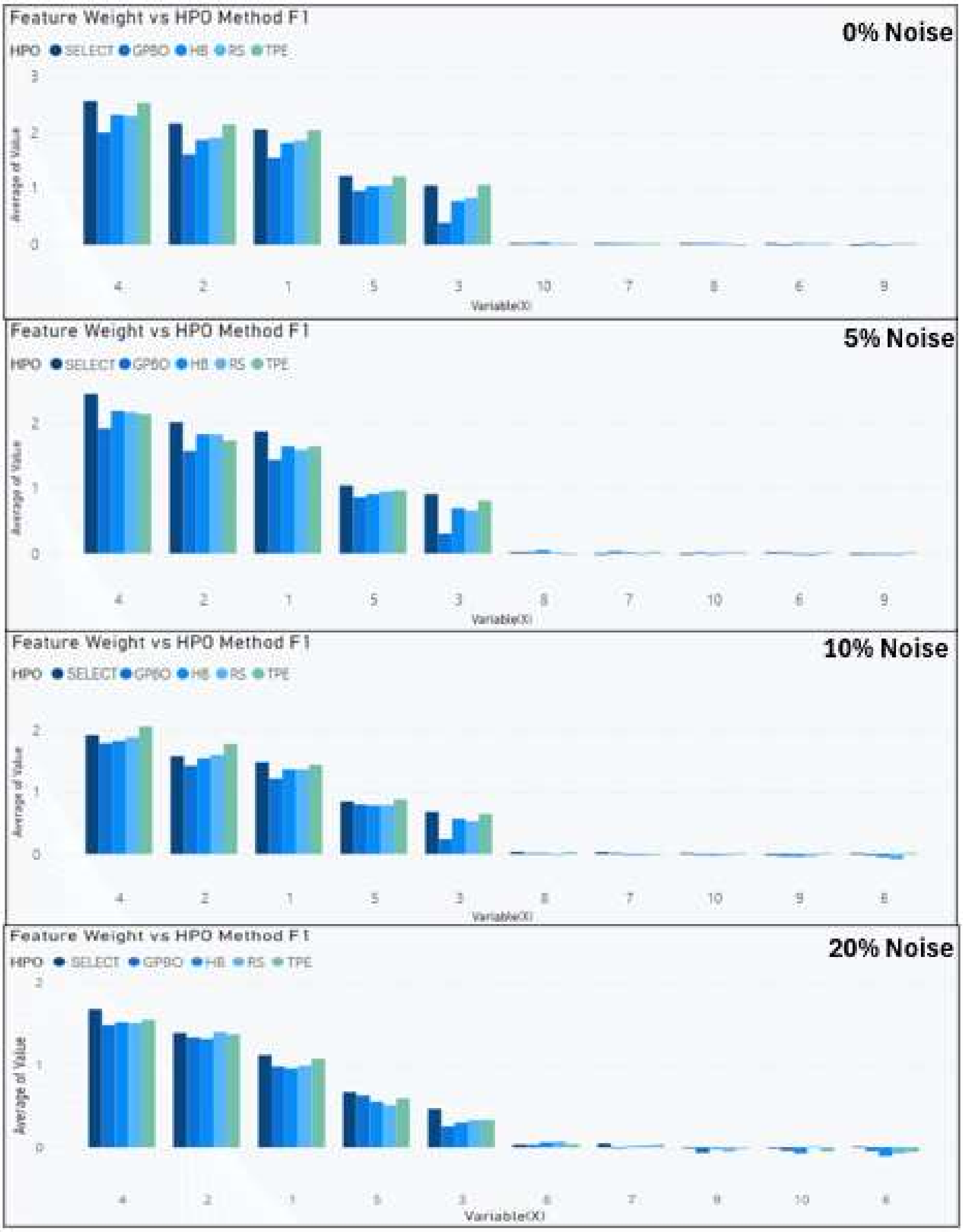


Figure 4-5 HPO Feature Importance, make_friedman1, 500 instances, 10 features, 0% -20% noise.

Figure 4-5 clearly shows that the SELECT method has assigned a higher importance to the relevant features over the benchmark HPO methods at 0%,5% and 20% noise levels respectively. Compared to the SELECT method, TPE has assigned a slightly higher importance at 10% noise to features X4 and X2, while achieving the same at X5 and less at X3 and X1. Despite the similar importance ranking to TPE, the SELECT method managed to achieve a better accuracy of MAE = 1.26 (MAE of TPE is 1.5), as shown in Table 4-3. This suggests that the feature importance assigned by the SELECT method is a more accurate interpretation of CSFs within the dataset.

Absolute Feature Importance: Make_friedman1 (1000,10)

Figure 4-6 shows the feature importance ranking of make_friedman1 with 1000 instances and 10 features. Again, the SELECT method has assigned a higher importance to the features for all noise levels. At 0% noise level, TPE achieves a similar level of performance but then progressively performs worse as compared to the SELECT method when noise levels increase. HB, on the other hand, improves comparatively as noise levels are increased. GPBO is the overall worst of all HPO methods, most notably failing to identify the significance of X3 at 20% noise level.

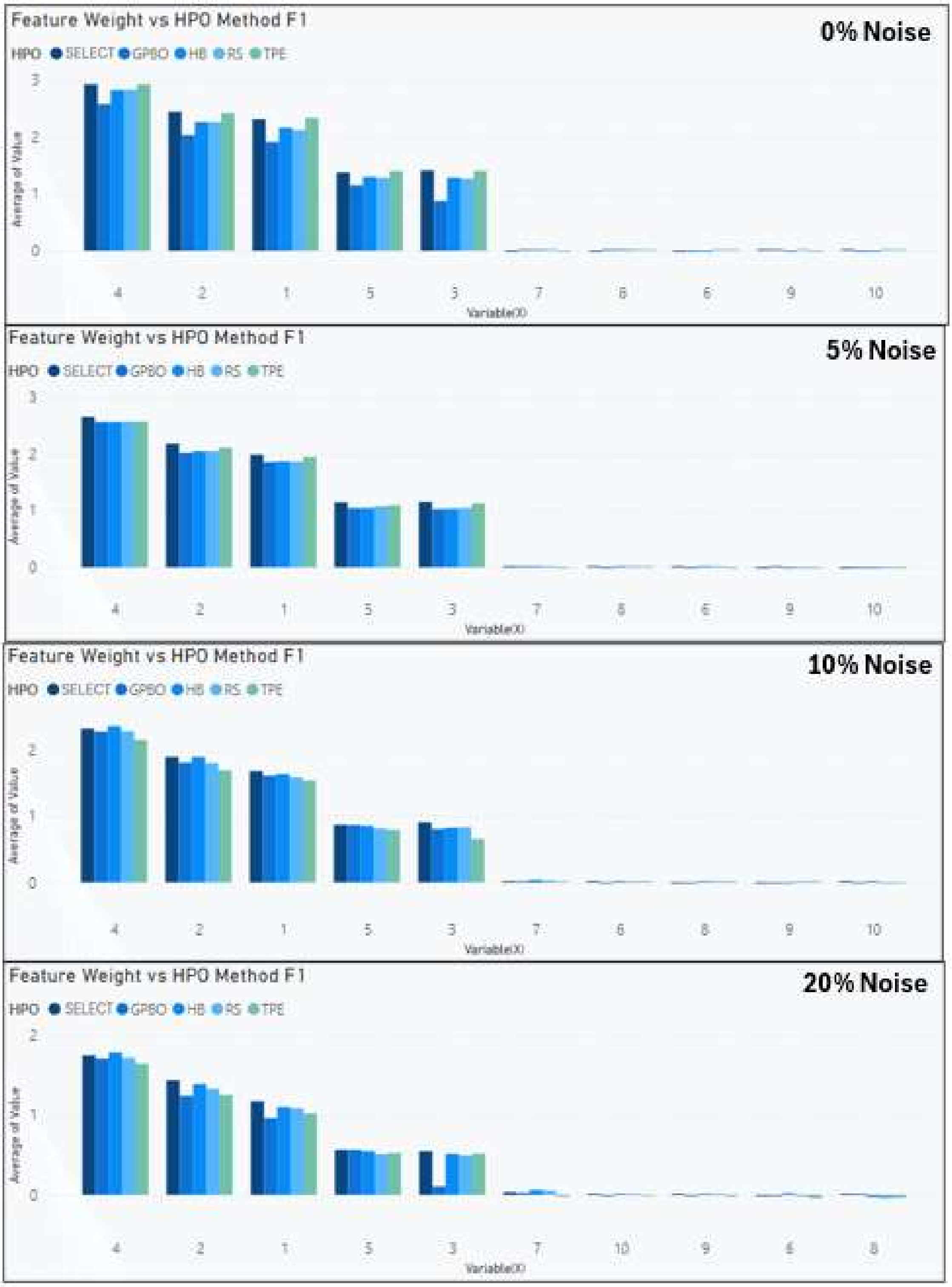


Figure 4-6 HPO Feature Importance, make_friedman1, 1000 instances, 10 features

Absolute Feature Importance: Make_friedman1(500,20)

Increasing the feature quantity from 10 features to 20 features in **make_friedman1** will increase the quantity of disconnected features from 5 to 15, as the same X1-X5 features produces the 'y' prediction value but with an additional 10 randomly generated numbers for each instance. In other words, the important features only represent 5/20 = 25% of the features in the dataset, making the differentiation more challenging.

The results from this feature importance test are shown in Figure 4-7. Once again, all HPO methods were able to differentiate X1-X5 from other disconnected features although the GPBO model did not identify X3 as relevant at the noise level of 20%.

All HPO methods except TPE determined the correct ranking of X4, X2, X1, X5 then X3 throughout all noise levels. Both TPE and the SELECT method assigned the highest level of importance to the relevant features throughout, with the novel approach assigning the highest amount of importance at both 0% and 5% noise levels, while TPE assigned more importance at 10% and 20% noise levels. It is noted that even TPE can assign a higher importance to the same feature than the SELECT method, it does not necessarily mean that TPE is a better method. The emphasis is that the assignment of a "correct" importance to a feature must accurately describe its influence over the target variable leading to a better prediction. This is evident from the fact that the SELECT method was able to obtain a higher MAE than TPE at 10% and 20% noise levels in this dataset, shown in Table 4-3.

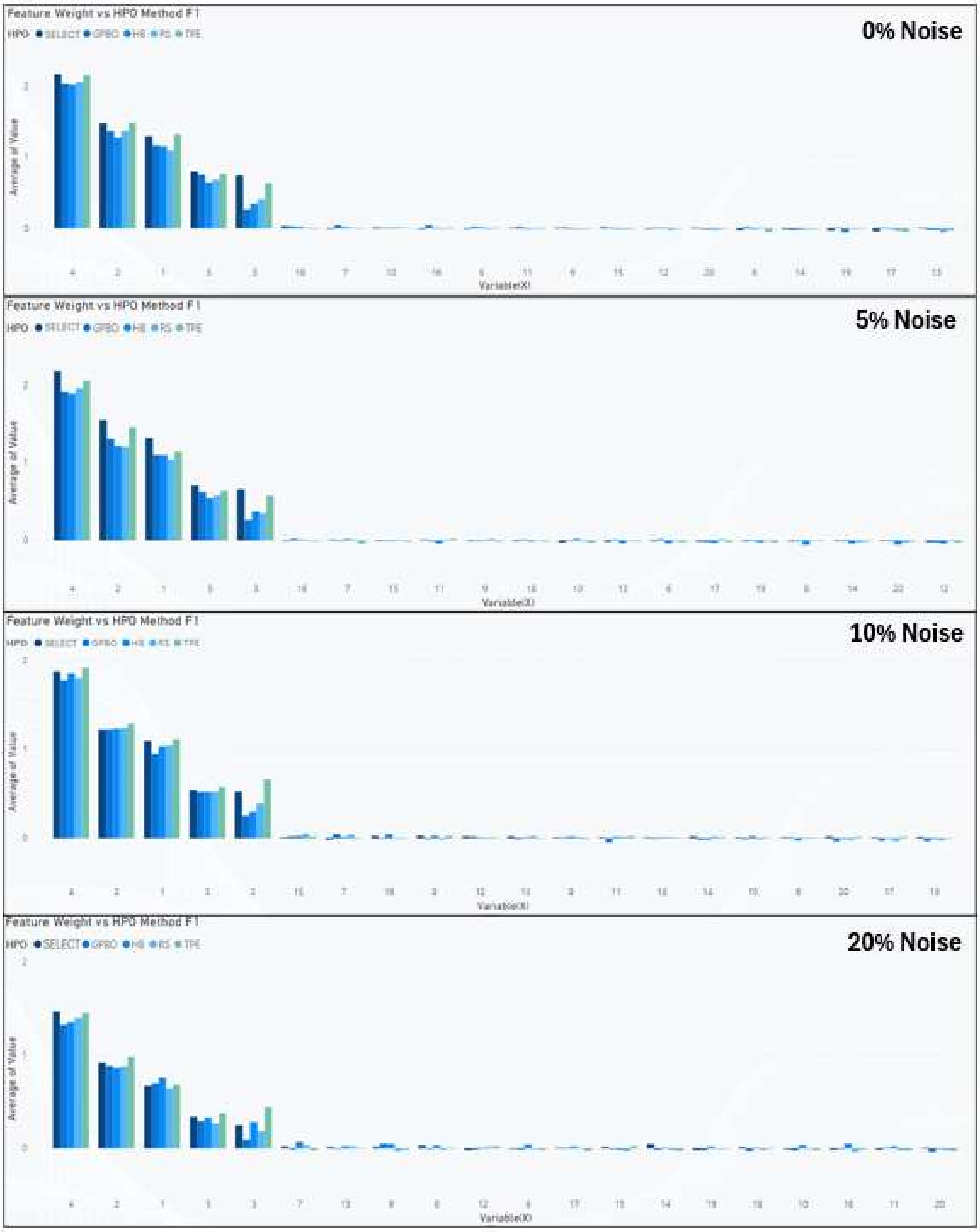


Figure 4-7 HPO Feature Importance, make_friedman1, 500 instances, 20 features

Absolute Feature Importance: Make_friedman2(500,4)

The results of the absolute feature importance with the **make_friedman2** dataset with 500 instances and 4 features can be seen in Figure 4-8. All HPO methods produced models assigned a feature ranking of X3, X2, X1 then X4, which was the same as a previous work, (Kamalov, 2021). The absolute feature importance of X3 ranges between 237-306 and that of X2 between 192-247, in which both are significantly higher than the absolute feature importance of X1 between 1.8-4.1 and that of X4 between 0.2- (-2.2). These results can be explained by Equation 12, as shown below.

$$0 \leq x_1 \leq 100$$

$$40\pi \leq x_2 \leq 560\pi$$

$$0 \leq x_3 \leq 1$$

$$1 \leq x_4 \leq 11$$

$$y = \sqrt{(x_1{}^2 + (x_2x_3 - \frac{1}{x_2x_4})^2} + \epsilon$$

(3)

Using the equation for y, the range of x_2x_3 goes between 0-1759, and, $\frac{1}{x_2x_4}$, ranges between 5x10^{-5} - 8x10^{-3} meaning that the significance of X4 is negligible inside the same brackets; regardless of which value between 1-11 (equation 12) is generated, X4 will have little impact on the model performance. The significant value of x_2x_3 would then be squared, resulting in a potential range between 0-3.1x10^{6}, dwarfing the potential impact of the ${x_1}^2$ component which would range between 0-1x10^4. This would suggest that X2 and X3 have the largest impact, followed by X1 and then X4. The absolute feature importance results obtained by all HPO methods are well-aligned with this logic. This also agrees with a previous study, (Liu and Liu, 2020), which utilised several ML models to analyse the top attributes in **make_friedman2**, highlighting the X2 and X3 variables as both significantly important but failed to detect the importance of X4 and X1. This may be due to a sub-optimal underlying model in (Liu and Liu, 2020), which further emphasises the benefit of HPO for ANN based feature importance.

The SELECT method has assigned the highest importance overall to the most significant features. Being the only method, the SELECT method was able to identify the positive importance of X4 at both 0% and 5% noise levels. Combining this with the better prediction accuracy for all noise levels, the SELECT method continues to show evidence of superior capability in recognising the feature relationships within this dataset.

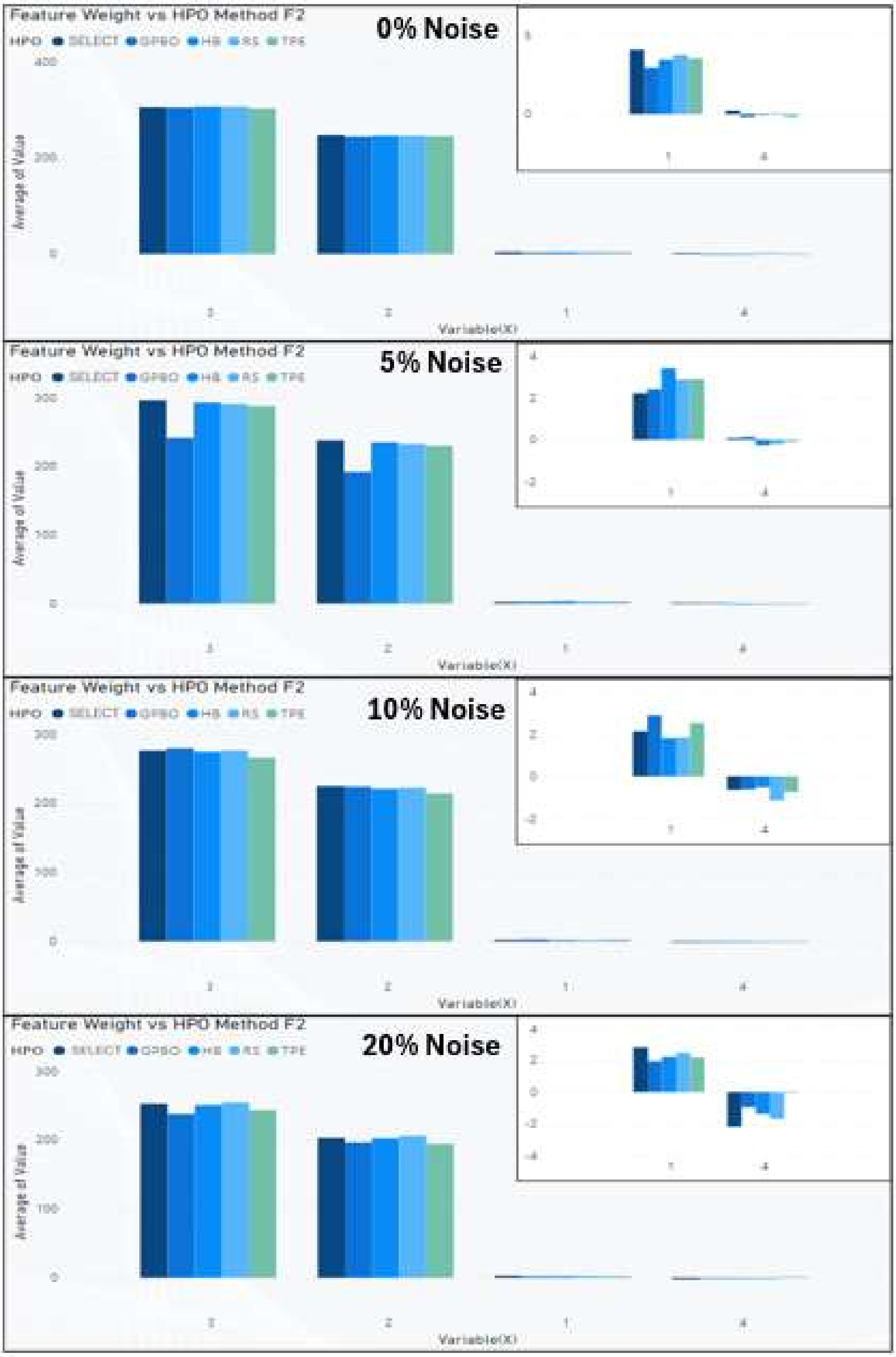


Figure 4-8 Feature Importance, make_friedman2, 500 instances, 4 features.

4.4.2 Stage 2: Feature Importance Validation

From Stage 1, the model optimised by the SELECT method not only demonstrates higher predictive accuracy, but also greater ability to differentiate influential features from non-influential ones. Stage 2 will compare the performance of different feature importance methods on the SAME model optimised by the SELECT method. The three well-known feature importance tools including SHAP, PFI and LIME will be examined with their relative importance following the same process as discussed in Section 4.2.5.

With the help of PowerBI software, the feature importance analysis will be discussed over the **make_friedman1** dataset and its variations, followed by the **make_friedman2** dataset and its variations. The two real-world datasets, **Boston Housing** and **Concrete Compressive Strength** datasets will be examined, followed by the discussion of the experimental results.

Relative Feature Importance: Make_friedman1(500,10)

Beginning with the **make_friedman1** dataset with 500 instances and 10 features, the graphs of the feature importance at all noise levels can be seen in Figure 4-9. All methods of feature importance have distinguished between the 5 connected variables and the latter 5 disconnected variables, while the order is interpreted differently. These results are consistent throughout all noise levels, highlighting a robustness in each feature importance method.

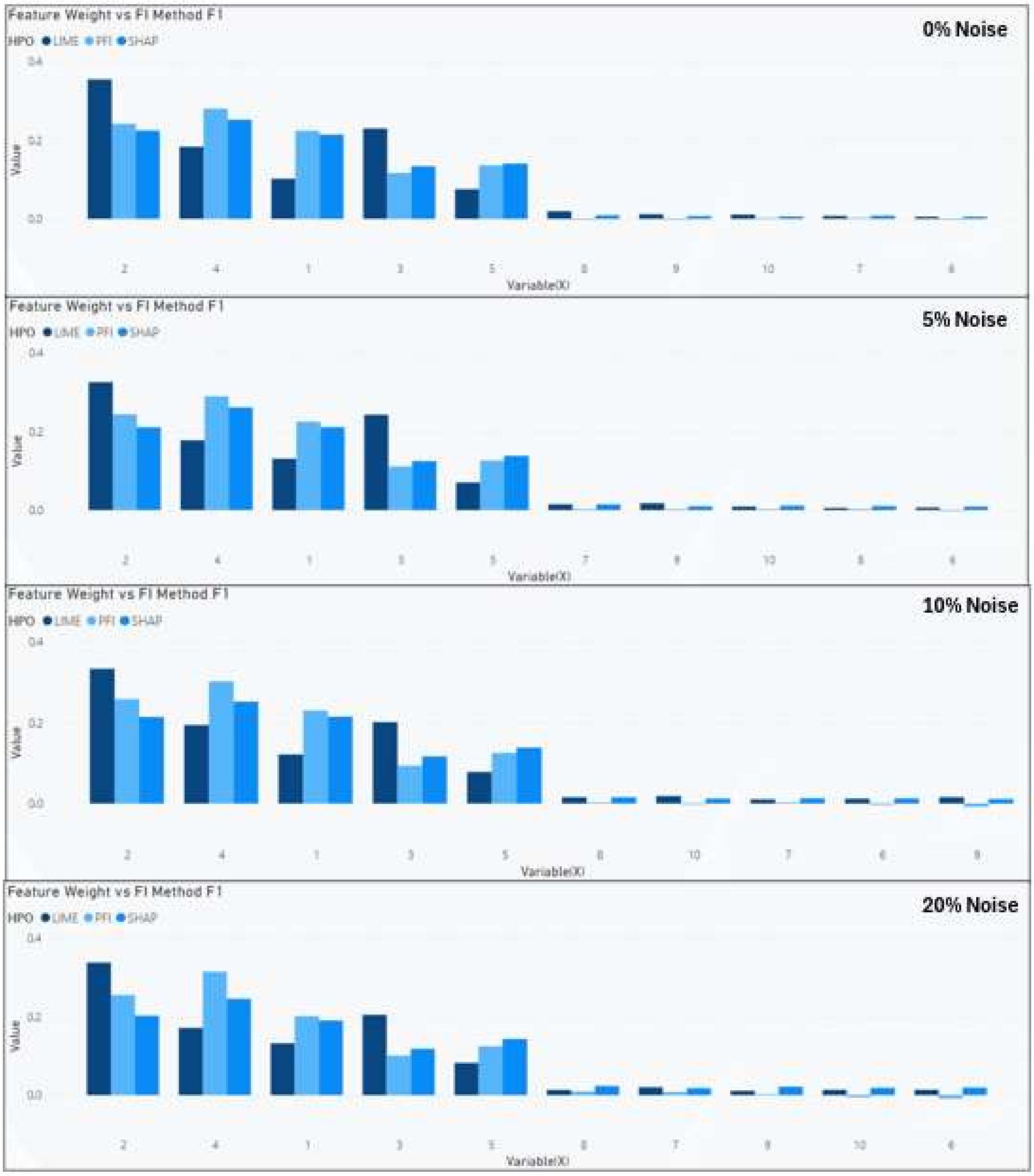


Figure 4-9 Validation Feature Importance, make_friedman1, 500 instances, 10 features.

Both SHAP and PFI can generate the same ranking of features which is X4, X2, X1, X5 and then X3 in descending levels of importance. Specifically, PFI allocates higher importance values to X4, X2 and X1, than SHAP, and SHAP assigns lower importance values to X5 and

X3 than PFI. Despite the above differences, these two approaches are deemed similar in terms of feature importance analysis.

LIME method, on the other hand, has returned a different feature ranking by assigning a much higher importance values to X2 and X3, than X4, resulting in a ranking of X2, X3, X4, X1 and X5, as opposed to the ranking, X4, X2, X1, X5, and X3, obtained by SHAP and PFI methods. This ranking suggests that LIME method allocates higher importance values to X3 and places X2 over X4. These differences indicate that LIME might be capturing local nuances in the model's behaviour that differ from the global perspective given by PFI and SHAP. This also implies that, in certain instances, X3 exhibits a higher level of importance compared to others. This can be analysed in reference to the **make_friedman1** equation, shown again below.

$$y = 10\sin(\pi x_1 x_2) + 20(x_3 - \frac{1}{2})^2 + 10x_4 + 5x_5 + \epsilon \tag{4}$$

X3 is part of a squared component and inside a bracket where the 0.5 is subtracted from a normally distributed value between 0 and 1, this would suggest that more commonly the values would tend towards 0, while the tales of the normal distribution would more occasionally produce higher numbers. The occasional elevated importance of X3 could be attributed to this aspect of the equation and its interaction with other variables in a way that can significantly influence the model's predictions. The squared component amplifies the impact of X3 in specific contexts, contributing to its sporadic but notable importance.

Relative Feature Importance: Make_friedman1(1000,10)

Increasing the number of instances to 1000 in the **make_friedman1** dataset, Figure 4-10 shows that all feature importance methods now produce the same ranking of the top three features, X4, X2, and X1. For the ranking of the last two features, all methods tend to suggest X5, and X3 at 0% noise level. When noise levels increase, all methods tend to swap the ranking between X5 and X3 due to variations in noise. This has also been witnessed in a previous work (Greenwell et al., 2020).

It is noted that, with larger instances, LIME has assigned similar importance to the features as SHAP and PFI. It is because the larger dataset provides LIME with a richer set of examples to build local approximations and reduce the impact of randomness in the sampling process.

Figure 4-10 Validation Feature Importance, make_friedman1(1000,10)

Relative Feature Importance: Make_friedman1(500,20)

Increasing the number of features in **make_friedman1** to 20, with 15 disconnected features, adds further difficulty for the feature importance methods to distinguish important features.

Figure 4-11 indicates that all feature importance methods rank all relevant features as the top features. Both SHAP and PFI have continued to produce similar importance ranking by once again assigning X4, X2 and X1 with the most importance and X5 and X3 to a lesser extent.

Figure 4-11 Validation Feature Importance, make_friedman1, 500 instances, 20 features.

LIME appears to have assigned a high level of importance to X4 but a reduced amount of importance to all other connected features, as compared to the case of **make_friedman1**(500,10) as shown in Figure 4-9. This suggests that the increased uncertainty from a greater number of disconnected features impacts the local evaluations of importance for LIME. It's notable that the challenges faced by LIME in interpreting the importance of features with a higher quantity of disconnected features may be attributed to its local nature of evaluation and the relatively limited dataset of 500 instances, highlighting the method's sensitivity to dataset characteristics. The difficulty is emphasised further when the noise level is increased to 20%. The assigned relative importance of X2 and X3 by LIME method have reduced to a point where they could be deemed as unimportant. Even with the increase in noise levels, PFI and SHAP methods show resilience to the uncertainty and continue to highlight the connected features.

Relative Feature Importance: Make_friedman2(500,4)

With the **make_friedman2** dataset having 500 instances and 4 connected features, the relative importance results by LIME, SHAP and PFI can be visualised in Figure 4-12. The importance values of X3 and X2 are rated much higher than that of X1 and X4, agreeing with previous findings, (Liu and Liu, 2020). SHAP is the most consistent method among all the approaches. PFI once again, doesn’t assign importance to X4, particularly when the noise level increases. LIME method is the least stable method among all three approaches, with X3 and X2 being assigned similar levels of importance at 20% noise label, while the ranking of X1 and X4 varies throughout the noise levels.

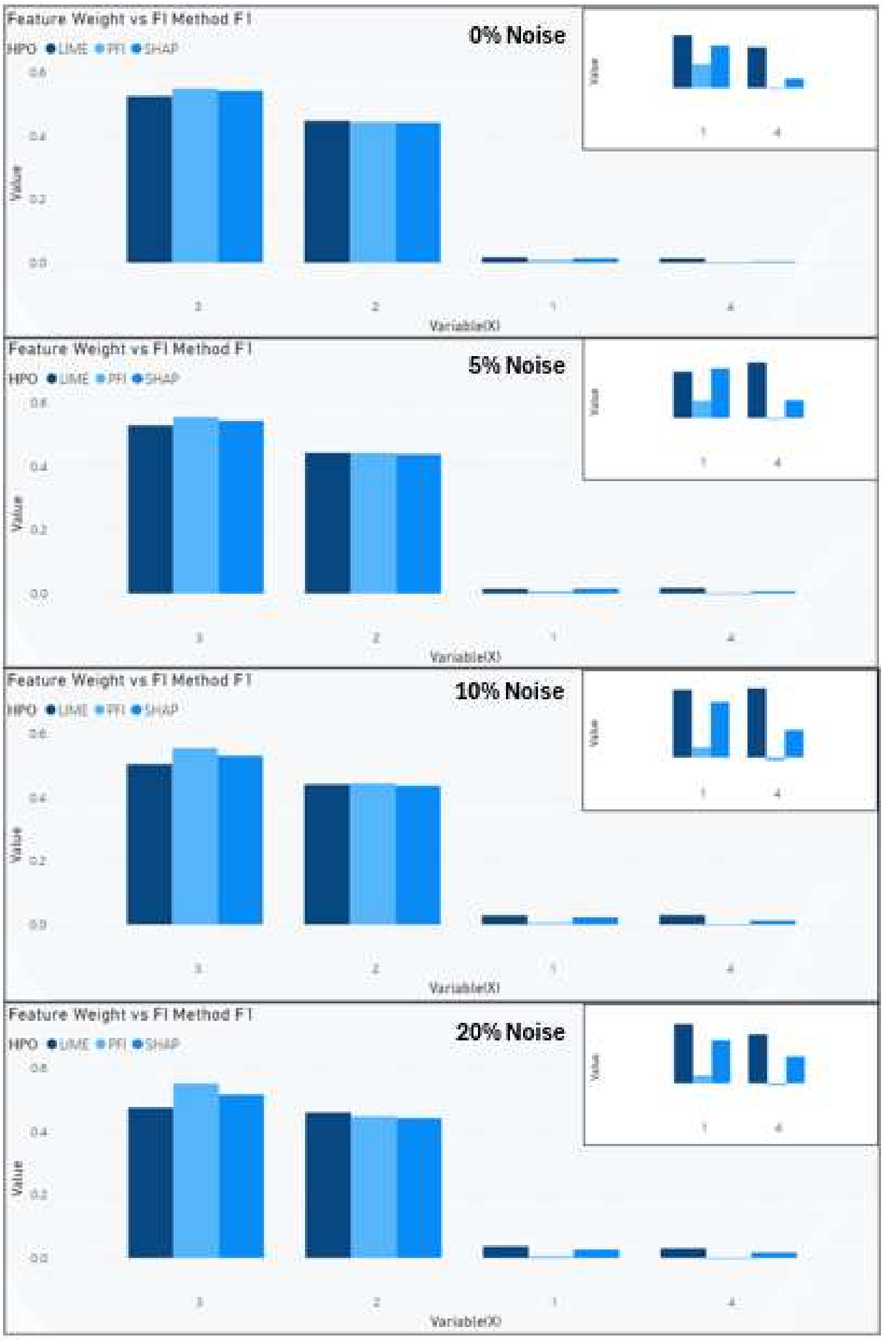


Figure 4-12 Validation Feature Importance, make_friedman2, 500 instances, 4 features, 0% noise.

Relative Importance of Boston Housing Dataset

Next, a real-world dataset, **Boston Housing**, is examined and it consists of 14 features in total. Each of these features is described in Table 4-4. A total of 13 input features have been used to predict the target feature, MEDV, the median value of owner-occupied homes in $1000s (Bataineh and Kaur, 2018).

Table 4-4 Feature descriptions for the Boston Housing dataset

Feature	Description
CRIM	Per capita crime rate by town
ZN	Proportion of residential land zoned for lots over 25,000 sq. ft.
INDUS	Proportion of non-retail business acres per town
CHAS	Charles River dummy variable (1 if tract bounds river; 0 otherwise)
NOX	Nitric oxides concentration (parts per 10 million)
RM	Average number of rooms per dwelling
AGE	Proportion of owner-occupied units built prior to 1940
DIS	Weighted distances to five Boston employment centres
RAD	Index of accessibility to radial highways
TAX	Full-value property tax rate per $10,000
PTRATIO	Pupil-teacher ratio by town
B	$1000(Bk - 0.63)^2$ where Bk is the proportion of Black residents by town
LSTAT	Percentage of lower status of the population
MEDV	Median value of owner-occupied homes in $1000s (target feature)

Figure 4-13 shows the feature importance results by SHAP, PFI and LIME methods , and Table 4-5 reports the feature ranking, 1 being the most important and 13 being the least. The two features, LSTAT and RM, are ranked high by all methods, particularly SHAP and PFI. The same ranking can be seen in previous studies, e.g.(Parr et al., 2024, Oh, 2019, Chen, 2021). Further to this, SHAP and PFI have produced the same level of ranking for the top 5 features, namely LSTAT, RM, CRIM, RAD and DIS in descending order. LIME is also able to include LSTAT, RM and CRIM in the top 5 rankings as well, showing the benefit of using multiple approaches for feature analysis.

Compared to other research, e.g. (Oh, 2019), they found that LSTAT was the most important feature for predicting MDEV, and RM, CRIM, RAD and DIS were deemed as highly important while AGE, CHAS and INDUS were rated as least influential. The above results are well-aligned with the findings of both SHAP and PFI in this experiment but LIME has assigned a high importance to CHAS, which is not found in the literature. LIME suggests that in local circumstances, the feature, CHAS, can occasionally have a significant impact on performance, though not frequently. This aligns with the binary nature of the feature itself that only 7% of instances have a value of 1 compared to 93% with a value of 0. Globally, CHAS may not substantially contribute to predictive accuracy, but it demonstrates a high impact on the rare occurrences when it does influence predictions. This is a good example of how the comparison between these importance methods can lead to a more robust understanding of the dataset in the real-world.

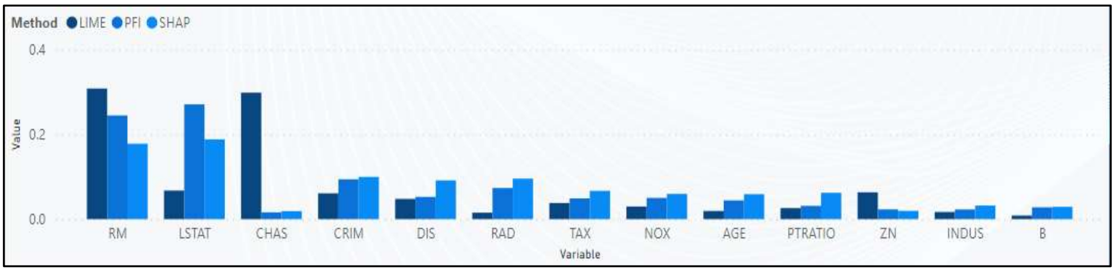


Figure 4-13 Feature importance graph for Boston Housing dataset

Table 4-5 Ranked feature importance for SHAP, LIME and PFI importance for the Boston Housing dataset

Feature	SHAP	PFI	LIME
LSTAT	1	1	3
RM	2	2	1
CRIM	3	3	5
RAD	4	4	12
DIS	5	5	6
TAX	6	7	7
PTRATIO	7	9	9
NOX	8	6	8
AGE	9	8	10
INDUS	10	12	11
B	11	10	13
ZN	12	11	4
CHAS	13	13	2

Relative Importance of Concrete Compressive Strength Dataset

Another real-world dataset, **Concrete Compressive Strength**, is examined. In the domain of civil engineering, the strength of concrete is a highly non-linear function of age and the material composition. This dataset consists of 1,030 instances of the compressive strength of concrete and the associated features, as listed in Table 4-6(Shi and Shen, 2022).

Table 4-6 Feature descriptions for the Concrete Compressive Strength dataset

Feature	Description
Cement	Amount of cement in a cubic meter mixture (kg/m^3).
Blast Furnace Slag	Amount of blast furnace slag in a cubic meter mixture (kg/ m^3).
Fly Ash	Amount of fly ash in a cubic meter mixture (kg/ m^3).
Water	Amount of water in a cubic meter mixture (kg/ m^3).
Superplasticiser	Amount of superplasticiser in a cubic meter mixture (kg/ m^3).
Coarse Aggregate	Amount of coarse aggregate in a cubic meter mixture (kg/ m^3).
Fine Aggregate	Amount of fine aggregate in a cubic meter mixture (kg/ m^3).
Age	Age of the concrete in days.
Concrete Compressive Strength (MPa)	The target variable, representing the compressive strength of the concrete.

The results of the feature importance analysis using SHAP, PFI, and LIME are graphically depicted in Figure 4-16 while the numeric ranking of features is presented in Table 4-7. Notably, SHAP and PFI have similar feature importance rankings, with only slight variations observed for *Blast Furnace Slag* and *Age*, both are ranked in Top 3. The Top 5 features are *Cement, Age, Blast Furnace Slag* and *Water*. A similar study, by (Jiang et al., 2022) using SHAP has also highlighted *Cement*, *Age* and *Water* as the top features, with *Blast Furnace Slag* not included as a factor. This approach also showed the *Fine Aggregate*, *Fly Ash*, *Coarse Aggregate*, and *Superplasticiser* as the least important features, agreeing with the results from both PFI and SHAP. These importance results also agree with other studies, such as (Nguyen-Sy et al., 2020), who found the *Cement*, *Age* and *Water* to be of high importance using XGBoost, although they also included *Superplasticiser* as having a high level of importance as well. *Age*, *Cement* and *Water* seem to be consistently deemed as high performers for feature importance, as reported in (Wan et al., 2021) who also utilised XGBoost for feature importance. While the current study assigned a much higher importance for *Coarse Aggregate*, *Fine Aggregate* and *Blast Furnace Slag*, it assigned *superplasticiser* as having the second least important feature. This suggests that even with the same dataset, there is a variability in the results if a different feature importance approach is adopted.

This experiment has shown that the three well-known feature importance tools can produce similar importance measures, particularly a high degree of similarity can be found between PFI and SHAP. LIME method can identify the Top 5 features, the same as

PFI and SHAP, however, the ranking of those features is different from that of the other two methods. This suggests that, in specific instances, each of these features can have a substantial impact on concrete compressive strength. Globally, *cement content, Age, Water,* and *Blast Furnace Slag* consistently emerge as the most crucial factors while *Coarse* and *Fine Aggregate* as well as *Superplasticiser* are deemed as least importance.

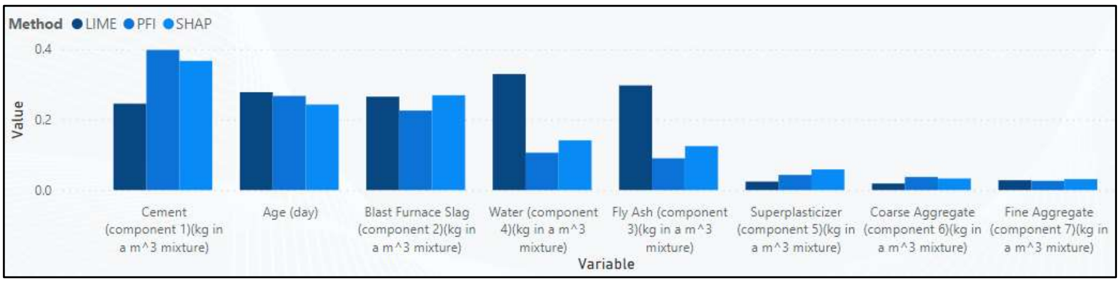


Figure 4-14 Feature importance graph for Concrete Compressive Strength dataset

Table 4-7 Feature importance for SHAP, LIME and PFI importance for the Concrete Compressive Strength dataset

Feature	SHAP	PFI	LIME
Cement	1	1	5
Blast Furnace Slag	2	3	4
Age	3	2	3
Water	4	4	1
Fly Ash	5	5	2
Superplasticiser	6	6	7
Coarse Aggregate	7	7	8
Fine Aggregate	8	8	6

4.5 Summary

In this chapter, a comprehensive exploration of the effectiveness of the SELECT HPO method and its comparison against other benchmark HPO algorithms was conducted in Stage 1. The experiments on the **make_friedman1** and **make_friedman2** datasets scrutinised the algorithmic performance across varying dataset sizes and noise levels, providing a nuanced understanding of the capabilities of the SELECT method.

The exploration into the predictive accuracy, measured by MAE, positioned the SELECT method as the best performer. Across various synthetic datasets, the models optimised by the SELECT method consistently outperformed models optimised by other benchmark algorithms, namely TPE, GPBO, HB, and RS. The superior predictive accuracy of the SELECT models remained even under increasing noise levels, increasing number of instances as well as disconnected features in the datasets, highlighting both the resilience and robustness.

The relevant feature ranking for all HPO methods was conducted in combination with PFI, resulting in consistent feature rankings in line with previous studies for both the **make_friedman1** (X4, X2, X1, X5 then X3) and **make_friedman2** (X3, X2, X1 then X4) datasets. The SELECT optimised model regularly applied higher ranking to the relevant features compared to the benchmarks through variations in noise, the size of the dataset and the synthetic relationship between the input features and the predicted variable. This shows the SELECT method is effective and robust in determining feature relationships, which will prove beneficial for determining CSFs for construction project success.

Stage 2 detailed a comprehensive feature importance analysis of the SELECT method combined with three well-known feature importance tools—SHAP, PFI, and LIME. This stage evaluated the performance of these tools across various datasets, including two synthetic datasets, **make_friedman1** and **make_friedman2**, and two real-world datasets, **Boston Housing**, and **Concrete Compressive Strength**.

The combination of the SELECT method and the three feature importance tools provided a comprehensive analysis of the feature relationships in all datasets. Utilising all three feature importance methods, it presents a holistic perspective with globally consistent identification of significant features, coupled with the nuanced insights from local approximations, underlining the advantages of this combined approach. This provides a broad scope of understanding and explainability for determining the CSFs in construction.

This chapter justified the capability of the SELECT method for determining feature relationships and presented the benefits of combining this approach with multiple feature importance tools. The next step is to integrate this combined algorithm into a DSS to determine the CSFs in construction. The next chapter will discuss and explain the integration of this effective tool into the DSS developed throughout this research period.

5 Decision Support System Development

5.1 Introduction

The purpose of this chapter is to develop a Decision Support System (DSS) by incorporating the SELECT method, and integrating it with the SHAP, LIME and PFI algorithms, to provide an effective tool for improving project sustainability in the construction sector.

This research was carried out in collaboration with an industrial sponsor, Galliford Try Ltd. Hence, useful inputs were collected from staff to understand the industrial guidance and requirements, and project data was gathered to support the DSS development.

As discussed before, sustainability data was not available due to factors outside of the control of the researcher. In response to this challenge, the DSS has been developed with the use of a sample set of data supplied from Galliford Try Ltd. This has led to the incorporation of a key characteristic of the DSS, which is the ability to easily adapt to new data and metrics as they become available in the future. This stage of the development creates the functionality of evaluating project performance, optimised through the SELECT method. The future development of the DSS, once sustainability data becomes available, will incorporate the sustainability dimension into the optimisation of construction project performance.

This chapter will be structured by first discussing the sample dataset and the limitations leading to a focus on the functionality of the system. Then the chosen interface will be discussed, followed by an explanation of the flexible feature importance tool with the integrated SELECT method. This explanation will specify the functions of this flexible tool as well as the integration of the SELECT method into the DSS as well as the ability of adapting to future data. Other supporting tools (or functions) of the DSS will also be discussed.

5.2 DSS Data Collection and Preparation

5.2.1 Supplied Raw Data

In the initial stages of this research journey, Galliford Try provided a package of prepared documents to use for the initial prototype of the DSS. All these documents related to water infrastructure projects which were carried out in the UK and the documents were recorded during the period of 01/2017-12/2020. The raw data consisted of monthly reports, summarised purchase order details, the recorded planned and actual cost of projects, as well as records of the labour spend on each project.

Through collaboration with the relevant staff from the company, the raw data was prepared into a format for analysis covering a total of 126 projects, including the defined input features related to the characteristics of each project, as well as the preferred performance metrics related to finance and delays in projects, these are shown in Appendix 5-1. These designations of data will be discussed next.

5.2.2 Performance Metrics

Three performance metrics were requested by the industrial sponsor in which two were about delay performance and another for financial performance. To be specific, the financial performance metric is the **Commercial Performance (CP)** and the delay performance metrics are the **Forecast Duration Accuracy at PO(FDA)**, and the **On-Site Forecast Duration Accuracy (OSFDA)**.

Commercial Performance

The CP is the measurement of the final cost of a project against the cost estimated at the planning stages (final target price), as shown in Equation 14. For example, if the final cost is over the estimation by 6%, the CP would be 1.06, otherwise it is less than 1 indicating under-spending.

$$\mathbf{CP} = \frac{Final\ Cost}{Final\ Target\ Price} \tag{14}$$

Forecast Duration Accuracy at PO and On-Site Forecast Duration Accuracy

Two metrics are related to project duration performance. The first is the FDA, which shows how well the project duration is estimated at the creation of the purchase order against the actual duration of the project, calculated through Equation 16.

$$\mathbf{FDA} = \frac{Actual\ Onsite\ Duration - Onsite\ Duration\ Forecast\ at\ PO}{Onsite\ Duration\ Forecast\ at\ PO} \tag{16}$$

The second metric is the OSFDA, which looks at how well the project duration is estimated when work fully mobilises on-site, compared to how long the actual duration is at completion. This is shown in Equation 17.

$$\mathbf{OSFDA} = \frac{Actual\ Onsite\ Duration - Onsite\ Duration\ Forecast\ at\ Fully\ Mobilised}{Onsite\ Duration\ Forecast\ at\ Fully\ Mobilised} \tag{17}$$

Like the FDA, the OSFDA also measures the difference between the plan and reality, but specifically at the beginning of on-site work. So, while FDA checks the accuracy of the project duration estimate at the purchase order stage, the OSFDA assesses how well the duration is estimated when work starts on-site. These are both taken as a fraction of the estimation. To clarify, if the actual duration of a project extends by 50% over the estimated duration, then this will be 0.5 or 50% as a percentage. If FDA or OSFDA is 0, it means that the duration estimate is the same as the actual duration upon project completion. If FDA or OSFDA is less than 0, the actual duration is shorter than what was estimated, otherwise there is a delay in the project from the plan.

5.2.3 Input Features

Project characteristic data was supplied for the development of the DSS. These cover 5 project areas: the timeline of the project, the allocation of personnel, the project location, financial details, and the nature of the project work. A list of the input features with their descriptions is shown in Table 5-1.

Table 5-1 The input features used for developing the decision support system

Group	Feature Name	Description	Primary Source
Time	On-site Start Month	The month when the project began work on site.	Monthly project progress reports ranging between 2017-2020.
	Design Duration%	Percentage of the project spent on design stage and then on-site. Before April of 2018, there was a different format which did not include the on-site/design split so capex 2 was used as the split.	
	On-site Duration%		
	Project Duration (Weeks)	The full length of the project in weeks	
Personnel	Proposal Team	The percentage of the total recorded staff hours booked for estimators on a project.	Project Staff Bookings recorded for all projects
	Design Team	The percentage of the total recorded staff hours booked for CAD design engineer, design engineers, design leads and process engineers summed together on a project.	
	Project Management	The percentage of the total recorded staff hours booked for planners, project managers and quantity surveyors summed together on a project.	
	Site Management Staff	The percentage of the total recorded staff hours booked for site managers, site engineers, site foremen and mechanical supervisors summed together on a project.	
	Overheads	The percentage of the total recorded staff hours booked for project overheads.	
	Health and Safety Staff	The percentage of the total recorded staff hours booked for health and safety.	
	Commissioning Team	The percentage of the total recorded staff hours booked for commissioning engineers on a project.	
Location	County	The location of the site in which the project takes place, grouped together by county.	Selected the locations, project ID's and project descriptions from the monthly project progress reports
Financial	Average of Cost Intensity	The total expenditure of a project divided by the number of weeks spent **on-site**	Purchase Order Details summarised by Galliford Try
	First Net Construction Band	Classified construction cost range: <£1M, £1M-£2.5M, £2.5M-£5M, £5M-£10M,>£10M	Purchase Order Details summarised by Galliford Try
	Sum of Total Construction Score	This is the cost of the project	Purchase Order Details summarised by Galliford Try
Project Nature	Chem Dosing	The percentage of the total project expenditure, put into all classifications listed, is based off of interpretation from the Purchase Orders for each project. Essentially the nature of the work.	Purchase Order Details summarised by Galliford Try
	Civil Installation		
	Elec Installation		
	ICA		
	MCCs/MCP		
	Mechanical Install		
	New Building		
	New Water Retain Structure		
	Power Supply		
	Pumps/Booster Set		
	RGF Refurbishment/New		
	Scraper Bridges		
	Screen and Compactor/ Grit Removal		
	Security		
	Tank		
	Temporary Works		
	TTU		

5.2.4 Data and the Focus on DSS Functionality

It is crucial to note that those performance metrics (Section 5.2.2) and input features (Table 5-1) mostly contribute to the economic performance of a project, which can be measured in project cost as well as delay (time). Although project delays can impact both environmental and social sustainability through increased waste and consumption of resources and material, the supplied data is not sufficient for a comprehensive evaluation of all three dimensions of sustainability.

If there was a larger availability of data which relates to all three of the environmental, social, and economic sustainability metrics, key data would include:

- **Material Data:** Information on the environmental impact of materials, including carbon footprint, recyclability, toxicity, and energy consumption during production. This data helps assess which materials contribute to lower carbon emissions and support a circular economy.
- **Energy and Water Usage:** Data on energy and water consumption across project stages—from raw material extraction to construction and operation. Monitoring these metrics would support energy efficient project practices, enable water conservation, and reduce greenhouse gas emissions associated with construction activities.
- **Waste Generation and Management:** Data on waste types, quantities, and disposal methods throughout the project lifecycle. Tracking waste data can inform strategies to minimise construction waste, identify opportunities for reuse or

recycling, and reduce landfill contributions, ultimately promoting a circular economy approach.

- **Labour and Social Metrics:** Information on workforce conditions, safety standards, and labour practices helps assess the social impact of construction projects.

Overcoming the challenge related to data availability resulted in the development focus of the DSS shifting from improving project sustainability to creating the bespoke functionality that can adapt to sustainability data in the future. To develop the DSS, the supplied data is only used to demonstrate the functionality of DSS in modelling the relationships among project features and identifying CSFs. It is expected that the DSS will be able to improve project sustainability when sustainability data become available in the future.

5.3 Design Architecture of the DSS

The DSS can be broken up into three interactive components: the user interface, the inference engine, and the knowledge base, shown in Figure 5-1. Each of these components contributes to the successful functionality of the system. In this section, the summary of the functionality of each of these components will be given before an overview of the overall functionality of the DSS is provided in the subsequent sections.

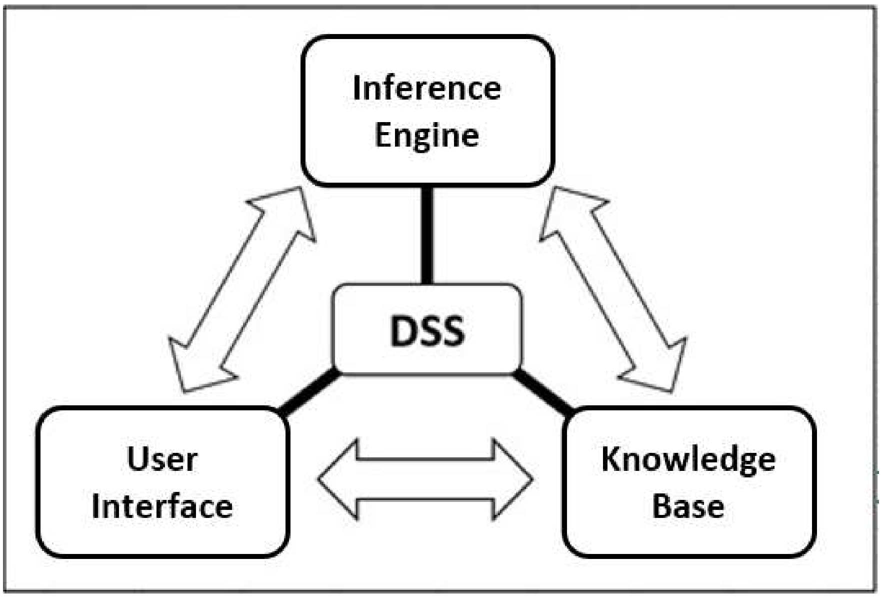


Figure 5-1 The three key components of a DSS

User Interface: This allows the user to interact with the DSS through input selections and visual representations of knowledge and analyses. The selected interface for this system is PowerBI, which will be discussed in more detail in the following section.

Inference Engine: This is the analytical component of the DSS which is responsible for processing data and generating insights. This component is composed of python code and libraries collected and developed for the processing of data, training neural networks, hyperparameter optimisation, the feature importance functionalities, and the integration of these packages into the PowerBI interface. This component is used to connect the external datasets, create internal refined datasets, and compute all aspects of the functionality for the DSS pages to operate effectively. This is an evolution of the base functionality of PowerBI data processing and analysis with the newly developed SELECT HPO method and the integration python code and packages.

Knowledge Base: This is the foundation source data which is supplied to the system for processing and analysis, to generate insights. The sample dataset shown in Appendix 5-1 is the primary source of data at present. Utilising the same format, this dataset can be amended in the future to a greater knowledge for analysis in the system.

5.4 DSS Overview of Functionality

PowerBI was primarily selected at the request of the industrial sponsor to support smooth integration of the DSS into their current systems. This software is beneficial for the DSS development for multiple reasons, (Aspin, 2016) as listed below:

User-friendly Interface: The tool is user friendly for development purposes, allowing for ease of creating dashboards of information, collecting, and connecting data and is also interpretable for persons of varying levels of technical expertise.

Data Visualisation Capabilities: PowerBI has a high level of capability for creating various visual representations related to data (Gonçalves et al., 2023). There are standard visuals which can be seamlessly created, while other visuals can be downloaded, or produced with the help of Python or R programming visuals.

Data Source Flexibility and Connectivity: Data can be utilised from a wide variety of sources in varying formats. These can be connected through details in the data with a high level of flexibility. The use of Power Query allows for data sources to be manipulated for use in varying formats and prepared for analysis (Krishnan, 2017).

Interactive: There is a capability for creating filters, buttons, and other interactive tools to allow users to explore the relationships between their data allowing for enhanced decision making with the aid of a graphical user interface (Becker and Gould, 2019).

Sharing and Collaboration: Upon the completion of developed dashboards related to data, the user can publish reports to the PowerBI service, allowing for the sharing of insights and capabilities with a broad group of stakeholders (Seturidze and Topuria, 2021).

PowerBI serves as the interface for all functions developed in the current DSS. It manages connections to the inference engine files and storage locations for both internal and external datasets. Figure 5-2 shows the contents page of the DSS linked to all the other pages in the system. It is designed with a colour scheme matching the Galliford Try logo for visual appeal.

On the bottom right corner of the content page, a red button "Measurements for Project Performance" recalls the definition of the current performance metrics to ensure clarity for users. Additionally, there are other buttons to access to different functions of the DSS. The primary function, the "Flexible Feature Importance" tool, is positioned at the top of the page, followed by other functions. The "Flexible Feature Importance" function allows users to perform the feature importance analysis which will be detailed in the following section.

Figure 5-2 All Pages in the DSS

5.5 Flexible Feature Importance Function

This section will first explain the key components of the flexible feature importance function before discussing the integration of this tool with the SELECT method using

PowerBI and Python. Figure 5-3 shows the flexible feature importance analysis function. For ease of explanation, each component of this page is labelled by a letter and will be discussed from "A" to "G" as follows.

5.5.1 Flexible Feature Importance Function Page Layout

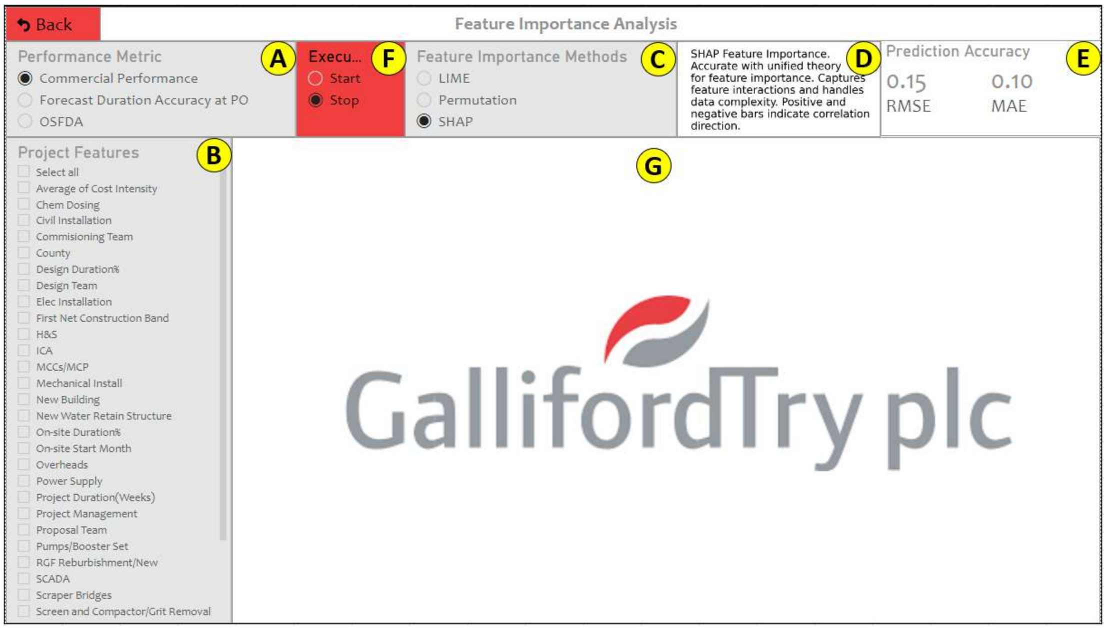


Figure 5-3 Flexible feature importance analysis page before executing analysis

A – The Performance Metric: This component empowers users to choose the performance metric against which the neural network is trained for feature importance analysis. It's important to emphasise that the optimal hyperparameter settings for training the neural network are pre-defined for each performance metric. This design ensures that users can achieve the best performance tailored to each metric, individually.

B – Project Features: Project features are the core inputs to the feature importance analysis when making prediction over the performance metrics (outputs). This component acts as a filter, empowering users to selectively include specific features (inputs) in the analysis. This flexibility ensures users can tailor the analysis to their specific needs and preferences.

C – Feature Importance Methods: Users can choose from three well-known feature importance methods, offering them flexibility in feature importance analysis. This selection also enables users to compare the performance of multiple feature importance methods under the same feature selection and performance metrics, fostering a comprehensive and resilient approach to feature importance. The available methods are SHAP, LIME, and PFI which are extensively discussed in Chapter 4. Key points related to these methods in the DSS are as follows:

- A positive PFI results, shows the feature is contributing to the prediction model performance, while a negative value shows that the feature is not contributing to the prediction performance.
- With SHAP, and LIME, these approaches highlight the direction of the relationship between the features and the performance metric in terms of a positive or negative value where a positive (negative) importance indicate a positive (negative) correlation.

D – Information Visual: A visual has been added to provide situational information related to the function of the Flexible Feature Importance tool, depending on the selections made on the page at the time.

E – Base Model Performance: As the user has the option to alter the input features and the performance metric, this will induce changes in the predictive accuracy of the model. To allow for a reference point in model performance, this component presents the current performance of the optimised neural network with all input features included. This is presented as both the root mean squared error (RMSE) and the mean absolute error (MAE).

F – Execute Button: This component allows users to initiate the training with the current selection. It also enables users to de-activate the training and update the selection.

G – Feature Importance Result Visual: This visual is created using Python and uses the selected values shown in Figure 5-3, for the Performance Metric 'A', Project Features 'B', Feature Importance Method 'C' and Execute 'F' as inputs.

If the 'F' input is set to execute, then the feature importance Python pipeline will begin. The 'Project Features' and 'Performance Metric' values will form the 'visual dataset' for the ANN. The ANN will previously have been optimised with the SELECT method for each chosen 'Performance Metric' value. The ANN will train on the 'visual dataset' and then carry out feature importance according to the selected 'Feature importance Method'. The results will then be presented graphically in the form of a bar chart, an example is given in Figure 5-4.

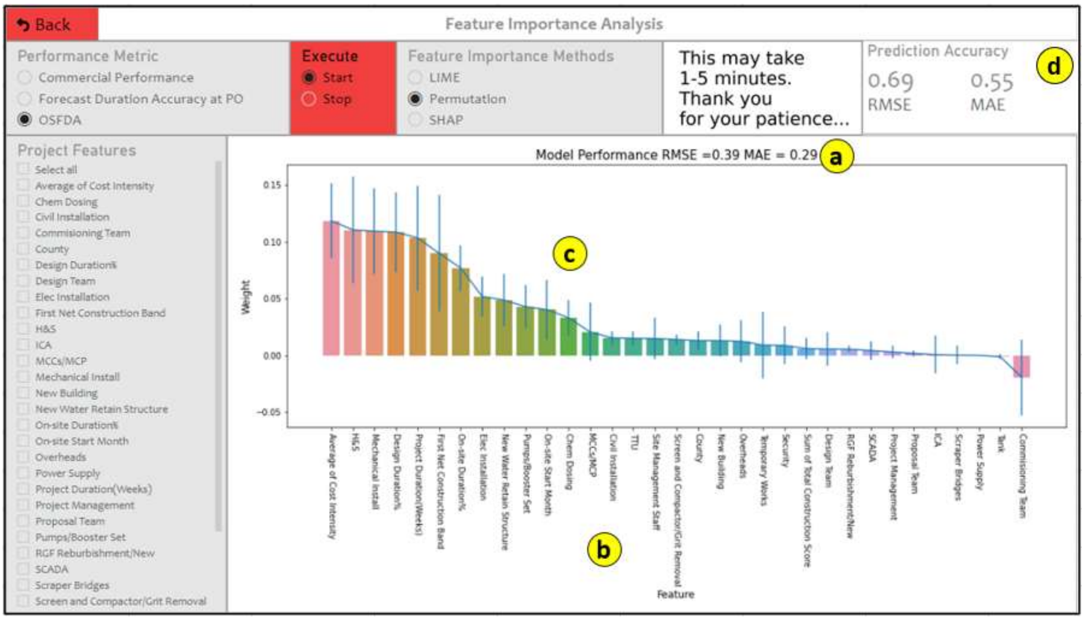


Figure 5-4 An example of feature importance visual results

Figure 5-4 shows, visually, the feature importance weights (shown in "c") of all selected input features in the x-axis (shown in "b") related to the chosen performance metric using the model optimised by the SELECT method together with the selected feature importance tool. The prediction accuracy of the trained and optimised ANN for the selected project features is measured by RMSE and MAE which is presented at 'a' to provide a confidence level of the findings. This can be compared to the prediction accuracy of the optimised ANN with all Project Features included, shown at 'd', to determine if the new feature selection has improved or reduced the prediction accuracy.

5.5.2 Flexible Feature Importance Development Challenges

Multiple challenges were encountered during the process of developing the flexible feature importance tool. This section will discuss these challenges, highlighting how they were resolved. The key objectives for the flexible feature importance page to function effectively and for the goal of incorporating future data for the analysis of the CSFs are listed below:

- **Train, Predict and Present:** The tool must be able to train neural networks and produce visual representations of the results for interpretation.
- **Dynamic User Interaction:** The user must be able to interact with the buttons on the page to direct the use of the tool dynamically.
- **Optimise Performance:** The tool must be able to optimise the neural network hyperparameters for each performance metric using the SELECT method.
- **Adaptability for New Data:** The tool must be able to adapt to new data and performance metrics as they become available in the future.

The challenges encountered in achieving these objectives, and the solutions to these challenges are shown in Table 5-2 and will be discussed further in the following sub-sections.

Table 5-2 Flexible Feature Importance Development: Challenges and Solutions

Function	Challenge	Solution
Train, Predict and Present	Power BI is not equipped with ANN or feature importance software packages.	Use Python visual function. Develop a Python code pipeline which can read the data, determine feature importance and present the findings.
Dynamic User Interaction	Power BI Python visuals cannot detect filtering or dynamic changes in data.	Create columns in the Python Visual input dataset to represent user selections. Develop Python code to interpret user inputs and guide the operation of the Python Visual.
Optimise Performance	Power BI Python Visuals have a 5 minute time limit for script operation, making HPO impossible.	Utilise Power Query Editor. Develop Python code to apply the novel HPO method to defined performance metrics.
Adaptability for New Data	Future data is not able to be explicitly inputted into the DSS as this is still unknown.	Utilise Power Query Editor. Develop Python code to allow for future amendments to the current dataset, with HPO application.

Train, Predict and Present

Although Power BI has a significant amount of flexibility in manipulating datasets and presenting information in built-in standard visuals, there is not a standard visual at present which can utilise ANN software for analysis. To overcome this challenge, the 'Python Visual' in PowerBI was used in combination with a Python code pipeline, the overview of the steps in this pipeline is shown in Figure 5-5.

This Python visual can accept a dataset in a tabular format, which is then prepared for the ANN, the ANN is trained, predictions are made on a test set, feature importance is carried out and the findings are then presented. This is achieved with a Python library titled 'CGRNN' which has been developed during this research. The developed CGRNN library has been equipped with all the Python functions required in the DSS.

```
Train, Predict and Present
Import relevant libraries
        (import CGRNN_code as CGRNN)
Copy the dataset for amendment

Split and prepare the data for use in this code
        CGRNN.prepare()
Train neural network
        CGRNN.build_NN_model()
Carry out feature importance
        CGRNN.feature_importance()
Make predictions with model

Calculate model performance
        MAE and RMSE
Combine model performance and feature importance into a graph
```

Figure 5-5 Pseudocode for the train, predict and present function in the feature importance tool

Dynamic User Interaction

The Python visual can accept data in a tabular format but the ability to filter or dynamically alter the input dataset in a Python visual is not currently supported in Power BI. This means that there is no standard way of selecting the ‘Performance Metric’, the ‘Project Features’, the ‘Feature Importance Method’ or when to ‘Execute’ the feature importance process. These are capabilities present in the developed Flexible Feature Importance page in Figure 5-2. In addition to this, the optimised parameters for a specific Performance Metric defined by the SELECT algorithm could not be selected.

The solution to dynamic user interaction through button selection in the Python visual, was achieved by inserting additional columns into the Python visual which represented each of the user selections, shown in Figure 5-6.

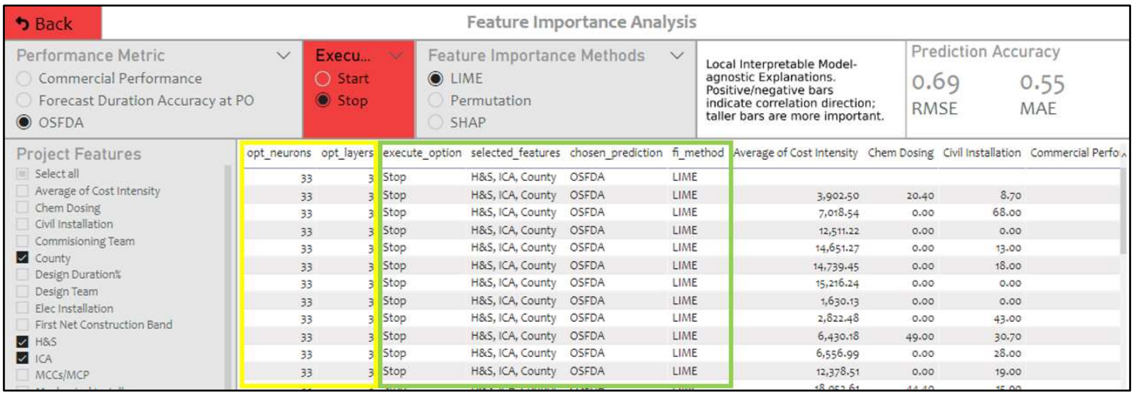


Figure 5-6 Image showing the visual for feature importance with variables in the columns

From Figure 5-6, the user selection criteria are shown in the green box and the corresponding optimised neural network architectures for the chosen Performance Metric are shown in the yellow. With this information inserted into the visual, Python can be used to make logical decisions on how to interpret the data then action the code according to the inputs. A description of this process is provided in Figure 5-7.

```
Train, Predict and Present with Dynamic User Interaction
Import relevant libraries
            (import CGRNN_code as CGRNN)
Copy the dataset for amendment

Define the importanct variables from the columns
            Execute
            Performance Metric
            Project Features
            Feature Importance Method
            Optimised Parameters
Eliminate the unselected input features and other prediction variables
            Using 'Project Features' and 'Performance Metric'
If 'Execute' = 'Stop:
            Present the Galliford Try Logo
                        Using 'Execute'
Else:
            Split and prepare the data for use in this code
                        CGRNN.prepare()
            Train neural network
                        CGRNN.build_NN_model(Performance Metric, Optimised Parameters)
            Carry out feature importance
                        CGRNN.feature_importance(Feature Importance Method)
            Make predictions with model

            Calculate model performance
                        MAE and RMSE
            Combine model performance and feature importance into a graph
```

Figure 5-7 Dynamic user interaction functions for the feature importance tool

Optimise Performance and Adaptability for Future Data

A crucial limitation in Power BI is a 5-minute time limit for running Python visuals. This limit can easily cover the training of a single ANN for operation, but the repeat training required for HPO requires a larger time to achieve the desired result. This would make it impossible to optimise the ANN for each performance metric using the Python visual. This is where the use of Power Query Editor (PQE) for an extended application of Python script was used. PQE is a data preparation tool in Power BI that allows users to transform, clean, and shape data from various sources before loading it into Power BI for analysis and visualisation

The PQE has a Python function which allows a user to manipulate a dataset without the same time constraint of a visual but can only present the results in the format of a table. Using this capability, a source dataset was amended with a Python script which allowed for the incorporation of the SELECT HPO method and adaptability for future changes to the dataset. An overview of this code is shown in Figure 5-8.

```
Optimise Hyperparameters
Import relevant packages
        (import CGRNN_code as CGRNN)
Selections for  operation
        Prediction variables
        HPO (Yes/No)
        filepath for save files
Save the relevant datasets variables
        Performance Metric
        Project Features
if HPO = Yes:
        for Prediction variables:
                Copy the dataset for amendment
                        dataset_copy
                Remove the other performance metrics from dataset_copy

                Split and prepare the data for use in this code
                        CGRNN.prepare()
                Optimise neural network hyperparameters
                        CGRNN.build_NN_model(Performance Metric, HPO=Yes)
                Record best hyperparameters
                        optimised_architectures
                Make predictions with model
                        Load predictions into dataset for all instances
                Calculate model performance on test set (MAE and RMSE)
                        performance_table
else:
        for Prediction variables:
                Copy the dataset for amendment
                        dataset_copy
                Remove the other performance metrics from dataset_copy

                Split and prepare the data for use in this code
                        CGRNN.prepare()
                Train neural network with optimised hyperparameters
                        CGRNN.build_NN_model(Performance Metric, optimised_architectures, HPO=No)
                Make predictions with model
                        Load predictions into dataset for all instances
                Calculate model performance on test set (MAE and RMSE)
                        performance_table
```

Figure 5-8 Pseudocode for updating datasets and hyperparameter optimisation in the decision support system

Figure 5-9 shows an image of the PQE in the DSS. The 'source_input_dataset', highlighted in the red box in Figure 5-9, is the externally supplied dataset for the Flexible Feature Importance page. The code in Figure 5-8 allows a user to select which columns in a 'source_input_dataset', represent the 'Performance Metrics' and whether to activate the novel HPO method or not. From these selections, the code will carry out the optimisation for each selected performance metric and then save the Project Features, the Performance Metrics, and the corresponding optimised ANN hyperparameters as new datasets in the PQE, highlighted in green in Figure 5-9.

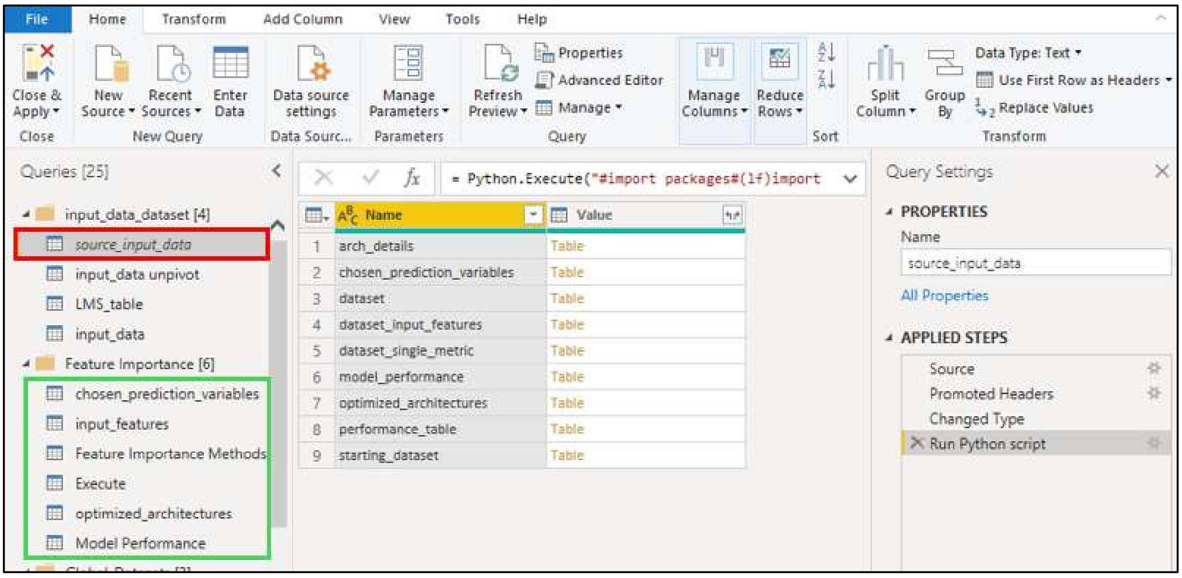


Figure 5-9 Power Query Editor in Power BI with a Python script accessed

The code in Figure 5-8 also facilitates an adaptability for future amendments to the dataset. If a user alters the 'source_input_dataset' or changes the chosen performance metrics, and runs the same Python script in the PQE, then the HPO will repeat and the

Flexible Feature Importance page will update with the amended ‘Project Features’, ‘Performance Metrics’, and the corresponding optimised ANN hyperparameters.

5.6 Additional Functions of the DSS

In addition to the SELECT method and flexible feature importance function, the DSS also encompasses several other functions that are tailored to meet specific needs and requirements of the company. While these functions may not bear significant academic value, they provide a valuable contribution to enhance the overall usefulness of the system. This section will delve into these additional functions, shedding light on their purpose and how they complement the DSS.

5.6.1 Active Project Performance

The active project performance page was initially conceived as a focal point of the research to address the initial chosen knowledge gap in the literature. It aimed to predict project sustainability performance throughout the construction life cycle using AI. However, due to the lack of sustainability data, this is now a secondary function for the benefit of the industrial sponsor, which can be the source of further development with new data as it becomes available.

For dynamically changing project performance throughout the project lifecycle, the only performance metric which had sufficient sample data in this development, was in relation to the OSFDA, the delay performance from starting work on-site through to project completion. Utilising monthly reports spanning from January 2017 to December 2020.

The main function of the active project performance page follows these steps:

1. Convert recorded trends in OSFDA over time and project features from previously completed projects into a sequential training set.
2. Train an optimised CGRNN on the characteristics from past projects to learn the trends in OSFDA over time.
3. Using the CGRNN, predict the future trends in OSFDA for projects which are currently in operation on-site.
4. Using a developed method of temporal PFI to permute features along their sequence in a sequenced dataset, dynamically evaluate what are the most important features impact project performance over set periods of time.

The purpose of this tool is to provide project managers with the capability to predict potential changes in OSFDA monthly and to determine what may be the most important CSFs which impact project during user-defined periods of time.

The Active Project Performance page is depicted in Figure 5-10 and each of the visuals on the page will be discussed in the following sub-sections.

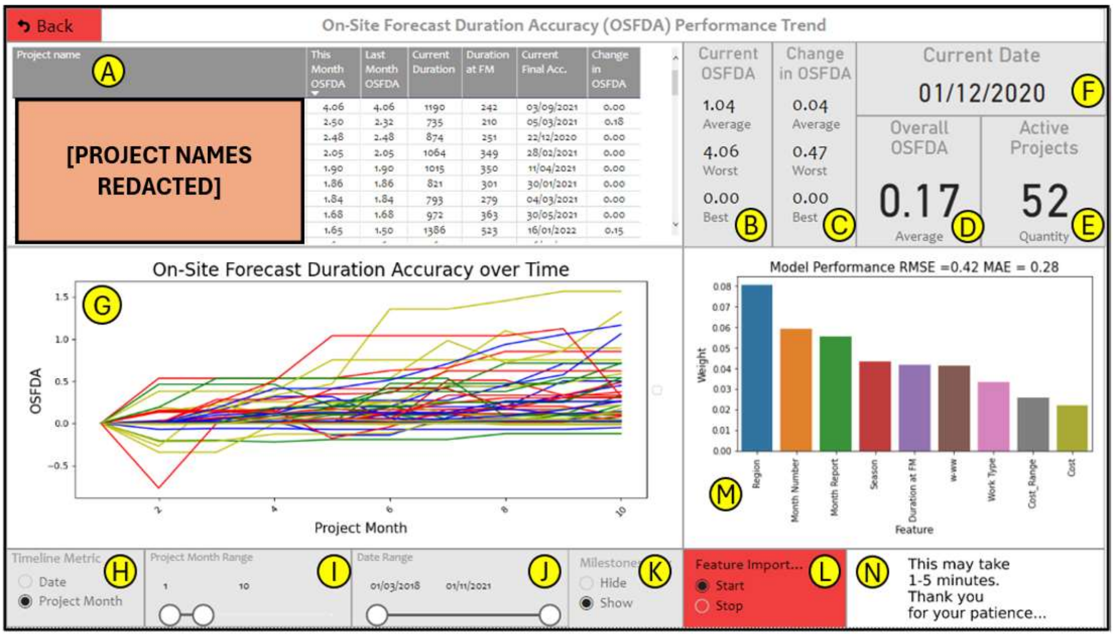


Figure 5-10 Active Project Performance Page

A - Active Project Table: This is a table of the current projects which have begun work on site but are not yet completed, hence known as active projects. Key details of each active project such as the initial planned duration and the current duration, the planned completion date, the OSFDA performance this month and last month as well as the change in OSFDA performance are shown. It allows users to see the current and recent OSFDA performance of all active projects and provides the interactive capability to perform, visually, project-to-project comparison by user selection.

B - Current OSFDA: This visual summarises the mean OSFDA, the worst OSFDA and the best OSFDA recorded for all active projects this month. Shown in Figure 5-10, the worst performing project has an OSFDA of '4.06', this suggests that the project is delayed by more than 4 times the initial planned duration. An OSFDA of '0' means the project is planned to finish on schedule with no delay.

C - Change in OSFDA: This visual summarises the change in OSFDA between the current month and the previous month for all active projects. The mean change in OSFDA, the best change in OSFDA and the worst change in OSFDA are shown here. From Figure 5-10, the worst change in OSFDA is 0.47, which says that the planned completion date for a project has increased by almost 50% of the initial planned duration in the past month.

D - Overall OSFDA: The overall OSFDA presents the mean OSFDA for all projects shown on the page at a given time, through user selection and the period selected. The user can select specific projects, a time range, or project month range and the mean OSFDA of the selection is presented here.

E - Active Projects Quantity: This visual presents the current quantity of active projects or the number of projects in a selection by the user.

F – Current Date: The current date is the date in which the analysis is being conducted. The dataset sample that was used for this development ranged until 01/12/2020, which is why this is the date on the page.

G, H, I, J and K – Interactive OSFDA Line Graph: This line graph presents the historical OSFDA performance and the predicted future performance up to 12 months ahead for all active projects. The timeline metric, 'H', can be changed between the months into a project and the change in date, the range of project month, 'I', and the range in date, 'J', for the line graph can also be defined by the user and the line graph will present the trends in OSFDA according to the user selections. Key milestones can also be highlighted, 'K', such as the initial and current planned completion dates of the project.

M, L and N: After the user has defined a timeline metric and range, this tool can use temporal PFI to determine what are the most important CSFs impacting OSFDA during the user selected range of time. 'M' presents the feature weighting in relation to the user selected period, 'L' allows the user to activate and deactivate the function, while 'N' provides the user with additional information related to the operation of the feature importance tool.

The combined functions on this page are intended to present the user with the current OSFDA performance from the current month and allow the user to study the overall OSFDA and individual OSFDA of active projects. The user can also delve into the trends in performance over selected time ranges and analyse what the CSFs. The functionality of this page has been developed and it is set up to be adaptable for new data as it becomes available.

Data Availability and Adaptability

This function has been developed with the intent for incorporating future data to improve performance. There is insufficient data to gain meaningful insights into the variations in performance at present, but the format of the data is created to allow for future improvements, as described below.

The external source dataset for the active project performance function is shown in Figure 5-11. The blue column is the project ID for referencing each project for all months that it is in operation. The green columns are the temporal data, highlighting the passing of time in multiple formats for the AI to study. The yellow box is related to each project's characteristics which may vary over time or remain constant. The red box should always be filled with '1', the purpose of this is related to teach the ai when a project is active. The performance metric is in purple.

Altering the purple column for another metric will change the trend prediction metric. Keeping this format, while altering the values, or increasing and decreasing the columns in the yellow box will allow the user to amend the project characteristics for future data availability. This includes both constant data, as seen in Figure 5-11, and dynamically changing data throughout the project lifecycle.

Project ID	Monthly Report	Month Number	Season	Month	Year	Duration	Cost	Cost_Range	Region	w-ww	Work Type	Stage	DSFDA
4031250000	01/03/2017	1	Spring	3	2017	228	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0
4031250000	01/04/2017	2	Spring	4	2017	263	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.153509
4031250000	01/05/2017	3	Spring	5	2017	263	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.153509
4031250000	01/06/2017	4	Summer	6	2017	280	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.22807
4031250000	01/07/2017	5	Summer	7	2017	267	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.171053
4031250000	01/08/2017	6	Summer	8	2017	280	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.22807
4031250000	01/09/2017	7	Autumn	9	2017	280	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.22807
4031250000	01/10/2017	8	Autumn	10	2017	280	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.22807
4031250000	01/11/2017	9	Autumn	11	2017	319	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.399123
4031250000	01/12/2017	10	Winter	12	2017	331	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.451754
4031250000	01/01/2018	11	Winter	1	2018	331	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.451754
4031250000	01/02/2018	12	Winter	2	2018	332	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.45614
4031250000	01/03/2018	13	Spring	3	2018	389	2525.7	c£2.5M-£5M	North	Water No	Water Quality	1	0.70614
4037530000	01/03/2017	1	Spring	3	2017	151	195.5	aUp to £1M	West	Water No	Water Quality	1	0
4037530000	01/04/2017	2	Spring	4	2017	151	195.5	aUp to £1M	West	Water No	Water Quality	1	0

Figure 5-11 Temporal dataset format for active project trend prediction

5.6.2 Past Project Performance

This function is used to explore past project statistics and is further divided into three distinct sub-functions: **Project Nature**, **Delivery Strategy**, and **Project Information**. Each of these sub-functions will be discussed below.

Project Nature

This sub-function, as illustrated in Figure 5-12, is an interactive tool that empowers users to visualise the historical performance of projects with varying characteristics and compare these projects across multiple metrics.

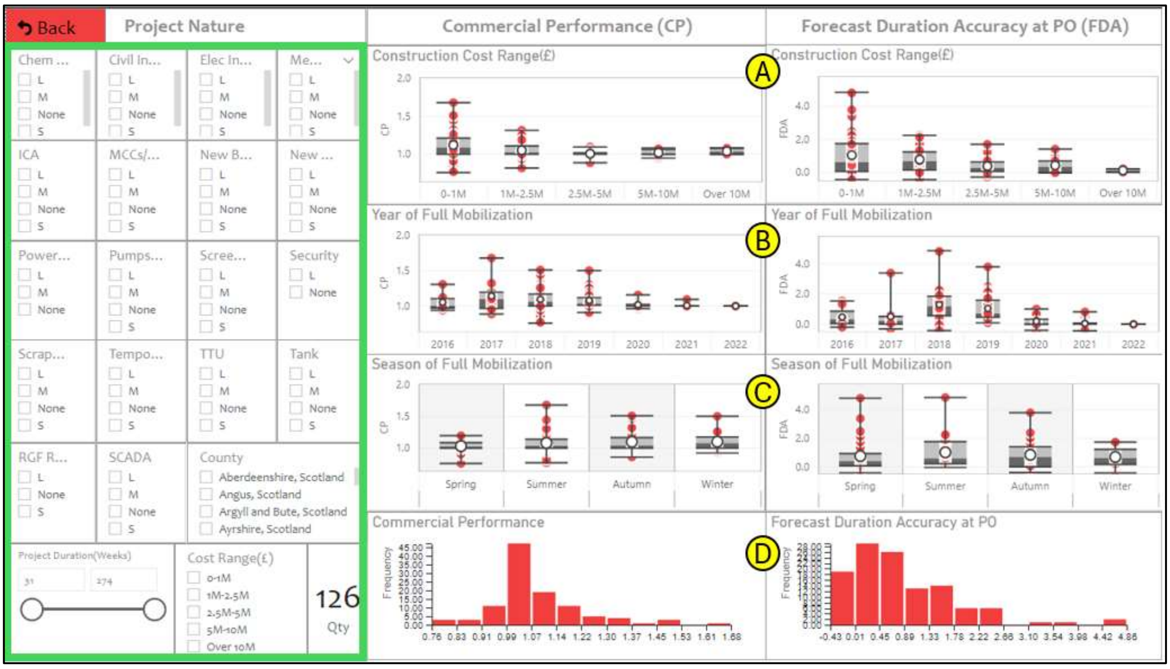


Figure 5-12 Project Nature Page

The left-hand side of the page, highlighted in the green box features filters associated with previous project work undertaken. The filtering criteria is listed in Table 5-3, covering the duration in weeks, the location by county, the project cost, and the nature of the project work. Users can select criteria to visualise statistical trends in the data, represented on the right-hand side of the page.

Table 5-3 Project nature page filtering criteria

Group	Feature Name	Description	Format
Time	Project Duration (Weeks)	The full length of the project in weeks	Adjustable Scale between the minimum and maximum project durations in weeks.
Location	County	The location of the site in which the project takes place, grouped together by county.	Selection by county, multiple selections are possible simultaneously.
Cost	First Net Construction Band	Classified construction cost range: <£1M, £1M-£2.5M, £2.5M-£5M, £5M-£10M, >£10M	Broken down into four classifications, multiple can be selected.
Project Nature	Chem Dosing Civil Installation Elec Installation ICA MCCs/MCP Mechanical Install New Building New Water Retain Structure Power Supply Pumps/Booster Set RGF Refurbishment/New Scraper Bridges Screen and Compactor/ Grit Removal Security Tank Temporary Works TTU	The percentage of the total project expenditure, put into all classifications listed, is based off of interpretation from the Purchase Orders for each project. Essentially the nature of the work.	Each of the project nature criteria is classified as None, S, M,L. None: There is none of the type of work in the project. S: This is in the lowest 33% magnitude of this work type recorded in past projects. M: This is greater the lowest 33% magnitude of this work type recorded in past projects but less than the highest 33% . L: This is in the highest 33% magnitude of this work type recorded in past projects.

The right-hand side of the page splits between delay performance (FDA), and financial performance (CP). These then compare the user selected criteria against the following metrics, referring to Figure 5-12:

A - Cost Range: The user selected project nature criteria is compared over increasing cost ranges in boxplots to determine if the financial impact of projects correlate with the user's selection.

B - Year of Full Mobilisation: This range covers the year of starting work on-site, allowing for comparison of project performance over each year in boxplots.

C – Season of Full Mobilisation: This visual also uses a boxplot for representation but compares projects which begin on site at different times of year, specifically which season.

D – Performance: Against both the delay and financial performance, this final graph is a histogram of the distribution of all recorded project performance from each of the user determined project types.

Delivery Strategy

This sub-function, as depicted in Figure 5-13, empowers users to adjust project delivery strategy controllable factors, listed in Table 5-4, including the allocation of labour hours, the ratio of design time to onsite duration, and the onsite start month.

Table 5-4 Delivery strategy page filtering criteria

Group	Feature Name	Description	Format
Time	On-site Start Month	The month when the project began work on site.	Selection from months of the year.
	Design Duration%	Percentage of the project spent on design stage and then on-site. Before April of 2018, there was a different format which did not include the on- [illegible]	Slider between the maximum and minimum.
	On-site Duration%		Slider between the maximum and minimum.
Personnel	Proposal Team	The percentage of the total recorded staff hours booked for estimators on a project.	Slider between the maximum and minimum.
	Design Team	The percentage of the total recorded staff hours booked for CAD design engineer, design engineers, design leads and process engineers summed together on a project.	Slider between the maximum and minimum.
	Project Management	The percentage of the total recorded staff hours booked for planners, project managers and quantity surveyors summed together on a project.	Slider between the maximum and minimum.
	Site Management Staff	The percentage of the total recorded staff hours booked for site managers, site engineers, site foremen and mechanical supervisors summed together on a project.	Slider between the maximum and minimum.
	Commissioning Team	The percentage of the total recorded staff hours booked for commissioning engineers on a project.	Slider between the maximum and minimum.

Based on a set of chosen project delivery factors, the range of past performance is presented in multiple ways listed below, referring to Figure 5-13:

A - Statistical Performance: The statistical performances from historical projects that match with the chosen factors are displayed under “Recorded Performance from Previous Projects”. This not only presents the best, most common, and the worst performances in CP and FDA, but also the distribution of performance for both metrics in the form of a histogram for each.

B – CGRNN Determined Performance: When the novel HPO method optimises ANN hyperparameters against the performance metrics: CP and FDA, it also makes predictions for all past projects and stores the results for each performance metric. These results are presented in the “Predicted Performance from Historical Data” section of the page.

Utilising the predictions alongside the statistical analysis provides a more comprehensive evaluation of the dataset. An ANN can determine trends in data which may not be noticed from standard statistical means. Using both methods allows a comparison between each for a more robust evaluation of the user’s selected criteria. It also allows the user to visualise how well the ANN optimised by the novel HPO method is performing against actual past instances.

C – Best Model Performance: The combination of the user selected criteria which achieves the best predicted performance against CP and FDA is presented here.

Figure 5-13 Delivery Strategy Page

Project Information

This sub-function, as depicted in Figure 5-14, provides users with access to project details associated with the other two sub-functions, **Project Nature,** and **Delivery Strategy**.

A - Project Nature: This is the project nature data for all criteria and all instances related to Table 5-3, the project nature page data.

B - Delivery Strategy: This is the delivery strategy selection criteria from the delivery strategy page, listed in Table 5-4.

C – Project Performance: This is the recorded performance from all projects, and the predicted performance from the ANNs optimised by the SELECT method.

The tables presented on this page can be selected, allowing users to highlight specific projects of interest and compare various characteristics across the pages related to Past Project Performance.

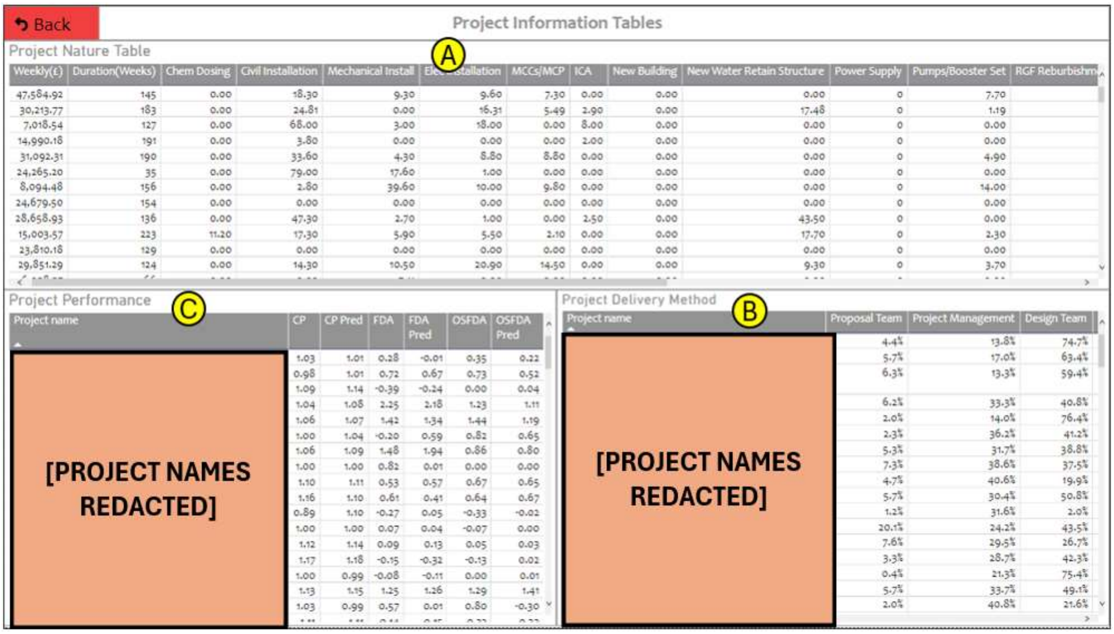


Figure 5-14 Project Information Page

Findings from Sample Dataset

This tool can assist managers in determining the project characteristics or strategies which improve or reduce performance over multiple metrics. From the sample data, valid findings from this tool include the following:

- Projects which begin in spring perform better for both financial and delay performance than other parts of the year, suggesting the best times to start on site for optimum performance.
- The CP performance for most projects ranges between 0.99 and 1.07, providing a scope for over expense.

- The FDA performance is mostly between 0 and 0.89 with a skewed distribution extending beyond this. This shows that most projects have delays of up to 89% of the initial planned duration, the most common delay is 79% over the planned duration.
- Projects which cost under £1,000,000 tend to perform worse than those which have a larger cost, there is also a larger range of performance.
- Projects with large civil installations lead to increases in delays and over expenditure.

5.7 Potential for the Future of the DSS

The DSS has the potential to improve decision-making processes in construction by providing data-driven insights into key performance metrics, such as project costs, timelines, and sustainability goals. By integrating advanced analytics and a flexible feature importance tool, the DSS allows managers to objectively evaluate the CSFs that most impact project outcomes. This can lead to more effective allocation of resources, better risk management, and a reduction in costly delays. For example, if the DSS identifies that certain materials consistently contribute to budget overruns, managers can proactively seek alternative suppliers or adjust project plans. Ultimately, this data-centred approach helps companies maintain tighter control over projects, improving profitability and operational efficiency.

As sustainability metrics are incorporated, the DSS can continuously analyse and highlight the most impactful factors influencing environmental performance, from material choices to energy efficiency practices. This adaptability supports the development of a digital product passport, enabling full traceability of sustainable components across the project lifecycle. By offering transparent and detailed reports on sustainability factors, the DSS not only helps companies comply with regulatory standards but also enhances their reputation in the market, appealing to clients who prioritise environmental responsibility.

5.8 Summary

This chapter documents the development process of the DSS, emphasising the design of its key functions and the relevant challenges. The development process began with an exploration of the raw data supplied by the sponsor company. This involved defining performance metrics related to project cost and delays, identifying input features, and recognising data limitations. As a result, the DSS has been incorporated a functionality to adapt to sustainability data in the future.

The chosen software for the DSS development is Power BI Desktop, due to its user-friendly attributes, robust data visualisation capabilities, flexible data source connectivity, and interactive functionalities.

The flexible feature importance tool is discussed alongside the integration of the SELECT HPO method which optimises the performance of this tool for each user defined performance metric. The flexible feature importance tool is also set up to adapt to new datasets and performance metrics, automatically optimising with the SELECT method for the best results.

This tool also offers an ability to dynamically identify the CSFs impacting the selected project performance metrics from multiple perspectives and provide an objective visualisation of the main findings which help to inform decision-making. A further advantage presents itself in the capability to smoothly adapt to new performance metrics and sustainability datasets as they become available, and this will allow the DSS to improve project sustainability performance as the original aim of this research.

The integration of these functionalities into the DSS creates a system which can allow decision makers to objectively analyse CSFs impacting any of the project performance measurements now and in the future with minimal technical understanding of AI technologies.

Moreover, the DSS is equipped with other supporting functions to inform decision making from historical project data. One of them is the active project performance sub-function which can dynamically analyse the important features contributing to OSFDA throughout the project lifecycle. It can also predict the future trends in performance and analyse the important features impacting pre-selected periods throughout project timelines. Another sub-function is to offer an interactive way of investigating past project performance, covering the nature of project work, and delivery strategies.

To conclude, the DSS is expected to provide a valuable tool to support decision-making in the construction sector. To justify the usefulness of the DSS, the validation of the DSS within an industrial setting will be discussed in the next chapter.

6 Validation of the Decision Support System

Chapter 5 discusses the development process of the Decision Support System (DSS) and its key functionalities. In this chapter, the focus shifts to the critical phase of system development, where the efficacy and impact of the DSS is systematically validated in an industrial context.

Senior industry experts, representing potential end-users within the sponsor company, play a central role in this validation. A demonstration video showcasing the DSS was presented to these experts, coupled with a semi-structured questionnaire designed to capture their perspectives on the DSS's capabilities. Both closed questions using Likert scale and open-ended questions are used to capture nuanced insights in the questionnaire.

The aim is to extract industrial feedback and explore the applicability of the DSS in the view of experts. The subsequent sections delve into the details of the validation methodology, participant demographics, and the analytical approach applied to extract meaningful results from the questionnaire.

6.1 Methodology

This section covers the methodology taken for collecting and analysing the expert feedback. This begins with an explanation of the demonstration video created for validation, the justification for chosen method of analysis, followed by an explanation of the semi-structured questionnaire design and how it was distributed.

6.1.1 The Video Demonstration

The demonstration of the DSS functionality came in the form of an 18-minute video which introduces the users to the project aim and objectives (as part of this research) and the definition of a DSS before demonstrating all main functionalities of the developed DSS and highlighting the potential development in the future. Specifically, this video covered the following topics, in order:

- **What is a DSS:** An explanation of what a DSS is and what are its key components*; the user interface, the inference engine*, and *the knowledge base*, as shown in Figure 6-1. These components will be programmed as main functionalities of the developed DSS.

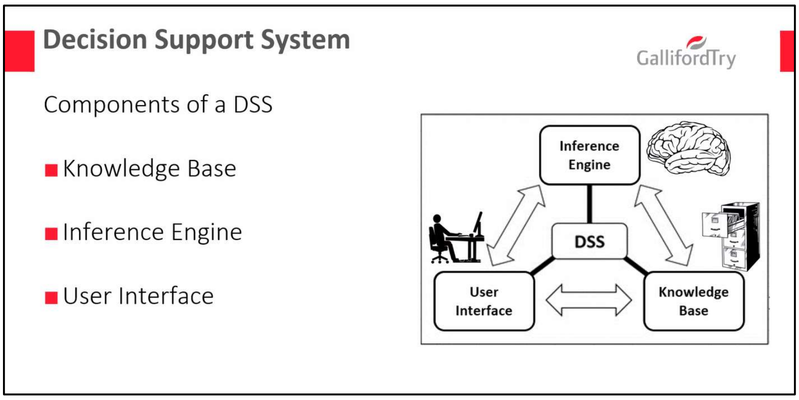


Figure 6-1 Components of a DSS

- **Project Overview:** An explanation of the collaborative project aim and objectives and highlighting the focus of enhancing performance in decision making with AI at this stage of development, as shown in Figure 6-2.

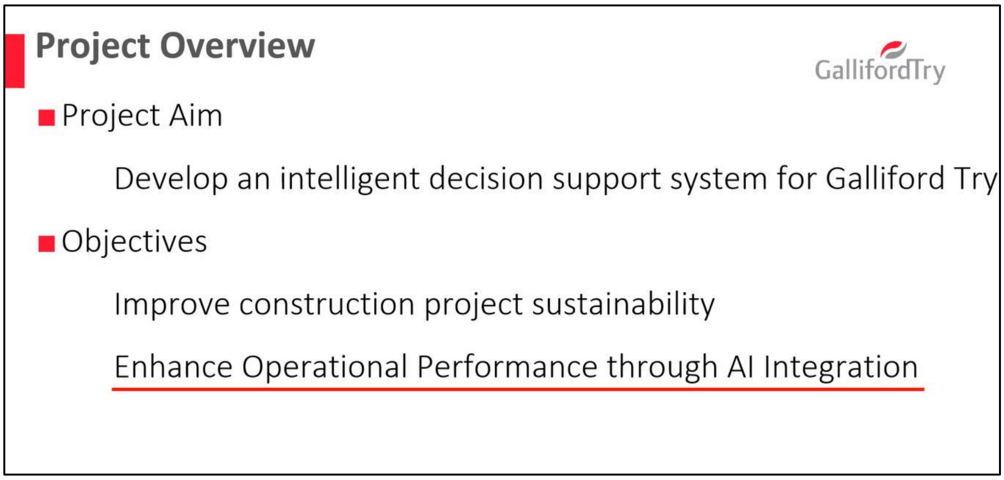


Figure 6-2 Collaborative Project Aim and Objectives

- **Contents Page and Performance Metrics:** The contents page, and the performance metrics page are first discussed, presenting the user interface, and providing an understanding of the metrics which all pages in the DSS evaluate the performance against.

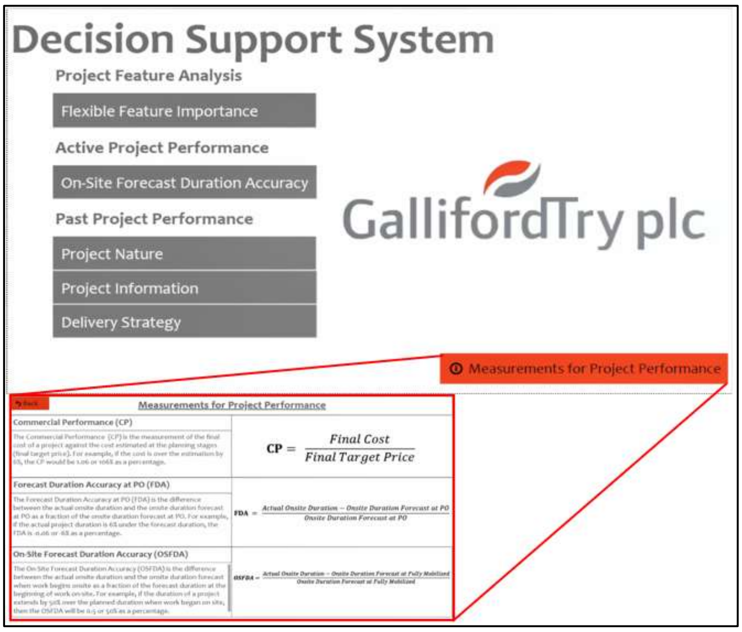


Figure 6-3 Contents page and Performance Metrics

- **Flexible feature importance Function:** All sections of the flexible feature importance page are discussed, and the uses demonstrated with an example of SHAP feature importance, as shown in Figure 6-4. An explanation of the optimised AI model and a demonstration and explanation of the results of PFI, LIME and SHAP importance is also provided.

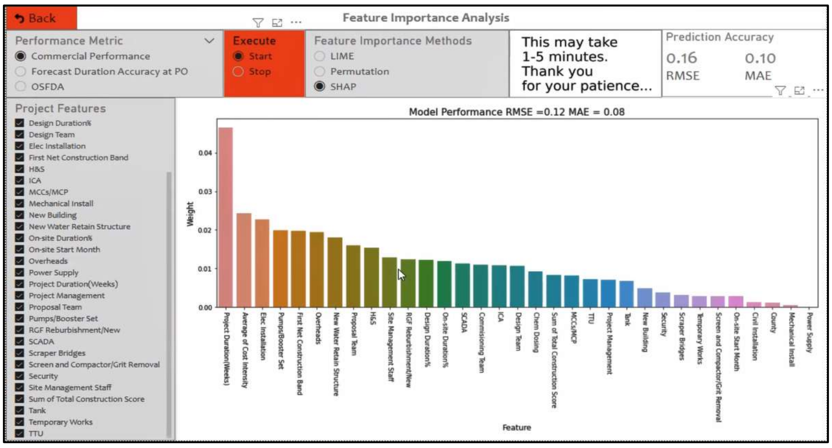


Figure 6-4 Flexible Feature Importance Page from Video Demonstration

- **Functionality of all other pages:** The functionality of the active project performance and the past project performance pages, as shown in Figure 6-5, are demonstrated.

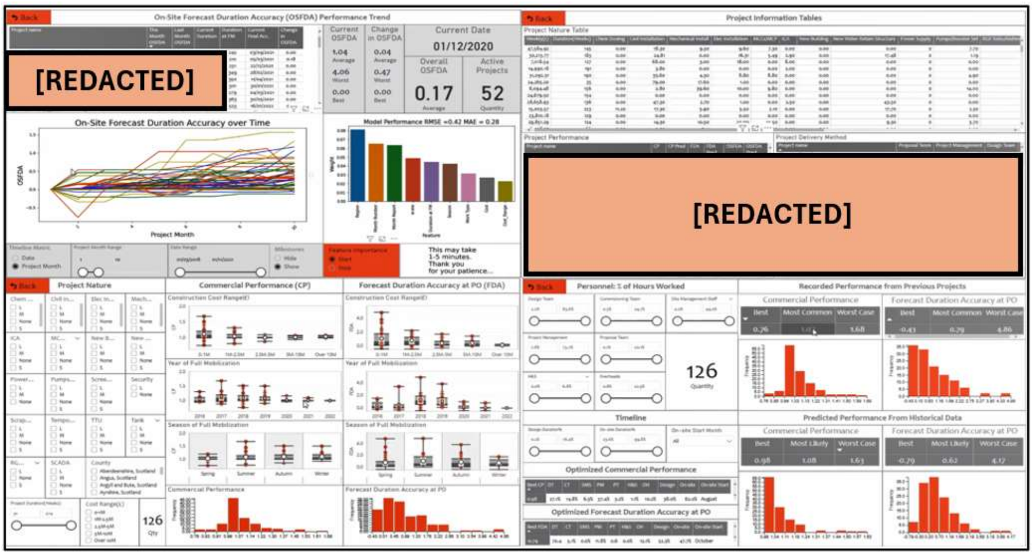


Figure 6-5 Active and Past Project Performance Pages of the DSS

- **Potential for the future:** The next stage of the DSS development is mentioned, as shown in Figure 6-6.

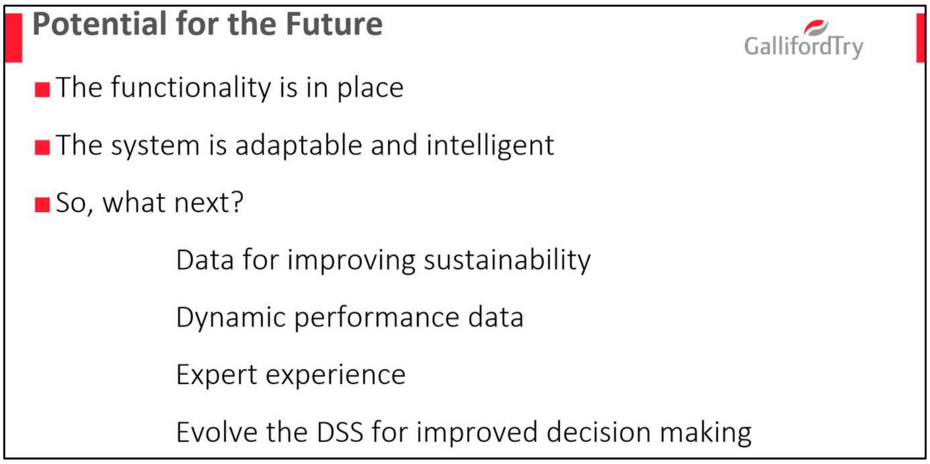


Figure 6-6 The next stage of the DSS development

6.1.2 Use of Questionnaire

The decision to employ a questionnaire for evaluating the industrial impact and efficacy of the DSS is driven by the advantages of collecting qualitative responses and acknowledges the limited available data at this stage of DSS development. This was also selected above the choice of interviewing persons from industry due to a lack of availability and time from the experts involved in the survey. The semi-structured questionnaire adopts a closed format, leveraging Likert scale questions, a methodology applied in previous studies related to decision-making in construction and sustainability (Shi et al., 2016, Yaseen et al., 2020, Murat Gunduz, 2021). This structured format facilitates quantifiable measurement of participants' perceptions, enabling straightforward comparison and statistical analysis of responses. The inclusion of open-ended questions enriches the analysis, providing participants with the freedom to share nuanced and qualitative insights from their unique perspectives.

The questionnaire method ensures a systematic and comprehensive approach to gathering feedback from senior industry experts, allowing for the extraction of insights grounded in significant industrial experience and practical considerations. Additionally, the questionnaire's versatility allows for the exploration of various dimensions, from the current capabilities of the DSS to its potential future applications, contributing to a holistic understanding of its usefulness in an industrial context.

A key limitation of the questionnaire is the subjective nature of the assessment which is mainly based on the opinions of experts rather than tangible results observed from a real-life implementation of DSS in an industrial setting. This limitation arises due to the current unavailability of sustainability data, with this research setting the foundation for the next stage of development. The subjectivity of this validation is mitigated by involving multiple senior experts with extensive experience in the construction industry. All participants bring diverse expertise in the construction sector, offering a well-rounded perspective on the applicability of the DSS.

6.1.3 Designing the Questionnaire

To systematically gather comprehensive feedback from industry experts on various aspects of the DSS, the questionnaire structure, as shown in Appendix 6-1, is designed to assess **the overall usefulness of the DSS, as well as its specific features, usability, and potential impact**. The questionnaire is divided into seven sections, as listed below.

1. **Brief -** The questionnaire begins with an overview, offering clarity to participants about the purpose of the validation.
2. **Demographics -** Participants are asked to provide key demographic information demonstrating their capacity to evaluate and understanding how their roles may benefit from the DSS.
3. **Understanding and Clarity -** This section assesses participants' engagement with the video and evaluates their comprehension of the demonstrated content.
4. **DSS User Interface -** This section assesses the visual appeal of the DSS interface and its effectiveness in facilitating users to interact with all DSS functions.

5. **Flexible Feature Importance Tool -** A key emphasis is placed on the feature importance tool, incorporating SELECT optimisation. Participants are queried on their understanding of the tool's functionality, the value attributed to various aspects (including prediction accuracy, metric optimisation, adaptability for future criteria), and their perceptions of its impact and usefulness.
6. **All functions in the DSS -** This section expands the focus to encompass the contribution of all functions within the DSS. It explores their collective impact on improving, potentially, construction project sustainability, relevance to industry needs, and integration with current systems. This section utilises both Likert scale quantification and qualitative open-ended questions to gather nuanced information from participants.
7. **Concluding Questions** - The concluding questions inquire about the likelihood of recommending the DSS, identify promising features, and solicit feedback on areas for improvement. The questionnaire closes by inviting any additional comments or suggestions from participants, underscoring the value of their input for research and future system development.

A pilot study involving one academic and one practitioner for this research was carried out. It aimed to gather feedback from both an academic and industrial perspectives by assessing the understandability, accessibility of files, and overall quality of the survey instrument. They scrutinised the questionnaire to ensure clarity and comprehension for the intended audience. Additionally, they evaluated its quality, considering factors such as relevance, coherence, and alignment with research objectives. The feedback served as a valuable quality assurance tool, identifying potential areas for improvement, and ensuring the questionnaire's robustness in alignment with the research goals. Key amendments from this process include the following, referring to Appendix 6-1:

Demographics: Responsibilities/duties question added to provide additional insight into the relevance of the participant’s experience from their work responsibilities as well as the duration.

Wording Clarity: The wording for Q2 was redefined for clarity from what is the ‘impression’ of the DSS to what is the ‘overall usefulness’. This new wording is better represented with the Likert scale.

Add reasons for answers: The addition of requesting reasons for the chosen answers for Q17, Q18 and Q19 to gain a larger insight into the overall impression of the DSS functionality.

Additional Question: The addition of Q20 to determine which of the participants are familiar with DSS technologies, providing a comparison to the developed DSS.

Distribution Method: The approach for distribution is agreed to be through a link to Google Forms for accessibility and ease of use for both the participants and the researcher.

These refinements contributed to a more polished and user-friendly survey, enhancing its effectiveness in collecting meaningful insights from participants.

6.1.4 Distribution of the Questionnaire

The distribution of the questionnaire was facilitated through **Google Forms** due to its user-friendly interface, accessibility, and efficient data collection and data management capabilities. This online approach streamlined the survey process, making it convenient for both researcher and participants, and eliminating the need for physical paperwork. Respondents could engage with the questionnaire at their own pace and convenience.

For distribution, an email containing the link to the demonstration video and the questionnaire, shown in Appendix 6-2, was sent to a designated employee within Galliford Try Ltd, who forwarded the same email to other relevant individuals. This targeted approach ensured a direct and tailored outreach.

6.1.5 Participants

Table 6-1 Participants of the validation questionnaire

Job Title	Duties/Responsibilities	Experience (Yrs)
Technical Manager	Technical elements of Design, support of the Engineering Team and Digital applications	33
Health, Safety and Environmental Systems Manager	Produced a Management System which is legally compliant, and assists those managing Health and Safety	29
Project Management Office Manager	Data Analysis, Operational Team Liaison, Concept Report Building, Operation Reporting, Framework Performance Review	22
Senior planner	Provide construction insights to project teams and management	20
Innovation and Research Lead	Responsible for innovation activity across the organisation.	20
Data & Systems Manager	Gather programme, commercial & safety data for the Environment business	12
Data Analyst	Project Management Office Overview	3
	Average	19.86
	Standard Deviation	10.06

Table 6-1 shows an average working experience of 19.86 years with a standard deviation of 10.06 years over a total of 7 participants. To increase diversity, feedback was collected from professionals with a diverse range of experience, including mostly those with at least 20 years of expertise in the construction industry, one having more than 10 years of experience and one having only 3 years of experience.

6.2 Results and Discussion

The closed questions using Likert scale will be first examined followed by the analysis of the open-ended questions.

The Likert scale used for each closed question, as shown in Appendix 6-3, contains 5 levels of rating, 1 being the worst and 5 being the best. Appendix 6-3 shows the distribution of responses across 5 different scales as well as the average result and standard deviation. Overall, the feedback is good, with all questions having a mean feedback value ranging between 3 and 5, with the standard deviation between 0.64 and 1.05. The results specific to each section of the questionnaire will be discussed below.

6.2.1 Video Demonstration Feedback

Table 6-2 Video Demonstration Feedback

	Numbered Likert Scale					**Summary**	
Description	1	2	3	4	5	Average	Standard Deviation
What is the overall usefulness of the DSS based on the video?			1	4	2	4.14	0.64

The participants generally found the DSS have a high level of usefulness (average = 4.14, SD = 0.64) from the video demonstration, with the Innovation & Research Lead and the Project Management Office (PMO) Manager scoring the top rank of 5 to this question. As they are both key persons with significant experience related to project management and research this is a positive outcome. The most common result was 4, with only the Health, Safety and Environmental (HS&E) Manager scoring 3, believing the DSS to be moderately useful. It is reasonable to tell why the HS&E Manager may find the DSS less useful at present as it has not taken environmental or safety metrics into consideration due to the lack of sustainability data. Having said that, the DSS is equipped with the capability to improve project sustainability performance in the future.

6.2.2 Understanding and Clarity

Table 6-3 Understanding and Clarity

	Numbered Likert Scale					Summary	
Description	1	2	3	4	5	Average	Standard Deviation
After watching the video, how well do you feel you understand the key features and functionalities of the DSS?			2	3	2	4.00	0.76
How clear and easy to follow were the explanations and demonstrations in the video?			2	2	3	4.14	0.83

There was a generally good understanding of the key features and functionalities of the DSS, as represented in Table 6-3. All responses ranged between a score of 3 to 5 with the average score of 4 and 4.14 for the understanding of the DSS functionalities and the clarity of the explanations in the video demonstration. This adds significance to the findings as the participants understand what they are evaluating based on the demonstration video.

6.2.3 User Interface

Table 6-4 User Interface

	Numbered Likert Scale					Summary	
Description	1	2	3	4	5	Average	Standard Deviation
How would you rate the ease of use of the DSS's user interface based on the video demonstration?			1	5	1	4.00	0.73
Did you find the DSS's user interface visually appealing and well-organised based on the video demonstration?			4	1	2	3.71	0.88

The user interface was received well, with all responses in the range between 3-5, as shown in Table 6-4. The interface scored highly for the ease of use with an average score of 4 and a standard deviation of 0.73, of all participants, 86% scored 4 or above, suggesting that the DSS interface is perceived as easy to operate. The visual appeal and page organisation also scored highly, but 57% of the participants scored 3 for this, suggesting an area for improvement in the presentation of the DSS.

6.2.4 Flexible Feature Importance Function

Table 6-5 Flexible Feature Importance Function

	Numbered Likert Scale					Summary	
Description	1	2	3	4	5	Average	Standard Deviation
Did the video demonstration provide a clear understanding of how the Feature Importance Tool works within the DSS?			3	2	2	3.86	0.83
How well do you understand the role and significance of the Feature Importance Tool based on the video?			2	3	2	4.00	0.76
How valuable do you consider the Tool's functionality for optimising project performance in construction projects?		1		4	2	4.00	0.93
How valuable do you consider the Tool's functionality for optimising for each performance metric?			1	4	2	4.14	0.64
How valuable do you consider the adaptability of the Feature Importance Tool for future datasets and metrics?			1	2	4	4.43	0.73
How confident are you in the accuracy of the Feature Importance Tool's assessments based on the video demonstration?			4	2	1	3.57	0.73
To what extent do you believe the tool can help project managers make informed decisions for project performance?		1	1	4	1	3.71	0.88

The Flexible Feature Importance function incorporates the SELECT method for optimising against each performance metric, so this section of the survey contains more questions which were purposefully designed to assess the usefulness of this function from the perception of the participants.

The results are positive overall, with the scores for all questions having an average ranging between 3.57 and 4.43, shown in Table 6-5. The most promising responses are in relation to the optimising capability and adaptability for future data, with average scores of 4.14 and 4.43 respectively. These aspects of the Flexible Feature Importance function, key contributions from this research, are perceived to be valuable by the participants; experts in decision making in the field of construction project management.

The first question in Table 6-5 is related to understanding the Flexible Feature Importance functionality based on the video. The average response is 3.86 and 57% of participants scored this as 4 or above, showing most participants at least finding the functionality to be clear from the video demonstration. The role and significance of the function is also well understood as 71% of applicants scored this at 4 or above. These findings can give confidence to the subsequent questions related to this function through a general understanding of the tool's functionality and significance.

86% of participants believe that this function is valuable for improving construction performance with the PMO Manager and Innovation & Research Lead both scoring this the highest value of 5. This is a very positive result as it shows a general appreciation of the feature importance function for the technical personnel with experience, particularly those who are responsible for construction project management and those whose job is to bring innovation into the construction environment. The HS&E Manager continues to assign a lesser score than the other participants, assigning 2 in optimising construction projects. Again, this may be due to the lack of sustainability measurement which is caused by the data unavailability.

In relation to the optimisation for different performance metrics, as discussed previously, the results are very positive. The vast majority, 86%, of responses consider the optimisation for new metrics to be 4 or above, with PMO Manager and Innovation & Research Lead once again giving a score of 5. This same result is mimicked for the adaptability for new datasets in the future. The technical participants have consistently given scores of 4 and 5, while the HS&E Manager has reacted less positively, rating this functionality between 2 and 3, for optimisation and future data adaptability respectively.

The respondents have scored from 3 to 5 in confidence to the accuracy of the feature importance function, with the PMO Manager scoring the highest ranking of 5, and the Senior Planner and Innovation & Research Lead scoring this at 4. All other participants have scored 3 for confidence in the predictive accuracy. In terms of helping project managers make informed decisions for project performance, the responses are mostly positive with an average score of 3.71 and 71% of participants scoring 4 or above. The PMO Manager scored 5, the Data and Systems Manager responding with a 3 and the HS&E Manager responded with a 2.

Overall, there is generally a positive perception of the impact of this function, with key experts such as the PMO Manager and the Innovation & Research Lead having the highest confidence in the capability and future potential of the system.

6.2.5 Usefulness of the DSS

Table 6-6 All Tools in the DSS Q14-Q17

	Numbered Likert Scale					Summary	
Description	1	2	3	4	5	Average	Standard Deviation
Based on the demonstration, how well do you think the DSS can assist in improving construction project sustainability?		2	1	3	1	3.43	1.05
How relevant do you find the DSS's functionalities to the construction industry's needs and challenges?			1	4	2	4.14	0.64
Based on the demonstration, how well do you think the DSS integrates various tools to support decision-making for construction projects?		1	1	4	1	3.71	0.88
How likely would you be to recommend the DSS, based on the video demonstration, to colleagues or peers in the construction industry?		1	1	3	2	3.86	0.99

Regarding how well the DSS integrates with other tools for supporting decision-making in construction projects, 71% of participants scored 4 and above with the PMO Manager scoring 5 as feedback. The Data and Systems Manager scored 3 while the HS&E Manager scored 2. The distribution of responses shows that the capability of the DSS to integrate with other tools/systems within the company is well-recognised except that the HS&E Manager expected more from the DSS.

The DSS functionalities were also believed to be relevant for the needs and challenges of the construction industry. Scores ranged from 3 to 5, averaging at 4.14 with 86% of participants scoring 4 or above.

There was more of a mixed reception for the evaluation of sustainability criteria. The mean result for the perceived capability for improving construction project sustainability was 3.43 and a standard deviation of 1.05. This suggests a perceived moderate contribution to sustainability with diverse opinions. This is due to scoring of 2 from the Data analyst and Data and Systems Manager, while the HS&E Manager scored 3 for the capability for improving construction project sustainability. This spread may be in relation to the fact that no environmental or social goal criteria or performance metrics were included in the DSS at present, providing area for improvement in the future. Despite this feedback, the PMO Manager scored a 5 for this question. Overall, 57% of participants scored this question with 4 or above so most participants believe the DSS can improve construction project sustainability, while others are less convinced.

When asked the likelihood of recommending the demonstration to colleagues and peers in the construction industry, 71% of participants responded with 4 or above as a score. Both the PMO Manager and the Innovation and Research Lead scored this with a 5 while the HS&E Manager once again responded with a lesser response of a 2.

6.2.6 Open Ended Responses

Q17 Reasons for response to DSS recommendation

The reasons given for the scores on Q17 question are provided in Appendix 6-4. The key observations made from these reasons are shown in Table 6-7:

Table 6-7 Reasons for decision to recommend the DSS

User	Rank	DSS Recommendation: Reasons for selected Ranking
Data & Systems Manager Technical Manager	3 4	There would be benefit of further data for improving system performance.
Data & Systems Manager Innovation and Research Lead	3 5	Suggest future collaboration to improve and harness the tool.
PMO Manager	5	The DSS takes away perception, which is still guiding the construction industry.
HS&E Systems Manager	2	Tools like this may not accommodate the diverse forms and nuances of project variables to be effective, and fears people will stop thinking and rely on such software, resulting in dangerous situations

These differing reasons show the nuance and diverse perspective presented by the participants of the study. There is a clear desire to develop this DSS further with more data as multiple participants can see the potential and benefit from incorporating such a tool into their field. The benefits of removing the subjective nature of decision making guiding the construction industry are a possible driving factor for this.

Not all persons are convinced of the capability of the system. The comments of the HS&E Manager show a concern for missed information in decision-making. This does however miss the point that a DSS is a support system, rather than one that replaces the expert knowledge. This is a point which could have been emphasised further to reduce the resistance to this technology. This comment does bring to light an understanding of why the HS&E Manager responded with the harshest scoring of all participants, averaging 2.86 over all closed questions, compared to average of 3.93 between all participants.

Q18 Which function is most promising and valuable, and why?

The responses to Q18 are shown in Appendix 6-5. The key observations made from the responses are shown in Table 6-8:

Table 6-8 Observations of the most promising and valuable DSS functions

User	Key Observations
Senior Planner PMO Manager	The active project performance tool for trend prediction is of most benefit.
HS&E Manager	The active project trend prediction tool to be of benefit but has reservations about the consistency of data and the method of implementation.
Data & Systems Manager	There is benefit of the project nature tool in studying projects by their defining characteristics, providing insights which may lead to improved efficiency, quality, and performance.
Data Analyst	Both the flexible feature importance function and active project performance tool to be the most promising.
Technical Manager Innovation and Research Lead	The DSS itself is the most value, with the future potential of the whole system.

A trend that can be noticed throughout the feedback is that each of the participants tends to value the tools which benefit their role the most. The Planner and PMO manager prefer the active project performance, the data analyst values both the feature importance and active project performance functions, and the Data and Systems Manager prefers the tool which studies the nature of past project data. The Innovation and Research Lead can see the benefit of all tools, and the Technical Manager can see the potential for future development.

The combined input of all applicants highlights the importance of considering the specific roles of all stakeholders and responsibilities when evaluating the perceived value and benefit of different tools within the DSS. It also highlights the need for tailored solutions that address the unique requirements and priorities of various stakeholders within the organisation. There is also clearly an enthusiasm within this expert pool to pursue future collaborative development.

Q19 Which functions is may need further improvements, and why?

The responses to Q19 are in Appendix 6-6. The key observations made from these responses are listed in Table 6-9:

Table 6-9 Feedback for functional improvements

User	Functional Improvements
Senior Planner	Including data related to external factors which would impact project performance.
Data and Systems Manager	Adding contemporary data for understanding of the current situation and trends.
Data Analyst	A difficulty in defining the optimal parameters in the project nature page, requiring complex analysis of the data to achieve this.
Technical Manager	Including risk data overlayed with time and money.
Innovation and Research Lead	Including the types of work, forms of contract and procurement methodology.

The most prominent trend in the feedback from all participants is the desire for increased data. Whether it is related to risk, external factors or contemporary projects, the focus of improvement is on the introduction of new data. This, combined with the previously established enthusiasm for future collaborations, shows a positive direction for further development of this DSS, and a trust in the capability.

There was no concern mentioned in relation to functions of each page except for the defined parameters on the project nature page, stated by the Data Analyst. A point to mention in relation to this, is the desired optimal parameters for the project nature page would be an exact application of the Flexible Feature Importance function. This is intended for finding the optimum CSFs against construction project performance and would handle the complexity stated by the Data Analyst. This would provide avenue of improvement would be a suitable area for future research.

Q20 Awareness of similar DSS software and comparison?

The response to similar tooling is shown in Appendix 6-7 and the key observations are shown in Table 6-10:

Table 6-10 User awareness of similar DSS software

User	Awareness of Similar DSS Software
Data and Systems Manager PMO Manager Innovation and Research Lead	Have not heard of a similar tool.
Technical Manager Senior Planner	Are aware of support technologies focused on improvements at the design stage.
Data Analyst	Refers to Primavera but mentions the proposed DSS has an advantage of gaining detailed insights from project characteristics.
Senior Planner	Knows of Oracle Analytics Cloud

It is clear from the feedback related to Q20 that most of the participants are not familiar with similar tools to the proposed DSS, and there are not sufficient comments related to comparing existing technologies with the DSS.

There is the mention of the design-based support technologies which support a different application. The Primavera tool is mentioned for project management but not for the capability of gaining detailed insights from project characteristic. Oracle Analytics Cloud is highlighted, but this tool does not utilise optimised deep learning to detect complex patterns in data.

Q21 Is there any additional feedback or comments?

The responses to Q21 are shown in Appendix 6-8 and the key observations are shown in Table 6-11:

Table 6-11 Additional user feedback

User	Additional Feedback
Data Analyst	Mentioned there was no data related to sustainability in the system at present.
PMO Manager	Complimented the demonstration on the clarity and provided information.
Technical Manager Innovation and Research Lead	Expressed an interest in assisting with the future development of the DSS.

The demonstration did mention that one of the goals was to improve construction project sustainability, but it was also highlighted that the focus would be on developing the DSS functionality due to a lack of sustainability data. The positive feedback from the PMO Manager, and the Technical Manager and Innovation and Research Lead wanting to assist in the future is evidence of the positive impact and potential of the DSS now and for the future.

6.3 Summary

This chapter covered the industrial validation of the DSS through the distribution of a well-justified survey combined with a video demonstration. The video presented the overall functionality of the DSS, specifically mentioning the integrated SELECT optimisation algorithm and data adaptability related to the feature importance function. The validation involved 7 experts in the field of construction project management, with an average of 20 years of experience between them. All these participants came from a wide range of expertise and responsibilities providing a diverse perspective to assess the DSS efficacy and impact. The key findings are listed in Table 6-12:

Table 6-12 Key findings from the validation survey

Key Finding
Overall, the feedback is good, with all questions having a mean Likert value ranging between 3.00 and 5.00, with the standard deviation between 0.64 and 1.05.
The flexible feature importance tool which incorporates the SELECT algorithm was believed to be of value with an average score of 3.95 over all related questions.
The functions for optimisation and data adaptability were of significant value with the average scores of 4.14 and 4.43 respectively.
71% of the participants scored 4 or 5 for recommending the DSS to colleagues and peers.
The PMO Manager and Innovation and Research Lead had the most positive reaction with all rankings between 4 and 5.
The HS&E Manager was the most resistant of the participants to the benefit of the DSS, showing concerns that mistakes may be made by relying too heavily on the DSS. This negative opinion may have arisen from the lack of sustainability data included in the study.

These findings provide clear evidence to support the efficacy and impact of the DSS especially the potential benefits of implementing the flexible feature importance function in an industrial setting. The DSS has been well-received in the validation as the majority of the participants have shown strong desire for collaboration and further development.

7 Conclusion

This chapter serves as the culmination of the thesis, covering the aims of the study and the key discoveries and contributions, highlighting the implications and significance of the completed work, and the opportunities for future research.

7.1 Research Gap in the Literature and Aim

Previous studies have focused on using meta-learning to predict learning curves on new datasets by studying previous datasets (Wistuba and Pedapati, 2020, Klein et al., 2017). Moreover, existing approaches to learning curve prediction have concentrated on halting poorly performing learning curves (Domhan et al., 2015). Previous research has presented findings that learning curve prediction of this kind would be difficult to harness effectively for improving HPO for deep learning. (Choi et al., 2018) carried out a study on learning curve prediction for the early termination of learning curves. They concluded that the shape of learning curves changes drastically depending on both the hyperparameter configurations and the variations in dataset. They also highlighted that additional tuning parameters make it challenging to effectively use learning curve prediction for variations in HPO task.

This has left a gap for an approach that can incorporate both the training and prediction of learning curves on the same dataset, allowing for the prediction of the performance of fully unseen learning curves based on training a subset of the hyperparameter search space.

The aim of this study was to advance the field of hyperparameter optimisation (HPO) and learning curve prediction by developing an innovative approach that overcomes existing limitations in current methodologies.

7.2 Objectives

To achieve this overarching aim, the specific objectives of this research included:

1. **Overcoming Limitations in Learning Curve Prediction:** Address the constraints of existing learning curve prediction methods with a more integrated framework that utilises both training and prediction within the same dataset.
2. **Creating a New HPO Approach:** Introduce a novel hyperparameter optimisation technique that leverages the newly developed learning curve prediction model, enhancing predictive accuracy and efficiency.
3. **Validating Against Existing Benchmarks:** Conduct comprehensive validation of the new HPO method against established benchmarks to demonstrate its effectiveness and reliability in practical applications.
4. **Integrating the HPO Method into a Feature Importance Analysis Tool:** Develop a tool that combines the HPO method with feature importance analysis techniques, enabling users to gain insights into the critical factors affecting model performance.
5. **Demonstrating Industrial Significance in a Decision Support System (DSS):** Showcase the applicability and relevance of the developed methodologies within

a practical DSS, highlighting their potential to contribute to informed decision-making in real-world scenarios.

7.2.1 The SELECT HPO Method

The SEquential LEarning Curve Training (SELECT) HPO algorithm has been developed in the form of the data pipeline as shown in Figure 7-1. The stages of the SELECT method are as follows:

1. Each dataset inserted into the pipeline is split into a training, validation, and test set, and split between labels and input features.
2. The learning rate is tuned using the training and validation sets to allow negligible difference between all hyperparameter configurations at the start of each training run, leading to a minimum by the end of a predefined epoch limit.
3. A representative sample group of hyperparameter configurations are trained, and the learning curves are recorder, converted into blocks and joined in sequence to create a training set for a sequence prediction model, in this case the CGRNN.
4. The CGRNN is trained with the sequential training set to learn the relationship between the hyperparameters and the learning curve shape.
5. The CGRNN predicts the learning curves with a single training window, without training them, and the predictions are ranked by the best results from the final steps in each learning curve, referenced in the green box in Figure 7-1.
6. The top predicted configurations are trialled to select the best model on the test set.

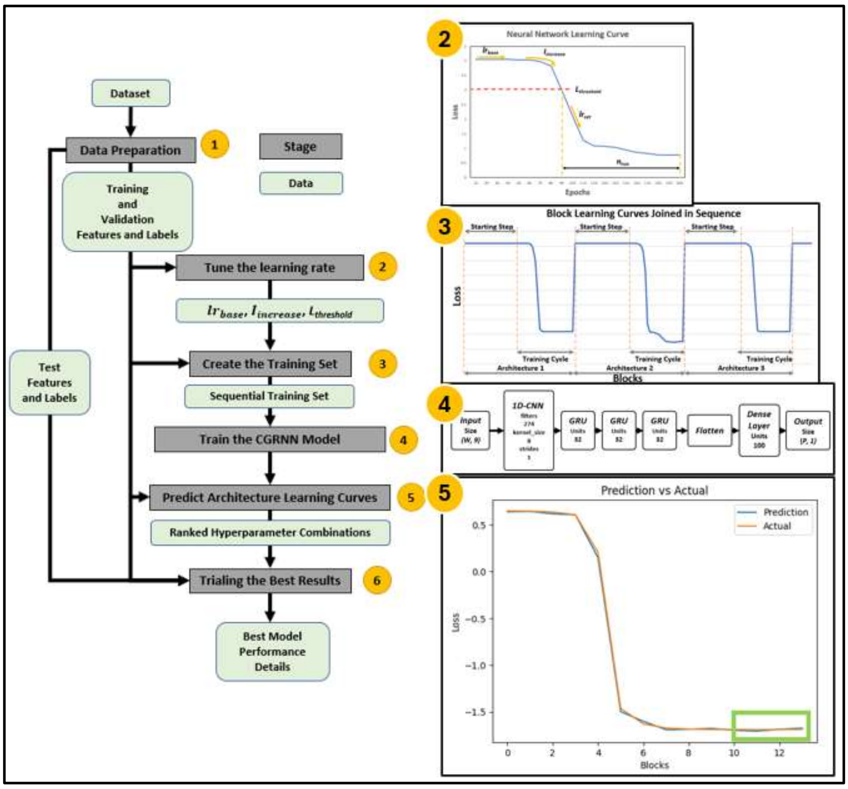


Figure 7-1 Novel HPO Process Flowchart

This thesis presents a learning curve prediction-based algorithm which achieves HPO with a high level of accuracy and better computation efficiency than GPBO, TPE, Hyperband and RS. This is on both synthetic and real datasets, evaluated through MAE and feature importance analysis. The SELECT method not only improves an HPO algorithm with learning curve prediction such as (Klein et al., 2017, Wistuba and Pedapati, 2020), but also re-defines learning curve prediction as a completely new approach to HPO which can predict the learning curves of hyperparameter configurations without running them, i.e. **completely unseen hyperparameter configuration learning curves, for the first time, can be predicted**.

The SELECT method contains tools which have been used in the past, such as the CGRNN. The CGRNN was selected through a trial-and-error comparison with the RNN, LSTM, GRU and 1DCNN, as well as hybrids of these. The combination of the GRU and 1DCNN proved the most effective in predictive accuracy and computational expense for training.

The SELECT method is also composed of a novel approach to creating a sequential training set for the CGRNN which makes it possible to predict fully unseen learning curves. This is the combination of three main components, shown in Figure 7-2. Each of these components achieves a key function to overcome the well-known difficulties in learning curve prediction for HPO(Choi et al., 2018).

- A loss dependent learning rate, which begins with a negligible impact on training an ANN and adapts to each configuration individually.
- The conversion of each learning curve into blocks of the average and range of loss, validation loss and the learning rate in parallel, rather than every epoch in series.
- The addition of a 'Starting Step' made of synthetic duplicates of the first instance of the learning curves.

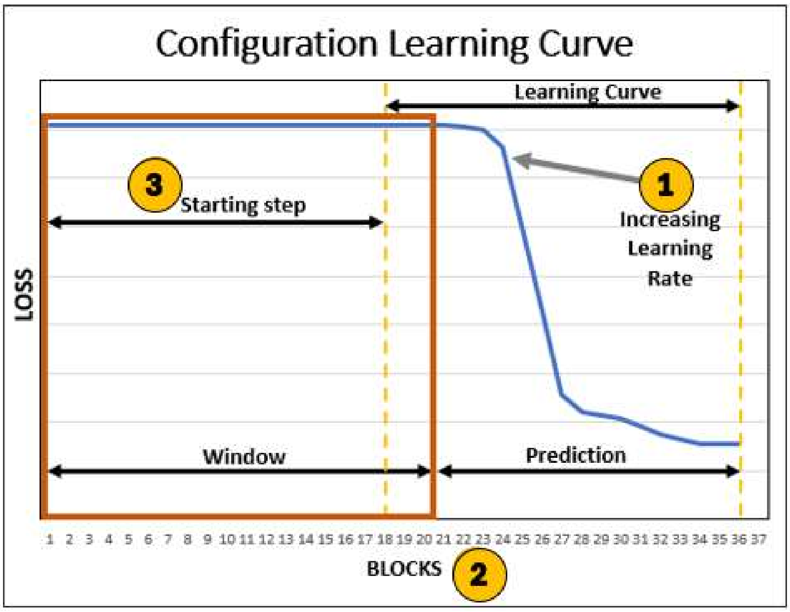


Figure 7-2 Three components for efficient learning curve prediction for HPO

By converting learning curve data into the format shown in Figure 7-2 and joining the learning curves as a sequence, shown as a line graph in Figure 7-3, a single optimised CGRNN can be trained to adapt to different datasets and predict unseen learning curves.

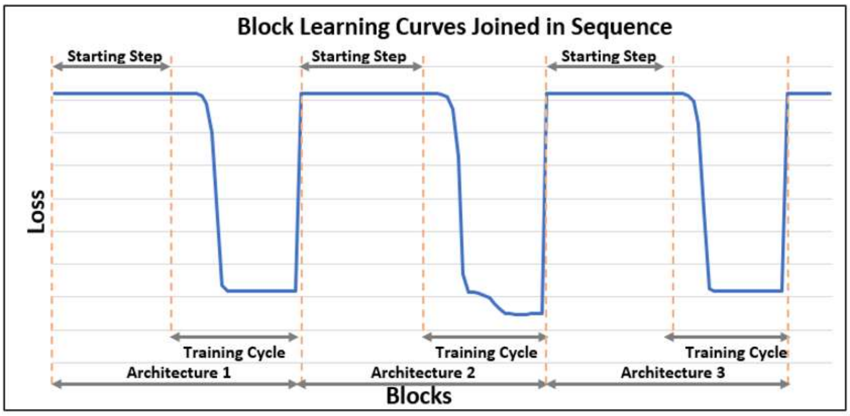


Figure 7-3 Example of the sequential training set as a line graph

With this developed sequential training set, the number of rows, the number of variables, the type of variables, and the range of feature scaled parameters are the same for whatever dataset enters the pipeline shown in Figure 7-1. The only variation is the relationship between the chosen prediction variable and the hyperparameters. This means the CGRNN does not need to be tuned for new datasets.

Additionally, all learning curves in this format begin with the exact same shape; this is the synthetic starting step and negligible difference in the learning curve data. So, this shape can be replicated with each hyperparameter configuration and inserted into the trained CGRNN to predict the learning curve of that configuration without any necessary training.

The combined steps in Figure 7-1 present a new method of HPO adopting learning curve prediction which is comparable to any standard machine learning pipeline, rather than the convergence seen in BO, HB or TPE. Most existing HPO methods iteratively observe configuration performance and converge on the best option through consecutive evaluations. The SELECT method prepares a sequential training set from a sample of the search space, trains a CGRNN and predicts the best results. This presents multiple additional benefits beyond finding high performing configurations. These additional benefits are listed below, from the Results and Discussion in Chapter 3 (section 3.5.4):

- **The trained iterations in the sequential training set do not interact:** The learning curves for the sequential training set can all be trained in parallel.
- **The pipeline has a set amount of computation for completion:** The optimum outcome is achieved after a predefined amount of computation for every dataset, rather than a termination criterion. This creates a level of consistency which is

beneficial for resource allocation and predicting the time needed for the optimisation process.

- **The relationship between hyperparameters and performance can be visualised:** The predictions from the CGRNN can be used to analyse the relationship between hyperparameters and performance, directing the search to include or exclude areas of the hyperparameter search space for better performance.

7.2.2 Feature Importance Analysis

This study shows the superior capability of the SELECT HPO method in predictive accuracy and determining the importance of features in data using both synthetic and real-world datasets. The SELECT method has outperformed existing HPO methods in identifying top-performing ANN models and uncovering complex dependencies within data, even under varying degree of complexity and uncertainty. Further to this, the novel HPO method has been effectively integrated into a DSS, combined with SHAP, PFI and LIME to create a holistic and objective way of finding the CSFs for sustainability in construction.

7.2.3 DSS Validation

The DSS was validated using a video demonstration and survey focusing on the functionality of the developed feature importance tool integrated into a DSS together with the novel HPO method. This validation involved 7 experts from the construction field with an average of 20 years of experience and a diverse background of expertise.

The feedback from industry experts, with an average score of 4.14 out of 5 and SD=0.64 for the overall usefulness of the DSS, indicates a high level of satisfaction with the system's capabilities. Specifically, the feature importance function received positive feedback, particularly regarding its optimising capability (mean=4.14, SD=0.64) and adaptability for incorporating new data (mean=4.43, SD=0.73).

Most participants expressed confidence in recommending the DSS to colleagues and peers (71% scored 4 or 5), but certain concerns were raised about the potential for over-reliance on the system and the need for further development through the addition of new data. There was also enthusiasm from multiple participants in the survey to pursue future collaboration in developing the DSS, incorporating new data.

Overall, the validation confirmed the promising potential of the DSS integrated with the novel HPO method for supporting decision-making in the construction industry.

7.3 Research Contributions

Multiple contributions have been made in this research. These contributions are as follows:

A Novel Method of Learning Curve Prediction for HPO:

A novel method has been developed to create a sequential training set for predicting learning curves of fully unseen data. This method, for the first time, addresses the challenges related to the significant variations in learning curves for new datasets and hyperparameter configurations. This contribution is combined into a machine learning pipeline which culminates in a new HPO method. This novel HPO method can achieve high predictive accuracy, computational efficiency, as well as computational consistency.

Further to this, this method is able to select high performing models which find the important relationships with data, verified through thorough analysis.

This novel method contributes significantly to the field of HPO by enabling a more effective and efficient application of learning curve prediction. The impact of this novel method is explicitly demonstrated through all experiments in this study.

Objective Approach to CSF Analysis for DSSs in Construction:
An objective approach to feature importance analysis has been developed specifically for DSSs in the construction sector. This approach provides an unbiased method for evaluating the importance of features in decision-making processes within the construction industry. This function is optimised with the SELECT HPO method and has been created for adaptability the analysis of new data. By enhancing the reliability and effectiveness of feature importance analysis, this contribution empowers decision-makers to make informed and data-driven decisions in construction projects.

7.4 Limitations

There are several limitations to this research which are broken down in different areas as follows:

7.4.1 Novel HPO Method (SELECT)

The hyperparameter search space was intentionally restricted to a subset of hyperparameters commonly found in ANNs, which were **the learning rate, the number of neurons per hidden layer and the number of hidden layers**. Other hyperparameters such as the optimiser, activation function, and number of epochs could be analysed, however, for the sake of simplicity, only the primary factors driving model performance were considered in this research.

Only **Adam optimiser** was selected during this study as it created a fair comparison between the benchmarks as this approach includes an adaptive learning rate at the beginning. The number of epochs for all experiments was defined through trial-and-error to balance time and effectiveness. The activation function was set to ReLu, which is standard for regression problems, but other variations of ReLu are available, such as Parametric ReLu, Flexible ReLu and Leaky ReLu (Apicella et al., 2021).

Tabular datasets with matrices of rows and columns were the only type used for the validation of the SELECT method, this includes both the real world and synthetic datasets used in all experiments in Chapter 3 and Chapter 4. The analysis of datasets used for image classification in a different format has yet to be investigated with this approach.

7.4.2 Feature Importance Analysis

The feature importance analysis conducted in this thesis was constrained by the limited range of techniques employed. While the use of three well-known methods, **SHAP, PFI, and LIME,** provided valuable insights into feature contributions, the reliance on these specific methods may have restricted the breadth of the analysis results. However, the choice of these three techniques was primarily driven by their widespread application and well-documented justification in the literature, which facilitated easier implementation and interpretation.

The limited number of datasets used in the feature importance analysis represent another notable limitation. With two real-world datasets and two synthetic datasets, the generalisability of the findings may be limited. Incorporating more datasets, and different types of datasets such as image classification could generate more evidence to justify the effectiveness of the SELECT method for feature analysis.

7.4.3 Data Availability

The current research had to deviate from the initial aim of improving construction project sustainability as the sustainability data failed to materialise through the research journey. This has resulted in the DSS not achieving the capability to enhance project sustainability, but it can adapt to new data in the future when available.

7.4.4 DSS Validation and Sample Size

The validation of the DSS conducted in this study is of subjective nature, and so a more objective validation, such as real-world implementation, may help to provide more tangible evidence of the impact of the system. The limited number of participants, 7 in total, may also impact the reliability of the DSS validation. Although the sample was small, the validation involved professionals with an average experience of 20 years, it also encompassed diverse perspectives from individuals holding various roles, including **Technical Manager, PMO manager, HS&E Manager, Data Analyst, Senior Planner, Data and Systems manager, and Research and Innovation Lead**. This diversity of perspectives contributed to a comprehensive evaluation of the DSS.

7.5 Opportunities for Future Research

Multiple avenues for future studies have arisen during this research. These can be split between the future development and testing of the novel HPO method, and the development of the DSS.

The current research tested the novel HPO method on **tabular datasets and regression-based problems**. The method could be further tested on time-series datasets, or image datasets and for classification problems to further validate this approach in a wider scope of application.

Another future area of study would be to expand the search space to include the **optimiser, activation function, and number of epochs**. This, combined with the use of **parallel computing** would provide a great opportunity to test the novel HPO method on a larger scale, while potentially reducing the time required through an upgrade in computation.

The next stage of development for the DSS would be through the collaboration with the same sponsor company to **collect contemporary dynamic performance data, and sustainability data**. The specifics of the data will need to be determined through consultations with experts in the field of construction project management. The participants from the survey have highlighted an interest in pursuing the next stage of development and the initial communications for this have already begun. The aim of this collaboration would be to gain the knowledge from the experts around the CSFs for construction sustainability, utilising the flexible feature importance function. This will help to facilitate the incorporation of the DSS into the existing decision-making process of the company, hence create a demonstrable real-life impact in the construction industry.

References

Fibonacci number.

The UCI Machine Learning Repository [Online]. Kolby Nottingham. Available: https://archive.ics.uci.edu [Accessed 18/11/2023 2023].

2020. Developing an Ensemble Predictive Safety Risk Assessment Model: Case of Malaysian Construction Projects. *International Journal of Environmental Research and Public Health,* 17**,** 8395.

ABBAS, J. & MYUNGHO, L. 2023. Comparative Performance Evaluation of State-of-the-Art Hyperparameter Optimization Frameworks. *The transactions of The Korean Institute of Electrical Engineers,* 72**,** 607-620.

ABIOYE, S. O., OYEDELE, L. O., AKANBI, L., AJAYI, A., DAVILA DELGADO, J. M., BILAL, M., AKINADE, O. O. & AHMED, A. 2021. Artificial intelligence in the construction industry: A review of present status, opportunities and future challenges. *Journal of Building Engineering,* 44**,** 103299.

ADETUNJI, A. B., AKANDE, O. N., AJALA, F. A., OYEWO, O., AKANDE, Y. F. & OLUWADARA, G. 2022. House Price Prediction using Random Forest Machine Learning Technique. *Procedia Computer Science,* 199**,** 806-813.

AKBARI, S., KHANZADI, M. & GHOLAMIAN, M. R. 2018. Building a rough sets-based prediction model for classifying large-scale construction projects based on sustainable success index. *Engineering, Construction and Architectural Management,* 25**,** 534-558.

AKINOSHO, T. D., OYEDELE, L. O., BILAL, M., AJAYI, A. O., DELGADO, M. D., AKINADE, O. O. & AHMED, A. A. 2020. Deep learning in the construction industry: A review of present status and future innovations. *Journal of Building Engineering,* 32**,** 101827.

ALAVI, B., TAVANA, M. & MINA, H. 2021. A Dynamic Decision Support System for Sustainable Supplier Selection in Circular Economy. *Sustainable Production and Consumption,* 27**,** 905-920.

ALEX, D. P., HUSSEIN, M. A., BOUFERGUENE, A. & FERNANDO, S. 2010. Artificial Neural Network Model for Cost Estimation: City of Edmonton’s Water and Sewer Installation Services. *Journal of Construction Engineering and Management,* 136**,** 745-756.

ALTMANN, A., TOLOŞI, L., SANDER, O. & LENGAUER, T. 2010. Permutation importance: a corrected feature importance measure. *Bioinformatics,* 26**,** 1340-1347.

ANDONIE, R. 2019. Hyperparameter optimization in learning systems. *Journal of Membrane Computing,* 1**,** 279-291.

ANDÚJAR-MONTOYA, M. D., GILART-IGLESIAS, V., MONTOYO, A. & MARCOS-JORQUERA, D. 2015. A Construction Management Framework for Mass Customisation in Traditional Construction. *Sustainability,* 7**,** 5182-5210.

ANTAL-VAIDA, C. 2021. Basic Hyperparameters Tuning Methods for Classification Algorithms. *Informatica Economica,* 25**,** 64-74.

APICELLA, A., DONNARUMMA, F., ISGRÒ, F. & PREVETE, R. 2021. A survey on modern trainable activation functions. *Neural Networks,* 138**,** 14-32.

ASPIN, A. 2016. *Pro Power BI Desktop*, Springer.

ASSAAD, R., EL-ADAWAY, I. H. & ABOTALEB, I. S. 2020. Predicting Project Performance in the Construction Industry. *Journal of Construction Engineering and Management,* 146**,** 04020030.

ASTERIS, P. G., SKENTOU, A. D., BARDHAN, A., SAMUI, P. & PILAKOUTAS, K. 2021. Predicting concrete compressive strength using hybrid ensembling of surrogate machine learning models. *Cement and Concrete Research,* 145**,** 106449.

AWAD, A. & FAYEK, A. R. 2012. A decision support system for contractor prequalification for surety bonding. *Automation in Construction,* 21**,** 89-98.

AZAPAGIC, A. & PERDAN, S. 2000. Indicators of Sustainable Development for Industry: A General Framework. *Process Safety and Environmental Protection,* 78**,** 243-261.

BACK, T. 1996. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*, Oxford University Press.

BAKER, B., GUPTA, O., RASKAR, R. & NAIK, N. 2017. Accelerating neural architecture search using performance prediction. *arXiv preprint arXiv:1705.10823*.

BAKER, H., HALLOWELL, M. R. & TIXIER, A. J. P. 2020. Automatically learning construction injury precursors from text. *Automation in Construction,* 118**,** 103145.

BALA, K., AHMAD BUSTANI, S. & SHEHU WAZIRI, B. 2014. A computer-based cost prediction model for institutional building projects in Nigeria. *Journal of Engineering, Design and Technology,* 12**,** 519-530.

BATAINEH, A. A. & KAUR, D. A Comparative Study of Different Curve Fitting Algorithms in Artificial Neural Network using Housing Dataset. NAECON 2018 - IEEE National Aerospace and Electronics Conference, 23-26 July 2018 2018. 174-178.

BECKER, L. T. & GOULD, E. M. 2019. Microsoft Power BI: Extending Excel to Manipulate, Analyze, and Visualize Diverse Data. *Serials Review,* 45**,** 184-188.

BĚLOHLÁVEK, R. & KLIR, G. J. 2011. *Concepts and Fuzzy Logic*, MIT Press.

BENGIO, Y., SIMARD, P. & FRASCONI, P. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks,* 5**,** 157-166.

BERGSTRA, J., BARDENET, R., KÉGL, B. & BENGIO, Y. 2011. *Algorithms for Hyper-Parameter Optimization*.

BERGSTRA, J. & BENGIO, Y. 2012a. Random Search for Hyper-Parameter Optimization. *The Journal of Machine Learning Research,* 13**,** 281-305.

BERGSTRA, J. & BENGIO, Y. 2012b. Random search for hyper-parameter optimization. *Journal of machine learning research,* 13.

BERGSTRA, J., YAMINS, D. & COX, D. 2013. Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures. *In:* SANJOY, D. & DAVID, M. (eds.) *Proceedings of the 30th*

International Conference on Machine Learning. Proceedings of Machine Learning Research: PMLR.

BILAL, M. & OYEDELE, L. O. 2020. Guidelines for applied machine learning in construction industry—A case of profit margins estimation. *Advanced Engineering Informatics,* 43**,** 101013.

BILAL, M., OYEDELE, L. O., QADIR, J., MUNIR, K., AJAYI, S. O., AKINADE, O. O., OWOLABI, H. A., ALAKA, H. A. & PASHA, M. 2016. Big Data in the construction industry: A review of present status, opportunities, and future trends. *Advanced Engineering Informatics,* 30**,** 500-521.

BISCHL, B., BINDER, M., LANG, M., PIELOK, T., RICHTER, J., COORS, S., THOMAS, J., ULLMANN, T., BECKER, M. & BOULESTEIX, A. L. 2021. Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery***,** e1484.

BISCHL, B., BINDER, M., LANG, M., PIELOK, T., RICHTER, J., COORS, S., THOMAS, J., ULLMANN, T., BECKER, M., BOULESTEIX, A. L., DENG, D. & LINDAUER, M. 2023. Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery,* 13.

BISONG, E. & BISONG, E. 2019. Introduction to Scikit-learn. *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners***,** 215-229.

BÖCKER, L. & MEELEN, T. 2017. Sharing for people, planet or profit? Analysing motivations for intended sharing economy participation. *Environmental Innovation and Societal Transitions,* 23**,** 28-39.

BOLÓN-CANEDO, V., SÁNCHEZ-MAROÑO, N. & ALONSO-BETANZOS, A. 2013. A review of feature selection methods on synthetic data. *Knowledge and information systems,* 34**,** 483-519.

BREIMAN, L. 1996. Bagging predictors. *Machine Learning,* 24**,** 123-140.

BROOKS, T. F., POPE, D. S. & MARCOLINI, M. A. Airfoil self-noise and prediction. 1989.

BUGATA, P. & DROTÁR, P. 2023. Feature Selection Based on a Sparse Neural-Network Layer With Normalizing Constraints. *IEEE Transactions on Cybernetics,* 53**,** 161-172.

BUHAMDAN, S., ALWISY, A. & BOUFERGUENE, A. 2020. Explore the application of reinforced learning to support decision making during the design phase in the construction industry. *Procedia Manufacturing,* 42**,** 181-187.

CALVO-PARDO, H., MANCINI, T. & OLMO, J. 2023. Optimal deep neural networks by maximization of the approximation power. *Computers & Operations Research,* 156**,** 106264.

CAR-PUSIC, D., PETRUSEVA, S., PANCOVSKA, V. Z. & ZAFIROVSKI, Z. 2020. Neural Network-Based Model for Predicting Preliminary Construction Cost as Part of Cost Predicting System. *Advances in Civil Engineering,* 2020.

CARTY GERARD, J. 1995. Construction. *Journal of Construction Engineering and Management,* 121**,** 319-328.

CASSOTTI, M., BALLABIO, D., TODESCHINI, R. & CONSONNI, V. 2015. A similarity-based QSAR model for predicting acute toxicity towards the fathead minnow (Pimephales promelas). *SAR and QSAR in Environmental Research,* 26**,** 217-243.

CHAOVALITWONGSE, W. A., WANG, W., WILLIAMS, T. P. & CHAOVALITWONGSE, P. 2012. Data Mining Framework to Optimize the Bid Selection Policy for Competitively Bid Highway Construction Projects. *Journal of Construction Engineering and Management,* 138**,** 277-286.

CHEN, J. M. 2021. An Introduction to Machine Learning for Panel Data. *International Advances in Economic Research,* 27**,** 1-16.

CHEN, L. & PAN, W. 2021. Review fuzzy multi-criteria decision-making in construction management using a network approach. *Applied Soft Computing,* 102**,** 107103.

CHEN, Q., MAO, P., ZHU, S., XU, X. & FENG, H. 2024. A decision-aid system for subway microenvironment health risk intervention based on backpropagation neural network and permutation feature importance method. *Building and Environment,* 253**,** 111292.

CHEN, Z., XIAO, F., GUO, F. & YAN, J. 2023. Interpretable machine learning for building energy management: A state-of-the-art review. *Advances in Applied Energy,* 9**,** 100123.

CHENG, M.-Y., TSAI, H.-C. & SUDJONO, E. 2010. Conceptual cost estimates using evolutionary fuzzy hybrid neural network for projects in construction industry. *Expert Systems with Applications,* 37**,** 4224-4231.

CHENG, M.-Y., TSAI, H.-C. & SUDJONO, E. 2012. Evolutionary fuzzy hybrid neural network for dynamic project success assessment in construction industry. *Automation in Construction,* 21**,** 46-51.

CHIANG, W. C., URBAN, T. L. & BALDRIDGE, G. W. 1996. A neural network approach to mutual fund net asset value forecasting. *Omega,* 24**,** 205-215.

CHOI, D., CHO, H. & RHEE, W. On the difficulty of DNN hyperparameter optimization using learning curve prediction. TENCON 2018-2018 IEEE Region 10 Conference, 2018. IEEE, 0651-0656.

CHOI, S.-W., LEE, E.-B. & KIM, J.-H. 2021. The Engineering Machine-Learning Automation Platform (EMAP): A Big-Data-Driven AI Tool for Contractors' Sustainable Management Solutions for Plant Projects. *Sustainability,* 13**,** 10384.

CHUNG, J., GULCEHRE, C., CHO, K. & BENGIO, Y. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

DE BOCK, K. W., COUSSEMENT, K., CAIGNY, A. D., SŁOWIŃSKI, R., BAESENS, B., BOUTE, R. N., CHOI, T.-M., DELEN, D., KRAUS, M., LESSMANN, S., MALDONADO, S., MARTENS, D., ÓSKARSDÓTTIR, M., VAIRETTI, C., VERBEKE, W. & WEBER, R. 2023. Explainable AI for Operational Research: A defining framework, methods, applications, and a research agenda. *European Journal of Operational Research*.

DEL BUONO, N., ESPOSITO, F. & SELICATO, L. Methods for Hyperparameters Optimization in Learning Approaches: An Overview. *In:* NICOSIA, G., OJHA, V., LA MALFA, E., JANSEN, G., SCIACCA, V., PARDALOS, P., GIUFFRIDA, G. & UMETON, R., eds. Machine Learning, Optimization, and Data Science, 2020// 2020 Cham. Springer International Publishing, 100-112.

DOLOI, H. 2013. Cost Overruns and Failure in Project Management: Understanding the Roles of Key Stakeholders in Construction Projects. *Journal of Construction Engineering and Management,* 139**,** 267-279.

DOMHAN, T., SPRINGENBERG, J. T. & HUTTER, F. Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves. Twenty-fourth international joint conference on artificial intelligence, 2015.

DONG, N., FU, Y., XIONG, F., LI, L., AO, Y. & MARTEK, I. 2019. Sustainable Construction Project Management (SCPM) Evaluation—A Case Study of the Guangzhou Metro Line-7, PR China. *Sustainability,* 11**,** 5731.

EBRAT, M. & GHODSI, R. 2014. Construction project risk assessment by using adaptive-network-based fuzzy inference system: An empirical study. *KSCE Journal of Civil Engineering,* 18**,** 1213-1227.

EFFROSYNIDIS, D. & ARAMPATZIS, A. 2021. An evaluation of feature selection methods for environmental data. *Ecological Informatics,* 61**,** 101224.

ELMOUSALAMI, H. H. 2019. Intelligent methodology for project conceptual cost prediction. *Heliyon,* 5**,** e01625.

ETHEM, A. 2021. 1 WHY WE ARE INTERESTED IN MACHINE LEARNING. *Machine Learning.* MIT Press.

FALKNER, S., KLEIN, A. & HUTTER, F. 2018. *BOHB: Robust and Efficient Hyperparameter Optimization at Scale*.

FALLAHPOUR, A., KUAN YEW, W., RAJOO, S., OLUGU, E. U., NILASHI, M. & TURSKIS, Z. 2020. A fuzzy decision support system for sustainable construction project selection: an integrated FPP-FIS model. *Journal of Civil Engineering and Management,* 26**,** 247-258.

FALLAHPOUR, A., UDONCY OLUGU, E., NURMAYA MUSA, S., YEW WONG, K. & NOORI, S. 2017. A decision support model for sustainable supplier selection in sustainable supply chain management. *Computers & Industrial Engineering,* 105**,** 391-410.

FATOUREHCHI, D. & ZARGHAMI, E. 2020. Social sustainability assessment framework for managing sustainable construction in residential buildings. *Journal of Building Engineering,* 32**,** 101761.

FERREIRO-CABELLO, J., FRAILE-GARCIA, E., MARTINEZ DE PISON ASCACIBAR, E. & MARTINEZ-DE-PISON, F. J. 2018. Metamodel-based design optimization of structural one-way slabs based on deep learning neural networks to reduce environmental impact. *Engineering Structures,* 155**,** 91-101.

FRIEDMAN, J. H. 1991. Multivariate adaptive regression splines. *The annals of statistics,* 19**,** 1-67.

G, D. E. P., H, C. J. B. & NAVAUX, P. O. A. Hyperparameter Optimization for Convolutional Neural Networks with Genetic Algorithms and Bayesian Optimization. 2022 IEEE Latin American Conference on Computational Intelligence (LA-CCI), 23-25 Nov. 2022 2022. 1-5.

GALJANIĆ, K., MAROVIĆ, I. & JAJAC, N. 2022. Decision Support Systems for Managing Construction Projects: A Scientific Evolution Analysis. *Sustainability,* 14**,** 4977.

GEIFMAN, Y. & EL-YANIV, R. SelectiveNet: A Deep Neural Network with an Integrated Reject Option. International Conference on Machine Learning, 2019.

GHAZIMORADI, M., KHEYRODDIN, A. & REZAIFAR, O. 2016. Diagnosing the success of the construction projects during the initial phases. *Decision Science Letters,* 5**,** 395-406.

GONÇALVES, C. T., GONÇALVES, M. J. A. & CAMPANTE, M. I. 2023. Developing Integrated Performance Dashboards Visualisations Using Power BI as a Platform. *Information,* 14**,** 614.

GONZÁLEZ, R. L. 2008. PhD Thesis Neural Networks for Variational Problems in Engineering Roberto López González.

GRANITTO, P. M., VERDES, P. F. & CECCATTO, H. A. 2005. Neural network ensembles: evaluation of aggregation algorithms. *Artificial Intelligence,* 163**,** 139-162.

GRAVES, A. 2012. *Supervised sequence labelling with recurrent neural networks / [internet resource],* Berlin

New York, Berlin

New York : Springer.

GREENWELL, B. M. 2022. *Tree-based methods for statistical learning in R*, Chapman and Hall/CRC.

GREENWELL, B. M., BOEHMKE, B. C. & GRAY, B. 2020. Variable Importance Plots-An Introduction to the vip Package. *R J.,* 12**,** 343.

GREIF, T., STEIN, N. & FLATH, C. M. 2020. Peeking into the void: Digital twins for construction site logistics. *Computers in Industry,* 121**,** 103264.

GU, J., WANG, Z., KUEN, J., MA, L., SHAHROUDY, A., SHUAI, B., LIU, T., WANG, X., WANG, G., CAI, J. & CHEN, T. 2018. Recent advances in convolutional neural networks. *Pattern Recognition,* 77**,** 354-377.

GUERLAIN, C., RENAULT, S., FERRERO, F. & FAYE, S. 2019. Decision Support Systems for Smarter and Sustainable Logistics of Construction Sites. *Sustainability,* 11.

HAGIWARA, M. 2021. *Real-World Natural Language Processing: Practical Applications with Deep Learning*, Manning.

HAPKE, H., HOWARD, C. & LANE, H. 2019. *Natural Language Processing in Action: Understanding, analyzing, and generating text with Python*, Simon and Schuster.

HATEFI, S. M. & TAMOŠAITIENĖ, J. 2019. An integrated fuzzy DEMATEL-fuzzy ANP model for evaluating construction projects by considering interrelationships among risk factors. *Journal of Civil Engineering and Management,* 25**,** 114-131.

HOCHREITER, S. & SCHMIDHUBER, J. 1997. Long Short-term Memory. *Neural computation,* 9**,** 1735-80.

HONG, J., KANG, H., AN, J., CHOI, J., HONG, T., PARK, H. S. & LEE, D.-E. 2021. Towards environmental sustainability in the local community: Future insights for managing the hazardous pollutants at construction sites. *Journal of Hazardous Materials,* 403**,** 123804.

HUSSAIN, D., HUSSAIN, T., KHAN, A. A., NAQVI, S. A. A. & JAMIL, A. 2020. A deep learning approach for hydrological time-series prediction: A case study of Gilgit river basin. *Earth Science Informatics,* 13**,** 915-927.

HUTTER, F., KOTTHOFF, L. & VANSCHOREN, J. 2019. *Automated Machine Learning : Methods, Systems, Challenges,* Cham, SWITZERLAND, Springer International Publishing AG.

IRIE, K., TÜSKE, Z., ALKHOULI, T., SCHLÜTER, R. & NEY, H. LSTM, GRU, highway and a bit of attention: An empirical overview for language modeling in speech recognition. Interspeech, 2016. 3519-3523.

JAÉN-VARGAS, M., REYES LEIVA, K. M., FERNANDES, F., BARROSO GONÇALVES, S., TAVARES SILVA, M., LOPES, D. S. & SERRANO OLMEDO, J. J. 2022. Effects of sliding window variation in the performance of acceleration-based human activity recognition using deep learning models. *PeerJ Comput Sci,* 8**,** e1052.

JAMIESON, K. & TALWALKAR, A. Non-stochastic Best Arm Identification and Hyperparameter Optimization.

JEONG, J. S. & RAMÍREZ-GÓMEZ, Á. 2018. Development of a web graphic model with Fuzzy-DEcision-MAking Trial and Evaluation Laboratory/Multi-Criteria-Spatial Decision Support System (F-DEMATEL/MC-SDSS) for sustainable planning and construction of rural housings. *Journal of Cleaner Production,* 199.

JIANG, Y., LI, H. & ZHOU, Y. 2022. Compressive Strength Prediction of Fly Ash Concrete Using Machine Learning Techniques. *Buildings,* 12**,** 690.

KAMALOV, F. 2021. Orthogonal variance decomposition based feature selection. *Expert Systems with Applications,* 182**,** 115191.

KANNAN, D., MINA, H., NOSRATI-ABARGHOOEE, S. & KHOSROJERDI, G. 2020. Sustainable circular supplier selection: A novel hybrid approach. *Science of The Total Environment,* 722**,** 137936.

KANWAL, A., LAU, M. F., NG, S. P. H., SIM, K. Y. & CHANDRASEKARAN, S. 2022. BiCuDNNLSTM-1dCNN — A hybrid deep learning-based predictive model for stock price prediction. *Expert Systems with Applications,* 202**,** 117123.

KAZMI, W., NABNEY, I., VOGIATZIS, G., ROSE, P. & CODD, A. 2021. An Efficient Industrial System for Vehicle Tyre (Tire) Detection and Text Recognition Using Deep Learning. *IEEE Transactions on Intelligent Transportation Systems,* 22**,** 1264-1275.

KEEN, P. G. W. 1980. Adaptive design for decision support systems. *SIGMIS Database,* 12**,** 15–25.

KERSTEN, G. E. 2000. Decision Making and Decision Support. *In:* KERSTEN, G. E., MIKOLAJUK, Z. & YEH, A. G.-O. (eds.) *Decision Support Systems for Sustainable*

Development: A Resource Book of Methods and Applications. Boston, MA: Springer US.
KHAN, S. A., KUSI-SARPONG, S., ARHIN, F. K. & KUSI-SARPONG, H. 2018. Supplier sustainability performance evaluation and selection: A framework and methodology. *Journal of Cleaner Production,* 205**,** 964-979.
KHANNA, V. V., CHADAGA, K., SAMPATHILA, N., CHADAGA, R., PRABHU, S., K S, S., JAGDALE, A. S. & BHAT, D. 2023. A decision support system for osteoporosis risk prediction using machine learning and explainable artificial intelligence. *Heliyon,* 9**,** e22456.
KIANI MAVI, R. & STANDING, C. 2018. Critical success factors of sustainable project management in construction: A fuzzy DEMATEL-ANP approach. *Journal of Cleaner Production,* 194**,** 751-765.
KIM, J., NGUYEN, A. D. & LEE, S. 2019. Deep CNN-Based Blind Image Quality Predictor. *IEEE Transactions on Neural Networks and Learning Systems,* 30**,** 11-24.
KIM, S. 2013. Hybrid forecasting system based on case-based reasoning and analytic hierarchy process for cost estimation. *Journal of Civil Engineering and Management,* 19**,** 86-96.
KINGMA, D. P. & BA, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
KLEIN, A., FALKNER, S., SPRINGENBERG, J. T. & HUTTER, F. Learning curve prediction with Bayesian neural networks. International Conference on Learning Representations, 2017.
KOO, C., HONG, T. & HYUN, C. 2011. The development of a construction cost prediction model with improved prediction capacity using the advanced CBR approach. *Expert Systems with Applications,* 38**,** 8597-8606.
KRAMER, O. 2016. Machine Learning. *In:* KRAMER, O. (ed.) *Machine Learning for Evolution Strategies.* Cham: Springer International Publishing.
KRISHNAN, V. 2017. Research data analysis with power bi.
KUPTAMETEE, C., MICHALOPOULOU, Z.-H. & AUNSRI, N. 2024. A review of efficient applications of genetic algorithms to improve particle filtering optimization problems. *Measurement,* 224**,** 113952.
KUZLU, M., CALI, U., SHARMA, V. & Ö, G. 2020. Gaining Insight Into Solar Photovoltaic Power Generation Forecasting Utilizing Explainable Artificial Intelligence Tools. *IEEE Access,* 8**,** 187814-187823.
KWAK, N. W. & LIM, D. H. 2021. Financial time series forecasting using AdaBoost-GRU ensemble model. *The Korean Data & Information Science Society,* 32**,** 267-281.
LAU, K., LÓPEZ, R. & OÑATE IBÁÑEZ DE NAVARRA, E. 2009. A neural networks approach to aerofoil noise prediction.
LEE, M. & YU-LAN, H. 2020. Corporate Social Responsibility and Corporate Performance: A Hybrid Text Mining Algorithm. *Sustainability,* 12**,** 3075.

LI, F., LIU, M., ZHAO, Y., KONG, L., DONG, L., LIU, X. & HUI, M. 2019. Feature extraction and classification of heart sound using 1D convolutional neural networks. *EURASIP Journal on Advances in Signal Processing,* 2019**,** 59.

LI, L., JAMIESON, K., DESALVO, G., ROSTAMIZADEH, A. & TALWALKAR, A. 2017. Hyperband: A novel bandit-based approach to hyperparameter optimization. *The Journal of Machine Learning Research,* 18**,** 6765-6816.

LI, Z., LIU, F., YANG, W., PENG, S. & ZHOU, J. 2022. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. *IEEE Transactions on Neural Networks and Learning Systems,* 33**,** 6999-7019.

LIN, C.-H. & NUHA, U. 2022. RobustSTL and Machine-Learning Hybrid to Improve Time Series Prediction of Base Station Traffic. *Electronics,* 11**,** 1223.

LINDAUER, M., EGGENSPERGER, K., FEURER, M., BIEDENKAPP, A., DENG, D., BENJAMINS, C., RUHKOPF, T., SASS, R. & HUTTER, F. 2022. SMAC3: A Versatile Bayesian Optimization Package for Hyperparameter Optimization. *J. Mach. Learn. Res.,* 23**,** 1-9.

LIU, Q. & LIU, F. 2020. Selective Cascade of Residual ExtraTrees. *SN Computer Science,* 1**,** 354.

LIU, S., ZHANG, H. & JIN, Y. 2022. A survey on computationally efficient neural architecture search. *Journal of Automation and Intelligence,* 1**,** 100002.

LIU, Y., SUN, Y., XUE, B., ZHANG, M., YEN, G. G. & TAN, K. C. 2023. A Survey on Evolutionary Neural Architecture Search. *IEEE Transactions on Neural Networks and Learning Systems,* 34**,** 550-570.

LOVE, P. E., FANG, W., MATTHEWS, J., PORTER, S., LUO, H. & DING, L. 2023. Explainable artificial intelligence (XAI): Precepts, models, and opportunities for research in construction. *Advanced Engineering Informatics,* 57**,** 102024.

LUNDBERG, S. M. & LEE, S.-I. 2017. A unified approach to interpreting model predictions. *Advances in neural information processing systems,* 30.

LUTHRA, S., GOVINDAN, K., KANNAN, D., MANGLA, S. K. & GARG, C. P. 2017. An integrated framework for sustainable supplier selection and evaluation in supply chains. *Journal of Cleaner Production,* 140**,** 1686-1698.

MACHLEV, R., HEISTRENE, L., PERL, M., LEVY, K. Y., BELIKOV, J., MANNOR, S. & LEVRON, Y. 2022. Explainable Artificial Intelligence (XAI) techniques for energy and power systems: Review, challenges and opportunities. *Energy and AI,* 9**,** 100169.

MANDLER, H. & WEIGAND, B. 2023. Feature importance in neural networks as a means of interpretation for data-driven turbulence models. *Computers & Fluids,* 265**,** 105993.

MARCÍLIO, W. E. & ELER, D. M. From explanations to feature selection: assessing SHAP values as feature selection mechanism. 2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), 7-10 Nov. 2020 2020. 340-347.

MARSHALL, P., HIRMAS, A. & SINGER, M. 2018. Heinrich's pyramid and occupational safety: A statistical validation methodology. *Safety Science,* 101**,** 180-189.

MARTÍNEZ-ROJAS, M., MARÍN, N. & VILA, M. A. 2016. The Role of Information Technologies to Address Data Handling in Construction Project Management. *Journal of Computing in Civil Engineering,* 30**,** 04015064.

MARZOUGHI, F. & ARTHANARI, T. 2016. A Conceptual Framework for a Navigational Support System for Construction Projects. *Procedia Computer Science,* 100**,** 449-457.

MARZOUK, M. M. & AHMED, R. M. 2011. A case-based reasoning approach for estimating the costs of pump station projects. *Journal of Advanced Research,* 2**,** 289-295.

MINHAS, M. R., POTDAR, V. & SIANAKI, O. A. A Decision Support System for Selecting Sustainable Materials in Construction Projects. 2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA), 16-18 May 2018 2018. 721-726.

MOHAMMAD KABIR YAQUBI, S. S. 2019. The Automated Cost Estimation in Construction. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)***,** 5.

MORALES-HERNÁNDEZ, A., VAN NIEUWENHUYSE, I. & ROJAS GONZALEZ, S. 2023. A survey on multi-objective hyperparameter optimization algorithms for machine learning. *Artificial Intelligence Review,* 56**,** 8043-8093.

MORONEY, L. 2020. *AI and Machine Learning for coders*, O'Reilly Media.

MOTZ, M., KRAUß, J. & SCHMITT, R. H. 2022. Benchmarking of hyperparameter optimization techniques for machine learning applications in production. *Advances in Industrial and Manufacturing Engineering,* 5**,** 100099.

MULIAUWAN, H. N., PRAYOGO, D., GABY, G. & HARSONO, K. 2020. Prediction of Concrete Compressive Strength Using Artificial Intelligence Methods. *Journal of Physics: Conference Series,* 1625**,** 012018.

MUMALI, F. 2022. Artificial neural network-based decision support systems in manufacturing processes: A systematic literature review. *Computers & Industrial Engineering,* 165**,** 107964.

MURAT GUNDUZ, H. M. A. L. 2021. Go/No-Go Decision Model for Owners Using Exhaustive CHAID and QUEST Decision Tree Algorithms. *Sustainability,* 13**,** 815.

MUSOLF, A. M., HOLZINGER, E. R., MALLEY, J. D. & BAILEY-WILSON, J. E. 2022. What makes a good prediction? Feature importance and beginning to open the black box of machine learning in genetics. *Human Genetics***,** 1-14.

NAGY, B., GALATA, D. L., FARKAS, A. & NAGY, Z. K. 2022. Application of artificial neural networks in the process analytical technology of pharmaceutical manufacturing—a review. *The AAPS Journal,* 24**,** 74.

NATH, P. C., MISHRA, A. K., SHARMA, R., BHUNIA, B., MISHRA, B., TIWARI, A., NAYAK, P. K., SHARMA, M., BHUYAN, T., KAUSHAL, S., MOHANTA, Y. K. & SRIDHAR, K. 2024. Recent advances in artificial intelligence towards the sustainable future of agri-food industry. *Food Chemistry,* 447**,** 138945.

NGUYEN-SY, T., WAKIM, J., TO, Q.-D., VU, M.-N., NGUYEN, T.-D. & NGUYEN, T.-T. 2020. Predicting the compressive strength of concrete from its compositions and age using the extreme gradient boosting method. *Construction and Building Materials,* 260**,** 119757.

NIKMEHR, B., HOSSEINI, M. R., MARTEK, I., ZAVADSKAS, E. K. & ANTUCHEVICIENE, J. 2021. Digitalization as a Strategic Means of Achieving Sustainable Efficiencies in Construction Management: A Critical Review. *Sustainability,* 13**,** 5040.

O'MALLEY, T. A. B., ELIE AND LONG, JAMES AND CHOLLET, FRAN\C{C}OIS AND JIN, HAIFENG AND INVERNIZZI, LUCA AND OTHERS 2019. Keras Tuner. https://github.com/keras-team/keras-tuner.

OFORI, G. 1990. *The Construction Industry: Aspects of Its Economics and Management*, Singapore University Press.

OH, S. 2019. Feature interaction in terms of prediction performance. *Applied Sciences,* 9**,** 5191.

OMAR, M. F., MOHD NAWI, M. N., CHE-ANI, A. I., SANIAH SULAIMAN, N. I. & GOH, K. C. 2016. Innovative Approach for IBS Vendor Selection Problem. *MATEC Web of Conferences,* 47**,** 04018.

ORTIZ-GONZALEZ, J. I., DURAN-HERAS, A. & CASTILLA-ALCALA, G. Why Do Traditional Project Management Methods Hinder the Competitiveness of the Construction Industry? *In:* AVILÉS-PALACIOS, C. & GUTIERREZ, M., eds. Ensuring Sustainability, 2022// 2022 Cham. Springer International Publishing, 225-232.

PAN, Y. & ZHANG, L. 2021. Roles of artificial intelligence in construction engineering and management: A critical review and future trends. *Automation in Construction,* 122**,** 103517.

PAPADAKI, I. N. & CHASSIAKOS, A. P. 2016. Multi-objective Construction Site Layout Planning Using Genetic Algorithms. *Procedia Engineering,* 164**,** 20-27.

PARR, T., HAMRICK, J. & WILSON, J. D. 2024. Nonparametric feature impact and importance. *Information Sciences,* 653**,** 119563.

PATEL, R. A., PATEL, M. B., PATEL, K. A. & PATEL, D. A. APPLICATION OF NEURAL NETWORK IN PREDICTING THE OUTCOME OF CONSTRUCTION DISPUTES. Proceedings of International Structural Engineering and Construction, 2023. LDR-01.

PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R. & DUBOURG, V. 2011. Scikit-learn: Machine learning in Python. *the Journal of machine Learning research,* 12**,** 2825-2830.

PREETI, BALA, R. & SINGH, R. P. 2022. A dual-stage advanced deep learning algorithm for long-term and long-sequence prediction for multivariate financial time series. *Applied Soft Computing,* 126**,** 109317.

QUINLAN, J. R. Combining instance-based and model-based learning. Proceedings of the tenth international conference on machine learning, 1993. 236-243.

RANJBARI, M., SHAMS ESFANDABADI, Z., ZANETTI, M. C., SCAGNELLI, S. D., SIEBERS, P.-O., AGHBASHLO, M., PENG, W., QUATRARO, F. & TABATABAEI, M. 2021. Three pillars of sustainability in the wake of COVID-19: A systematic review and future research agenda for sustainable development. *Journal of Cleaner Production,* 297**,** 126660.

RAO, H. R., SRIDHAR, R. & NARAIN, S. 1994. An active intelligent decision support system — Architecture and simulation. *Decision Support Systems,* 12**,** 79-91.

RENGASAMY, D., ROTHWELL, B. C. & FIGUEREDO, G. P. 2021. Towards a More Reliable Interpretation of Machine Learning Outputs for Safety-Critical Systems Using Feature Importance Fusion. *Applied Sciences,* 11**,** 11854.

RIBEIRO, M. T., SINGH, S. & GUESTRIN, C. " Why should i trust you?" Explaining the predictions of any classifier. Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016a. 1135-1144.

RIBEIRO, M. T., SINGH, S. & GUESTRIN, C. 2016b. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* San Francisco, California, USA: Association for Computing Machinery.

RONG, G., LI, K., SU, Y., TONG, Z., LIU, X., ZHANG, J., ZHANG, Y. & LI, T. 2021. Comparison of Tree-Structured Parzen Estimator Optimization in Three Typical Neural Network Models for Landslide Susceptibility Assessment. *Remote Sensing,* 13**,** 4694.

RUDNICKI, W. R., WRZESIEŃ, M. & PAJA, W. 2015. All Relevant Feature Selection Methods and Applications. *In:* STAŃCZYK, U. & JAIN, L. C. (eds.) *Feature Selection for Data and Pattern Recognition.* Berlin, Heidelberg: Springer Berlin Heidelberg.

SALMA AHMED, S. E.-S. 2021. Critical Review of the Evolution of Project Delivery Methods in the Construction Industry. *Buildings,* 11**,** 11.

SANTOS, J., BRESSI, S., CEREZO, V. & LO PRESTI, D. 2019. SUP&R DSS: A sustainability-based decision support system for road pavements. *Journal of Cleaner Production,* 206**,** 524-540.

SASSI, R. J., AFFONSO, C. & FERREIRA, R. P. Rough Neuro-Fuzzy Network Applied to Traffic Flow Breakdown in the City of Sao Paulo. 2011 International Conference on Management and Service Science, 2011. IEEE, 1-5.

SETTOUTI, N. & SAIDI, M. 2024. Preliminary analysis of explainable machine learning methods for multiple myeloma chemotherapy treatment recognition. *Evolutionary Intelligence,* 17**,** 513-533.

SETURIDZE, R. & TOPURIA, N. 2021. A way of developing collaboration between universities and businesses in a time of COVID-19. *Kybernetes,* 50**,** 1661-1678.

SHAHRIARI, B., SWERSKY, K., WANG, Z., ADAMS, R. P. & FREITAS, N. D. 2016. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE,* 104**,** 148-175.

SHAZIYA, H. & ZAHEER, R. Impact of Hyperparameters on Model Development in Deep Learning. *In:* CHAKI, N., PEJAS, J., DEVARAKONDA, N. & RAO KOVVUR, R. M., eds. Proceedings of International Conference on Computational Intelligence and Data Engineering, 2021// 2021 Singapore. Springer Singapore, 57-67.

SHI, H., YIN, H. & WEI, L. 2016. A dynamic novel approach for bid/no-bid decision-making. *SpringerPlus,* 5**,** 1589.

SHI, M. & SHEN, W. 2022. Automatic Modeling for Concrete Compressive Strength Prediction Using Auto-Sklearn. *Buildings,* 12**,** 1406.

SMITH, L. N. Cyclical learning rates for training neural networks. 2017 IEEE winter conference on applications of computer vision (WACV), 2017. IEEE, 464-472.

SNOEK, J., LAROCHELLE, H. & ADAMS, R. P. 2012. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems,* 25.

SNYDER, H. 2019. Literature review as a research methodology: An overview and guidelines. *Journal of Business Research,* 104**,** 333-339.

SON, H., KIM, C. & KIM, C. 2012. Hybrid principal component analysis and support vector machine model for predicting the cost performance of commercial building projects using pre-project planning variables. *Automation in Construction,* 27**,** 60-66.

SOTO, B. G. D. & ADEY, B. T. 2016. Preliminary Resource-based Estimates Combining Artificial Intelligence Approaches and Traditional Techniques. *Procedia Engineering,* 164**,** 261-268.

SUI, G. & YU, Y. 2020. Bayesian Contextual Bandits for Hyper Parameter Optimization. *IEEE Access,* 8**,** 42971-42979.

SUZUKI, K. 2011. *Artificial Neural Networks: Industrial and Control Engineering Applications*, IntechOpen.

TANG, L. C. M., LEUNG, A. Y. T. & WONG, C. W. Y. 2010. Entropic Risk Analysis by a High Level Decision Support System for Construction SMEs. *Journal of Computing in Civil Engineering,* 24**,** 81-94.

TAPEH, A. T. G. & NASER, M. Z. 2023. Artificial Intelligence, Machine Learning, and Deep Learning in Structural Engineering: A Scientometrics Review of Trends and Best Practices. *Archives of Computational Methods in Engineering,* 30**,** 115-159.

TAYEFEH HASHEMI, S., EBADATI, O. M. & KAUR, H. 2020. Cost estimation and prediction in construction projects: a systematic review on machine learning techniques. *SN Applied Sciences,* 2**,** 1703.

TAYLAN, O., BAFAIL, A. O., ABDULAAL, R. M. S. & KABLI, M. R. 2014. Construction projects selection and risk assessment by fuzzy AHP and fuzzy TOPSIS methodologies. *Applied Soft Computing,* 17**,** 105-116.

TIPPING, M. E. 2001. Sparse Bayesian learning and the relevance vector machine. *Journal of machine learning research,* 1**,** 211-244.

TIXIER, A. J. P., HALLOWELL, M. R., RAJAGOPALAN, B. & BOWMAN, D. 2016a. Application of machine learning to construction injury prediction. *Automation in Construction,* 69**,** 102-114.

TIXIER, A. J. P., HALLOWELL, M. R., RAJAGOPALAN, B. & BOWMAN, D. 2016b. Automated content analysis for construction safety: A natural language processing system to extract precursors and outcomes from unstructured injury reports. *Automation in Construction,* 62**,** 45-56.

ULUBEYLI, S. & KAZAZ, A. 2015. Fuzzy multi-criteria decision making model for subcontractor selection in international construction projects. *Technological and Economic Development of Economy,* 22**,** 1-25.

UZAIR, M. & JAMIL, N. Effects of Hidden Layers on the Efficiency of Neural networks. 2020 IEEE 23rd International Multitopic Conference (INMIC), 5-7 Nov. 2020 2020. 1-6.

VARGAS, R. V. 2001. A new approach to PMBOK guide 2000. *Project Management Institute Annual Seminars & Symposium.* Nashville, TN. Newtown Square, PA: Project Management Institute.

VECOVEN, N., BEGON, J.-M., SUTERA, A., GEURTS, P. & HUYNH-THU, V. A. Nets versus trees for feature ranking and gene network inference. Discovery Science: 23rd International Conference, DS 2020, Thessaloniki, Greece, October 19–21, 2020, Proceedings 23, 2020. Springer, 231-245.

VISHNU, A. A., SURESH, A., KOSHY, R. A., SANJNA, S. & DAVIS, P. R. Hyper-parameter Optimised Artificial Neural Network Model for Failure Mode Identification of RC Shear Wall. International Conference on Structural Engineering and Construction Management, 2022. Springer, 973-985.

WAN, Z., XU, Y. & ŠAVIJA, B. 2021. On the Use of Machine Learning Models for Prediction of Compressive Strength of Concrete: Influence of Dimensionality Reduction on the Model Performance. *Materials,* 14**,** 713.

WANG, K., MA, C., QIAO, Y., LU, X., HAO, W. & DONG, S. 2021. A hybrid deep learning model with 1DCNN-LSTM-Attention networks for short-term traffic flow prediction. *Physica A: Statistical Mechanics and its Applications,* 583**,** 126293.

WANG, S.-C. 2003. Artificial Neural Network. *In:* WANG, S.-C. (ed.) *Interdisciplinary Computing in Java Programming.* Boston, MA: Springer US.

WANG, Y.-R., YU, C.-Y. & CHAN, H.-H. 2012. Predicting construction cost and schedule success using artificial neural networks ensemble and support vector machines classification models. *International Journal of Project Management,* 30**,** 470-478.

WEI, J., CHU, X., SUN, X. Y., XU, K., DENG, H. X., CHEN, J., WEI, Z. & LEI, M. 2019. Machine learning in materials science. *InfoMat,* 1**,** 338-358.

WEI, T., WANG, S., ZHONG, J., LIU, D. & ZHANG, J. 2022. A Review on Evolutionary Multitask Optimization: Trends and Challenges. *IEEE Transactions on Evolutionary Computation,* 26**,** 941-960.

WEN, G. Construction project risk evaluation based on Rough Sets and Artificial Neural Networks. 2010 Sixth International Conference on Natural Computation, 10-12 Aug. 2010 2010. 1624-1628.

WILLIAMS, T. P. & GONG, J. 2014. Predicting construction cost overruns using text mining, numerical data and ensemble classifiers. *Automation in Construction,* 43**,** 23-29.

WISTUBA, M. & PEDAPATI, T. Learning to rank learning curves. International Conference on Machine Learning, 2020. PMLR, 10303-10312.

XIE, Y., SHEN, J. & WU, C. 2020. Affine Geometrical Region CNN for Object Tracking. *IEEE Access,* 8**,** 68638-68648.

YANG, L. & SHAMI, A. 2020. On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing,* 415**,** 295-316.

YASEEN, Z. M., ALI, Z. H., SALIH, S. Q. & AL-ANSARI, N. 2020. Prediction of Risk Delay in Construction Projects Using a Hybrid Artificial Intelligence Model. *Sustainability,* 12**,** 1514.

YEH, I.-C. 1998. Modeling of strength of high-performance concrete using artificial neural networks. *Cement and Concrete research,* 28**,** 1797-1808.

YEH, I.-C. 2006. Analysis of strength of concrete using design of experiments and neural networks. *Journal of Materials in Civil Engineering,* 18**,** 597-604.

YOU, Z. & WU, C. 2019. A framework for data-driven informatization of the construction company. *Advanced Engineering Informatics,* 39**,** 269-277.

YOUSEFI, V., HAJI YAKHCHALI, S., KHANZADI, M., MEHRABANFAR, E. & ŠAPARAUSKAS, J. 2016. Proposing a neural network model to predict time and cost claims in construction projects. *Journal of Civil Engineering and Management,* 22**,** 967-978.

YU, W.-D. & SKIBNIEWSKI, M. J. 2010. Integrating Neurofuzzy System with Conceptual Cost Estimation to Discover Cost-Related Knowledge from Residential Construction Projects. *Journal of Computing in Civil Engineering,* 24**,** 35-44.

ZACCONE, G. & KARIM, M. R. 2018. *Deep Learning with TensorFlow: Explore neural networks and build intelligent systems with Python*, Packt Publishing Ltd.

ZENG, Z., ZHANG, H., ZHANG, R. & YIN, C. 2015. A novel feature selection method considering feature interaction. *Pattern Recognition,* 48**,** 2656-2666.

ZHANG, Q., OO, B. L. & LIM, B. T. H. 2019. Drivers, motivations, and barriers to the implementation of corporate social responsibility practices by construction enterprises： A review. *Journal of Cleaner Production,* 210**,** 563-584.

ZHAO, W., JIAO, L., MA, W., ZHAO, J., ZHAO, J., LIU, H., CAO, X. & YANG, S. 2017. Superpixel-Based Multiple Local CNN for Panchromatic and Multispectral Image Classification. *IEEE Transactions on Geoscience and Remote Sensing,* 55**,** 4141-4156.

ZIMA, K. 2015. The Case-based Reasoning Model of Cost Estimation at the Preliminary Stage of a Construction Project. *Procedia Engineering,* 122**,** 57-64.

ZÖLLER, M.-A. & HUBER, M. F. 2021. Benchmark and survey of automated machine learning frameworks. *Journal of artificial intelligence research,* 70**,** 409-472.

Appendices

Appendix 5-1 DSS Input Data

Project	Project_ID	CP	PDA	CSPDA	On-site Start Month	Design Duration%	On-site Duration%	Project Duration(weeks)	Proposal Team	Design Team	Project Management	Site Management Staff	Commissioning Team	H&S	Overheads	Average of Cost Intensity	Final Net Construction Band	Sum of Total Construction Score	MCC/MCP	Elec Installation	Mechanical Install	Civil Installation	ICA	TTU	Chem Dosing	New Building	New Water Retain Structure	Power Supply	Tank	Security	RGF Refurbishment/New	Temporary Works	Pump/Booster Set	Scraper Bridges	Screen and Compactor/Grit Removal	SCADA	County
1	402676	1.00	0.16	0.00	Feb	0.05	0.95	32	0.01	0.81	0.13	0.00	0.02	0.00	0.04	17512	1M-2.5M	1370903	5	6	7	60	1	0	0	0	0	0	0	0	0	0	2	7	2	0	Stirlingshire
2	403125	0.97	0.29	0.52	Jul	0.15	0.85	39	0.05	0.24	0.51	0.00	0.05	0.04	0.10	33511	2.5M-5M	2525658	1	2	0	23	3	40	0	0	2	0	6	0	0	0	4	0	0	0	Sutherland
3	403127	1.08	0.23	0.61	Feb	0.23	0.77	137	0.04	0.52	0.27	0.03	0.01	0.01	0.07	67579	Over 10M	1.3E+07	0	3	1	34	0	43	0	0	14	0	0	0	0	0	2	0	1	0	Highland
4	403753	1.09	0.39	0.59	Mar	0.12	0.88	112	0.04	0.27	0.41	0.00	0.10	0.03	0.14	1590	0-1M	155548	2	9	27	22	7	0	27	0	0	0	0	0	0	0	2	0	0	0	Stirlingshire
5	403754	1.12	0.35	0.53	Mar	0.13	0.87	104	0.04	0.25	0.40	0.00	0.16	0.02	0.12	2589	0-1M	270749	12	4	5	26	15	0	0	0	0	0	0	0	0	0	5	0	0	0	Stirlingshire
6	403773	0.94	0.01	0.02	Jun	0.01	0.99	19	0.05	0.13	0.44	0.00	0.17	0.04	0.11	2604	0-1M	134577	0	4	40	28	25	0	0	0	0	0	0	0	0	0	0	0	0	0	Stirlingshire
7	403774	1.31	0.07	0.07	Aug	0.09	0.91	48	0.04	0.35	0.41	0.00	0.10	0.03	0.08	2622	0-1M	123786	0	3	31	43	11	0	0	0	0	0	0	0	0	0	7	0	0	0	Stirlingshire
8	403895	1.20	3.39	2.61	Mar	0.08	0.92	172	0.03	0.32	0.38	0.00	0.12	0.02	0.13	1659	0-1M	293719	0	8	32	0	55	0	0	0	0	3	0	0	0	0	0	0	0	0	Argyl and Bute
9	403902	1.03	0.28	0.35	Jun	0.15	0.85	145	0.04	0.75	0.14	0.01	0.03	0.01	0.03	47585	5M-10M	5873339	7	10	9	18	0	0	0	0	0	0	3	0	0	3	3	12	3	0	Aberdeenshire
10	403917	1.00	0.65	0.73	Mar	0.38	0.62	175	0.04	0.33	0.42	0.04	0.06	0.02	0.09	33057	2.5M-5M	3603266	0	4	3	10	0	53	0	15	0	0	0	0	0	0	0	0	0	0	Sutherland
11	403939	1.02	-0.03	1.12	Jan	0.18	0.83	39	0.08	0.71	0.10	0.01	0.03	0.00	0.06	18792	1M-2.5M	1023161	6	7	5	17	0	0	29	0	20	0	0	0	0	2	0	3	0	0	Ayrshire
12	403961	1.07	2.39	1.33	Oct	0.30	0.70	137	0.04	0.43	0.29	0.11	0.03	0.01	0.07	4729	0-1M	656676	0	57	0	32	0	0	0	0	0	0	0	0	0	0	3	0	0	0	Renfrewshire
13	403962	1.47	2.07	1.50	Sep	0.34	0.66	136	0.04	0.45	0.33	0.03	0.05	0.02	0.08	1558	0-1M	171374	0	2	8	37	1	0	0	0	0	0	0	0	0	0	51	0	0	0	Dunbartonshire
14	403963	1.05	0.30	0.23	Mar	0.05	0.95	144	0.08	0.10	0.51	0.16	0.05	0.02	0.07	16883	2.5M-5M	2555070	0	4	5	90	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Lanarkshire
15	403967	1.00	0.01	0.00	Oct	0.21	0.79	137	0.03	0.71	0.17	0.00	0.02	0.00	0.07	65665	5M-10M	9663826	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	West Lothian
16	403968	1.22	1.22	1.51	Dec	0.30	0.70	157	0.05	0.55	0.30	0.01	0.03	0.00	0.06	9696	1M-2.5M	1127558	12	6	1	26	1	0	0	0	0	0	0	0	0	0	0	0	0	0	East Lothian
17	403973	1.32	0.81	0.53	Nov	0.35	0.65	156	0.03	0.54	0.20	0.03	0.07	0.01	0.07	16307	1M-2.5M	1656374	6	0	0	35	25	0	0	0	0	0	3	0	0	0	0	0	0	0	Peeblesshire
18	403986	1.00	0.65	0.35	Sep	0.05	0.95	139	0.02	0.50	0.28	0.05	0.08	0.02	0.05	13979	2.5M-5M	2636043	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	77	Stirlingshire
19	403988	1.00	0.42	0.33	Jul	0.34	0.66	134	0.04	0.33	0.40	0.05	0.05	0.03	0.05	7562	0-1M	655529	0	22	7	8	15	0	39	0	0	0	0	0	0	0	0	0	0	0	Isle of Lewis
20	403991	1.02	0.50	0.73	Mar	0.17	0.83	137	0.09	0.53	0.19	0.04	0.04	0.01	0.05	26429	2.5M-5M	4104053	31	0	28	6	1	0	29	0	0	0	0	1	0	0	0	0	0	0	Highland
21	403992	0.82	1.64	1.51	Jul	0.28	0.72	156	0.03	0.40	0.35	0.05	0.07	0.04	0.05	10462	1M-2.5M	1177676	0	20	17	5	22	0	34	0	0	0	0	0	0	0	0	0	0	0	Isle of Lewis
22	403994	1.04	0.91	2.04	Aug	0.47	0.53	248	0.03	0.39	0.20	0.02	0.02	0.01	0.03	18053	1M-2.5M	2375208	11	12	1	16	10	0	44	0	0	0	0	0	0	0	0	0	0	0	Isle of Lewis
23	403996	0.92	0.44	0.33	Jan	0.44	0.56	114	0.06	0.43	0.19	0.13	0.07	0.02	0.06	14279	0-1M	315916	0	9	2	37	3	0	0	0	17	0	0	0	0	0	0	0	0	0	Highland
24	404007	1.01	0.43	0.31	Mar	0.14	0.86	178	0.12	0.43	0.23	0.14	0.01	0.02	0.05	21378	2.5M-5M	3270910	2	8	13	22	0	0	0	0	0	0	0	0	0	10	10	22	0	0	Renfrewshire
25	404069	1.16	0.61	0.34	Dec	0.53	0.47	223	0.06	0.51	0.30	0.03	0.03	0.02	0.05	15004	1M-2.5M	1566945	2	6	6	17	0	0	11	0	16	0	3	0	0	0	2	3	0	0	Highland
26	404093	1.00	0.00	0.00	Jan	0.02	0.98	148	0.00	0.72	0.19	0.00	0.04	0.01	0.04	11203	1M-2.5M	1629269	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	90	Renfrewshire
27	404134	1.00	0.24	0.53	Jul	0.26	0.74	125	0.07	0.32	0.39	0.02	0.04	0.02	0.14	15720	1M-2.5M	1642425	7	11	13	16	5	14	9	0	0	0	0	0	0	0	3	0	0	0	Highland
28	500012	1.00	-0.20	0.32	Mar	0.03	0.97	35	0.02	0.41	0.36	0.00	0.08	0.03	0.09	24265	0-1M	835416	0	1	18	79	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Highland
29	500015	1.01	0.53	0.00	Jan	0.27	0.73	79	0.06	0.21	0.56	0.00	0.01	0.03	0.13	7695	0-1M	456865	0	4	0	96	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Lanarkshire
30	500035	0.89	-0.27	-0.33	Mar	0.11	0.89	129	0.01	0.02	0.32	0.44	0.10	0.02	0.09	23810	2.5M-5M	2731368	0	0	0	0	0	0	0	0	0	0	0	0	##	0	0	0	0	0	Lanarkshire
31	500043	0.86	-0.16	0.23	Sep	0.63	0.37	39	0.07	0.43	0.31	0.05	0.02	0.03	0.09	16698	0-1M	605883	10	17	0	49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Benbecula
32	500047	1.16	1.01	0.00	Nov	0.63	0.37	245	0.05	0.57	0.25	0.01	0.07	0.00	0.04	22138	1M-2.5M	2027210	7	0	6	15	42	0	23	0	0	0	0	0	0	0	5	0	0	0	Roxburghshire
33	500051	1.17	-0.15	-0.13	Sep	0.61	0.39	91	0.03	0.42	0.29	0.07	0.05	0.01	0.12	11777	0-1M	415563	0	0	0	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Lanarkshire
34	500052	1.00	0.65	0.39	Jan	0.18	0.82	129	0.11	0.59	0.21	0.01	0.02	0.01	0.05	36481	2.5M-5M	3872200	3	9	8	59	0	0	0	0	0	0	1	0	0	1	1	3	2	0	Lanarkshire
35	500197	0.97	-0.37	-0.53	Feb	0.53	0.47	31	0.04	0.33	0.34	0.13	0.05	0.01	0.10	24511	0-1M	353655	0	10	0	8	57	0	0	0	0	0	0	0	0	0	0	0	0	0	Lanarkshire
36	500229	1.04	0.50	0.00	Oct	0.01	0.99	111	0.07	0.70	0.12	0.01	0.03	0.00	0.07	20671	1M-2.5M	2267908	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	East Lothian
37	500244	1.11	0.74	0.32	May	0.40	0.60	143	0.03	0.34	0.31	0.03	0.15	0.01	0.08	14445	1M-2.5M	1242246	0	0	8	10	0	0	40	0	0	0	0	0	0	0	0	0	0	39	Peeblesshire
38	500245	1.08	0.45	0.53	Feb	0.40	0.60	184	0.06	0.56	0.18	0.00	0.01	0.01	0.06	55627	5M-10M	5116984	2	7	10	80	1	0	0	0	0	0	0	0	0	0	0	0	0	0	Selkirkshire
39	500494	1.45	4.66	4.03	Jun	0.20	0.80	213	0.05	0.39	0.32	0.00	0.05	0.02	0.15	2674	0-1M	454508	0	0	51	26	21	0	0	0	0	0	0	0	0	0	0	0	0	0	Stirlingshire
40	500496	1.22	2.03	2.13	Oct	0.22	0.78	140	0.06	0.43	0.29	0.03	0.02	0.02	0.09	6349	0-1M	311230	4	13	16	18	2	0	0	0	0	0	0	0	0	1	11	0	32	0	Dumfries and Galloway
41	500497	1.21	1.45	0.33	Jan	0.48	0.52	135	0.03	0.47	0.34	0.04	0.05	0.01	0.05	10049	0-1M	967610	0	3	7	12	0	0	0	0	0	0	9	0	0	0	0	35	8	0	Fife
42	500499	1.30	3.80	1.01	Oct	0.47	0.53	274	0.04	0.41	0.40	0.01	0.06	0.01	0.09	4438	0-1M	647938	0	5	15	7	3	0	20	0	0	0	0	0	0	4	0	0	0	0	Angus

Appendix 5-1 DSS Input Data Continued…

Project	Project ID	CP	FDA	QS/FDA	On-site Start Month	Design Duration%	On-site Duration%	Project Duration (Weeks)	Proposal Team	Design Team	Project Management	Site Management Staff	Commissioning Team	H&S	Overheads	Average of Cost Intensity	First Net Construction Band	Sum of Total Construction Score	MCC/SMCP	Elec Installation	Mechanical Install	Civil Installation	ICA	TTU	Chem Dosing	New Building	New Water Retain Structure	Power Supply	Tank	Security	RGF Refurbishment/New	Temporary Works	Pump/Booster Set	Sludge Bridges	Screen and Compactor/Grit Removal	SCADA	County
43	500573	1.00	0.61	0.35	Jun	0.35	0.65	161	0.15	0.52	0.24	0.00	0.03	0.02	0.03	42534	2.5M-5M	4429586	2	4	8	23	1	0	0	0	45	0	0	0	0	16	0	0	0	0	Argyll and Bute
44	500575	1.00	0.00	0.00	Jul	0.51	0.49	71	0.17	0.56	0.16	0.00	0.09	0.00	0.01	18427	0-1M	639665	0	0	57	34	7	0	0	0	0	0	0	0	0	0	0	0	0	0	Renfrewshire
45	500579	1.08	1.23	1.42	Dec	0.15	0.85	202	0.05	0.39	0.34	0.14	0.03	0.01	0.05	19885	2.5M-5M	3406017	3	5	8	2	1	0	0	0	0	0	0	0	0	1	0	0	46	0	Stirlingshire
46	500580	1.07	0.15	0.03	Jul	0.72	0.28	230	0.09	0.59	0.22	0.02	0.02	0.01	0.06	47094	2.5M-5M	3067834	4	7	23	40	0	0	0	0	0	0	0	0	0	0	3	0	11	0	Moray
47	500660	1.14	0.52	0.33	Jul	0.08	0.92	81	0.03	0.39	0.34	0.00	0.04	0.03	0.17	9156	0-1M	676406	0	13	14	12	2	0	50	0	0	0	0	0	0	0	0	0	0	13	Renfrewshire
48	500694	1.13	1.25	1.25	Dec	0.18	0.82	185	0.06	0.49	0.34	0.04	0.03	0.00	0.04	6765	1M-2.5M	1034113	3	9	21	3	0	0	0	0	0	0	0	0	0	0	22	0	27	0	West Lothian
49	500695	1.03	2.48	2.50	Mar	0.27	0.73	143	0.09	0.39	0.39	0.04	0.04	0.01	0.04	7644	0-1M	802629	0	7	50	2	20	0	0	0	0	0	0	0	0	0	1	0	0	0	Fife
50	500709	0.98	-0.40	0.00	Nov	0.61	0.39	121	0.08	0.57	0.26	0.00	0.03	0.00	0.11	52875	1M-2.5M	2485128	0	1	0	54	0	0	45	0	0	0	0	0	0	0	0	0	0	0	Ayrshire
51	500731	1.00	0.00	0.00	Jun	0.37	0.63	146	0.04	0.70	0.21	0.00	0.03	0.00	0.02	38038	2.5M-5M	3477771	7	8	16	31	0	0	1	0	8	0	0	0	0	2	3	3	3	0	Aberdeenshire
52	500735	1.17	0.40	0.38	Mar	0.24	0.76	138	0.05	0.20	0.41	0.14	0.06	0.05	0.09	16478	1M-2.5M	1734878	0	6	4	43	3	0	0	0	43	0	0	1	0	0	0	0	0	0	Highland
53	500742	1.26	1.73	1.50	Jan	0.40	0.60	164	0.07	0.23	0.33	0.24	0.04	0.03	0.06	6063	0-1M	594152	0	3	8	19	6	0	60	0	0	0	0	0	0	0	0	0	0	0	Highland
54	500750	1.32	0.00	0.35	Sep	0.29	0.71	71	0.06	0.19	0.58	0.00	0.06	0.01	0.09	14779	0-1M	745285	0	0	8	37	0	0	0	0	25	0	0	0	0	0	0	0	0	0	Highland
55	500752	1.38	0.24	0.45	Jul	0.46	0.54	104	0.04	0.20	0.50	0.00	0.03	0.04	0.20	5820	0-1M	328394	0	0	7	0	0	0	0	0	79	0	0	0	0	0	0	0	0	0	Highland
56	500754	1.02	1.54	1.54	Nov	0.25	0.75	48	0.02	0.23	0.42	0.00	0.17	0.05	0.10	2869	0-1M	103604	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Highland
57	500755	1.00	0.00	0.00	Apr	0.31	0.69	74	0.00	0.39	0.54	0.00	0.04	0.01	0.08	12517	0-1M	639860	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	93	Highland
58	500771	1.14	1.58	1.34	Jan	0.33	0.67	169	0.07	0.49	0.29	0.00	0.06	0.02	0.07	11481	1M-2.5M	1300585	4	17	3	40	5	0	0	0	17	0	0	0	0	11	2	0	0	0	Aberdeenshire
59	500784	1.03	1.22	0.77	Mar	0.27	0.73	143	0.11	0.47	0.31	0.00	0.04	0.01	0.05	16654	1M-2.5M	1739199	3	11	15	4	0	0	0	0	0	0	0	0	0	1	11	24	0	0	Fife
60	500789	1.10	0.53	0.37	Feb	0.17	0.83	136	0.05	0.20	0.41	0.08	0.07	0.07	0.13	28659	2.5M-5M	3213894	0	1	3	47	3	0	0	0	44	0	0	1	0	0	0	0	0	0	Highland
61	500793	1.00	0.82	0.00	Jun	0.53	0.47	154	0.07	0.38	0.39	0.00	0.07	0.01	0.09	24680	1M-2.5M	1780450	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	93	Argyll and Bute
62	500864	1.03	0.57	0.30	Aug	0.23	0.77	44	0.02	0.22	0.41	0.00	0.25	0.01	0.10	1630	0-1M	55890	0	0	0	0	92	0	0	0	0	0	0	0	0	0	0	0	0	0	Orkney Islands
63	500896	1.00	0.01	0.00	Jun	0.00	1.00	79	0.02	0.84	0.08	0.00	0.03	0.00	0.04	96851	5M-10M	7595906	12	0	13	11	3	0	4	9	3	0	6	0	0	0	5	0	0	0	Aberdeenshire
64	500958	1.14	1.82	1.30	Jul	0.27	0.73	200	0.07	0.64	0.20	0.01	0.04	0.01	0.03	6793	0-1M	985034	4	8	13	7	24	0	0	11	2	0	0	0	19	0	0	0	0	0	Shetland Islands
65	500965	1.06	1.17	0.30	Aug	0.10	0.90	121	0.06	0.50	0.29	0.03	0.06	0.01	0.03	12379	1M-2.5M	1343952	0	4	17	19	0	0	0	0	0	0	0	0	0	0	0	0	55	0	Fife
66	500968	1.15	0.59	0.75	Feb	0.20	0.80	128	0.02	0.46	0.25	0.12	0.07	0.03	0.05	9771	1M-2.5M	1000823	3	9	9	23	1	0	0	0	0	0	0	0	0	11	13	25	0	0	Moray
67	500973	1.10	0.34	0.27	Nov	0.33	0.67	86	0.06	0.35	0.25	0.00	0.18	0.05	0.12	11116	0-1M	638371	10	10	10	0	0	0	0	0	47	0	0	0	0	0	13	0	0	0	Orkney Islands
68	501001	0.97	1.71	1.47	Sep	0.20	0.80	237	0.06	0.62	0.24	0.00	0.03	0.01	0.04	19294	2.5M-5M	3665341	4	5	5	33	4	0	0	0	35	0	0	0	0	0	0	0	0	0	Shetland Islands
69	501021	1.00	0.07	-0.07	May	0.20	0.80	124	0.20	0.44	0.24	0.00	0.04	0.01	0.08	29851	2.5M-5M	2380865	15	21	11	14	0	0	0	0	9	0	2	0	0	0	4	4	0	0	Aberdeenshire
70	501022	1.00	0.00	0.00	Sep	0.31	0.69	112	0.13	0.65	0.15	0.00	0.02	0.00	0.04	24932	1M-2.5M	1923319	0	8	0	63	0	0	0	0	0	0	0	0	0	3	4	2	4	0	Roxburghshire
71	501031	1.04	2.25	2.44	Jul	0.18	0.82	221	0.05	0.49	0.29	0.00	0.08	0.03	0.06	6794	1M-2.5M	1224785	0	13	22	16	14	0	0	7	0	0	0	0	27	0	0	0	0	0	Shetland Islands
72	501092	1.07	0.98	1.30	Sep	0.23	0.77	148	0.09	0.32	0.45	0.01	0.04	0.03	0.07	15777	1M-2.5M	1791849	10	18	9	16	17	0	5	0	0	0	0	0	0	0	0	0	0	0	Orkney Islands
73	501152	1.12	1.52	2.14	Dec	0.31	0.69	162	0.05	0.57	0.27	0.02	0.04	0.03	0.09	7280	0-1M	812210	5	6	3	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	East Lothian
74	501153	1.25	0.67	0.38	Jan	0.19	0.31	156	0.09	0.30	0.44	0.05	0.02	0.02	0.09	9074	1M-2.5M	1145865	0	2	11	52	0	0	0	0	27	0	0	0	0	4	0	0	0	0	Dumfries and Galloway
75	501156	1.19	1.61	1.20	Feb	0.33	0.67	195	0.06	0.13	0.52	0.08	0.06	0.02	0.13	8689	1M-2.5M	1138277	0	0	2	7	0	0	0	0	3	0	0	0	0	0	0	0	0	0	Clackmannanshire
76	501158	1.06	1.48	0.38	Oct	0.14	0.86	156	0.05	0.39	0.32	0.15	0.04	0.02	0.04	8094	1M-2.5M	1088129	10	10	40	3	0	0	0	0	0	0	0	0	0	0	14	23	0	0	Perthshire
77	501159	1.04	2.25	1.23	Jun	0.17	0.83	191	0.06	0.47	0.33	0.04	0.06	0.02	0.08	14990	1M-2.5M	2359883	0	0	0	4	2	0	0	0	0	0	0	0	0	16	0	43	6	0	Clackmannanshire
78	501168	1.00	0.08	0.00	Apr	0.76	0.24	156	0.11	0.72	0.08	0.00	0.03	0.01	0.05	14651	0-1M	540004	0	26	21	13	15	0	0	0	0	0	0	0	0	5	10	0	10	0	Ayrshire
79	501169	1.07	0.47	0.47	Jan	0.38	0.62	156	0.04	0.47	0.30	0.06	0.06	0.02	0.05	17459	1M-2.5M	1683114	37	12	12	19	2	0	0	0	0	0	0	0	0	0	0	0	0	0	Ayrshire
80	501172	0.95	0.06	-0.02	Nov	0.07	0.93	162	0.02	0.30	0.22	0.17	0.06	0.02	0.21	54208	5M-10M	8200916	0	7	7	5	5	0	0	0	0	0	0	0	0	0	33	41	0	0	Lanarkshire
81	501200	1.00	-0.43	0.00	Feb	0.62	0.38	100	0.17	0.58	0.17	0.00	0.03	0.00	0.05	44720	1M-2.5M	1712131	2	4	3	35	0	0	0	0	35	0	0	0	0	0	0	0	0	0	Orkney Islands
82	501203	1.18	2.25	2.48	Aug	0.27	0.73	171	0.04	0.52	0.34	0.01	0.03	0.01	0.06	6430	0-1M	802854	0	7	1	31	9	0	49	0	0	0	0	0	0	0	0	0	0	0	Fife
83	501209	1.12	0.09	0.05	Mar	0.27	0.73	66	0.08	0.27	0.29	0.16	0.10	0.02	0.09	6958	0-1M	335990	0	0	7	0	0	0	0	0	0	0	0	0	93	0	0	0	0	0	Dunbartonshire
84	501217	1.03	0.92	0.34	Mar	0.40	0.60	191	0.04	0.65	0.15	0.10	0.03	0.01	0.03	46988	5M-10M	5363372	11	29	0	10	10	0	0	16	0	0	0	0	12	2	2	0	0	0	Moray

Appendix 5-1 DSS Input Data Continued…

Project	Project_ID	CP	FDA	QSFDA	On-site Start Month	Design Duration%	On-site Duration%	Project Duration(Weeks)	Proposal Team	Design Team	Project Management	Site Management Staff	Commissioning Team	H&S	Overheads	Average of Cost Intensity	First Net Construction Band	Sum of Total Construction Score	MCC/MCP	Elec Installation	Mechanical Install	Civil Installation	ICA	TTU	Chem Dosing	New Building	New Water Retain Structure	Power Supply	Tank	Security	RGF Refurbishment/New	Temporary Works	Pumps/Booster Set	Sludge/Bridges	Screen and Compactor/Grit Removal	SCADA	County
85	501220	1.27	1.83	1.55	Aug	0.35	0.65	168	0.04	0.53	0.31	0.04	0.03	0.01	0.04	6557	0-1M	720332	5	14	10	28	1	0	0	17	0	0	0	0	0	0	11	0	0	0	Fife
86	501226	1.27	0.60	0.60	Jul	0.27	0.79	76	0.03	0.42	0.35	0.01	0.08	0.02	0.09	7858	0-1M	471481	0	0	8	22	0	0	70	0	0	0	0	0	0	0	0	0	0	0	Perthshire
87	501232	0.97	0.51	0.56	Sep	0.15	0.85	194	0.04	0.62	0.23	0.01	0.02	0.01	0.06	41327	5M-10M	6800270	12	14	3	12	1	0	26	10	0	0	0	0	6	0	0	0	0	0	Angus
88	501233	1.00	0.00	1.32	Mar	0.19	0.81	32	0.06	0.49	0.25	0.00	0.00	0.00	0.19	14423	0-1M	375000	0	0	20	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Moray
89	501243	1.00	0.03	0.20	Mar	0.02	0.98	139	0.06	0.63	0.14	0.02	0.01	0.00	0.12	214557	Over 10M	2.93E+07	0	9	11	15	4	0	12	0	27	1	0	1	0	5	1	0	0	0	Aberdeenshire
90	501253	1.50	0.71	0.63	Jan	0.50	0.50	155	0.04	0.50	0.23	0.08	0.05	0.02	0.07	10205	0-1M	784365	12	9	4	16	0	0	0	0	23	0	0	0	0	0	0	10	4	0	Highland
91	501257	1.05	2.25	0.90	Jun	0.17	0.83	158	0.03	0.48	0.27	0.06	0.09	0.02	0.05	9735	1M-2.5M	1272069	24	23	12	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Aberdeenshire
92	501258	1.05	4.81	2.05	Apr	0.22	0.78	196	0.05	0.20	0.47	0.11	0.09	0.02	0.06	3903	0-1M	593780	0	2	13	9	42	0	20	0	0	0	0	0	0	0	0	0	0	0	Orkney Islands
93	501281	0.94	1.74	0.75	Jan	0.17	0.83	134	0.02	0.39	0.33	0.02	0.11	0.02	0.06	13626	1M-2.5M	1512440	26	52	4	9	0	0	0	0	0	0	0	0	0	9	0	0	0	0	Lanarkshire
94	501290	1.06	1.42	1.44	Jun	0.02	0.98	190	0.02	0.76	0.14	0.02	0.01	0.01	0.03	31092	5M-10M	5760960	9	9	4	34	0	0	0	0	0	0	0	0	0	0	5	6	13	0	Highland
95	501315	1.00	0.00	0.00	May	0.14	0.86	183	0.00	0.56	0.23	0.00	0.13	0.00	0.03	137	0-1M	21598	30	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Stirlingshire
96	501318	1.00	0.75	0.89	Sep	0.31	0.69	105	0.05	0.45	0.25	0.03	0.14	0.01	0.08	6252	0-1M	452681	26	52	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Aberdeenshire
97	501379	1.00	0.17	0.00	Dec	0.49	0.51	141	0.10	0.54	0.30	0.00	0.03	0.00	0.02	32559	1M-2.5M	2330323	9	13	12	5	1	0	46	0	0	0	0	0	0	0	0	0	0	0	Midlothian
98	501388	1.00	-0.02	0.00	Oct	0.55	0.45	125	0.05	0.57	0.17	0.01	0.03	0.01	0.12	43915	1M-2.5M	2473055	12	13	0	32	0	0	34	0	0	0	0	0	0	0	0	0	0	0	West Lothian
99	501390	1.07	-0.01	0.00	Nov	0.47	0.53	196	0.05	0.78	0.11	0.00	0.02	0.00	0.03	74630	5M-10M	7793704	6	15	22	30	4	0	14	0	0	0	1	1	0	0	2	0	0	0	Midlothian
100	501393	1.05	0.63	0.54	Feb	0.39	0.61	174	0.06	0.66	0.13	0.00	0.04	0.01	0.08	44317	2.5M-5M	4696971	4	6	3	16	2	0	6	7	8	0	0	0	0	0	5	0	0	0	Selkirkshire
101	501402	1.00	0.33	0.33	Mar	0.04	0.96	115	0.08	0.11	0.59	0.08	0.02	0.01	0.10	14305	1M-2.5M	1573577	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Lanarkshire
102	501418	0.98	0.72	0.73	Mar	0.24	0.76	183	0.06	0.63	0.17	0.06	0.02	0.01	0.05	30214	2.5M-5M	4223928	5	16	0	25	3	0	0	0	17	0	0	0	18	0	1	0	0	0	Isle of Lewis
103	501500	1.26	0.83	0.98	Dec	0.33	0.67	157	0.04	0.31	0.44	0.04	0.04	0.03	0.10	2825	0-1M	296600	0	6	4	5	8	0	0	0	0	0	0	0	0	0	0	0	0	0	Berwickshire
104	501514	1.15	1.34	0.61	Mar	0.31	0.69	115	0.05	0.51	0.27	0.05	0.05	0.01	0.06	6602	0-1M	528758	0	19	32	5	0	0	0	0	0	0	0	0	0	5	0	36	0	0	Lanarkshire
105	501518	1.07	2.13	1.79	Mar	0.16	0.84	139	0.04	0.50	0.22	0.04	0.04	0.03	0.13	20607	1M-2.5M	2474018	3	2	1	1	0	0	0	0	0	0	0	0	0	20	37	0	25	0	Ayrshire
106	501529	0.76	0.44	-0.20	Apr	0.14	0.86	50	0.06	0.10	0.50	0.04	0.03	0.02	0.22	14739	0-1M	635502	0	0	0	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Lanarkshire
107	501572	0.97	0.93	0.53	Mar	0.05	0.95	79	0.12	0.14	0.57	0.01	0.00	0.02	0.13	5577	0-1M	419411	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Argyll and Bute
108	501579	1.00	0.71	0.71	Nov	0.34	0.66	169	0.05	0.49	0.23	0.01	0.05	0.01	0.08	47035	5M-10M	5267905	3	10	9	25	0	0	0	0	0	0	0	0	0	6	0	21	16	0	Stirlingshire
109	501592	1.57	1.06	0.36	Nov	0.30	0.70	165	0.03	0.33	0.40	0.12	0.04	0.02	0.05	8627	0-1M	995645	7	8	2	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	Berwickshire
110	501603	1.00	0.05	0.00	Mar	0.71	0.29	145	0.07	0.68	0.16	0.00	0.06	0.00	0.02	27916	1M-2.5M	1172471	6	24	0	11	1	0	0	0	0	0	4	0	0	5	11	12	9	0	Ayrshire
111	501634	0.98	1.33	1.35	Sep	0.00	1.00	121	0.02	0.34	0.42	0.00	0.05	0.05	0.11	19813	1M-2.5M	2263340	0	4	0	17	4	70	0	0	5	0	0	0	0	0	0	0	0	0	Highland
112	501666	1.05	0.71	1.70	Mar	0.46	0.54	208	0.04	0.52	0.29	0.04	0.04	0.01	0.07	9705	1M-2.5M	1031121	35	21	0	9	0	0	0	0	0	0	0	0	0	0	22	0	0	0	Berwickshire
113	501731	1.00	0.09	0.00	Dec	0.01	0.99	75	0.00	0.78	0.16	0.01	0.02	0.00	0.02	54805	2.5M-5M	4054718	0	3	9	27	2	0	0	0	27	0	0	0	0	0	0	0	0	0	Stirlingshire
114	501950	1.06	1.75	0.59	Mar	0.37	0.63	213	0.07	0.59	0.23	0.03	0.02	0.00	0.07	7306	0-1M	973002	10	7	5	31	0	0	0	0	0	0	0	0	0	0	47	0	0	0	Dunbartonshire
115	502243	1.00	0.00	0.00	Oct	0.52	0.48	55	0.07	0.70	0.12	0.00	0.05	0.00	0.12	48090	1M-2.5M	1270550	0	40	40	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Aberdeenshire
116	502268	1.05	0.71	2.75	Nov	0.42	0.58	150	0.05	0.48	0.29	0.02	0.01	0.03	0.11	14487	1M-2.5M	1272300	0	62	0	5	0	0	0	5	0	0	0	0	0	29	0	0	0	0	Lanarkshire
117	502358	1.00	-0.03	0.00	Nov	0.06	0.92	86	0.00	0.75	0.21	0.00	0.02	0.00	0.01	35753	2.5M-5M	2855707	5	7	8	50	3	0	0	0	40	2	0	0	0	2	0	0	0	0	Dumfries and Galloway
118	502598	0.77	1.92	1.52	Jul	0.17	0.89	51	0.06	0.25	0.53	0.03	0.01	0.02	0.11	17316	0-1M	783654	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Stirlingshire
119	502648	1.02	0.16	-0.79	Aug	0.38	0.62	137	0.16	0.51	0.24	0.00	0.02	0.00	0.05	41450	2.5M-5M	3530044	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Argyll and Bute
120	502649	1.00	-0.03	0.00	Dec	0.49	0.51	152	0.15	0.54	0.20	0.02	0.01	0.00	0.07	55122	2.5M-5M	4235525	7	11	40	0	3	0	0	16	0	0	1	0	0	6	2	0	0	0	Argyll and Bute
121	502650	1.00	0.21	-0.20	Jul	0.40	0.60	150	0.11	0.57	0.19	0.00	0.03	0.01	0.09	36450	2.5M-5M	3251782	7	7	5	24	3	0	0	11	0	0	0	0	0	0	1	0	0	0	Argyll and Bute
122	502651	1.09	-0.39	0.00	Apr	0.50	0.50	127	0.06	0.59	0.13	0.00	0.06	0.01	0.14	7019	0-1M	447181	0	18	3	68	8	0	0	0	0	0	0	0	0	0	0	0	0	0	Ayrshire
123	502960	1.00	-0.04	-0.34	Jul	0.27	0.79	76	0.07	0.12	0.41	0.24	0.14	0.01	0.06	65047	2.5M-5M	3874926	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	Ayrshire
124	503148	1.00	-0.42	-0.41	Apr	0.10	0.90	36	0.07	0.09	0.75	0.01	0.03	0.03	0.08	15216	0-1M	483094	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Midlothian
125	503266	1.06	1.71	1.49	Jul	0.28	0.72	104	0.12	0.34	0.29	0.10	0.08	0.01	0.05	9935	0-1M	750266	7	10	7	3	11	0	50	0	0	0	0	0	0	0	0	0	0	0	Selkirkshire
126	503434	1.00	0.00	0.07	Dec	0.14	0.86	57	0.09	0.51	0.32	0.00	0.02	0.00	0.06	22514	1M-2.5M	1103797	0	0	8	93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Lanarkshire

Appendix 6-1 DSS Questionnaire

Decision Support System Demonstration Questionnaire

Brief

The purpose of this questionnaire is to gather comprehensive feedback from industry experts like yourself regarding the functionality, usability, and potential impact of our Decision Support System (DSS) tailored for construction project sustainability. We value your insights to help us understand how well the DSS meets industry needs, especially in optimizing project performance, enhancing sustainability, and improving decision-making processes. Your feedback will contribute to the refinement and validation of our DSS for future development.

Demographics

Job Title

Duties/Responsibilities

Years of experience

Video Demonstration Feedback

Q1 Did you watch the video demonstration of the Decision Support System (DSS)? (Yes/No)

Q2 What is the overall usefulness of the DSS based on the video?

Not Useful	Slightly Useful	Moderately Useful	Useful	Very Useful

Understanding and Clarity

Q3 After watching the video, how well do you feel you understand the key features and functionalities of the DSS?

Not at all	Slightly	Moderate	Well	Very Well

Q4 How clear and easy to follow were the explanations and demonstrations in the video?

Not at all	Slightly	Moderate	Clear	Very Clear

DSS User Interface (UI)

Q5 How would you rate the ease of use of the DSS's user interface based on the video demonstration?

Very Difficult	Difficult	Moderate	Easy	Very Easy

Q6 Did you find the DSS's user interface visually appealing and well-organized based on the video demonstration?

Not at all	Slightly	Moderate	Appealing	Very Appealing

Flexible Feature Importance Tool

Q7 Did the video demonstration provide a clear understanding of how the Feature Importance Tool works within the DSS?

Not at all	Slightly	Moderate	Clear	Very Clear

Q8 How well do you understand the role and significance of the Feature Importance Tool based on the video?

Not at all	Slightly	Moderate	Well	Very Well

Q9 How valuable do you consider the Tool's functionality for optimizing project performance in construction projects?

Not at all	Slightly	Moderate Value	Valuable	Very Valuable

Q10 How valuable do you consider the Tool's functionality for optimizing for each performance metric? (Commercial Performance, Forecast Duration Accuracy at PO and On-Site Forecast Duration Accuracy)

Not at all	Slightly	Moderate Value	Valuable	Very Valuable

Q11 How valuable do you consider the adaptability of the Feature Importance Tool for future datasets and metrics?

Not at all	Slightly	Moderate Value	Valuable	Very Valuable

Q12 How confident are you in the accuracy of the Feature Importance Tool's assessments based on the video demonstration?

Not at all	Slightly	Moderately	Confident	Very Confident

Q13 To what extent do you believe the tool can help project managers make informed decisions for project performance?

Not at all	Slightly	Moderately	Very Much	Extremely

All Tools in the DSS

Q14 Based on the demonstration, how well do you think the DSS can assist in improving construction project sustainability?

Not at all	Slightly	Moderately	Very Much	Extremely

Q15 How relevant do you find the DSS's functionalities to the construction industry's needs and challenges?

Not at all	Slightly	Moderately	Relevant	Extremely Relevant

Q16 Based on the demonstration, how well do you think the DSS integrates various tools to support decision-making for construction projects?

Not at all	Slightly	Moderate	Well	Very Well

Q17 How likely would you be to recommend the DSS, based on the video demonstration, to colleagues or peers in the construction industry?

Not Likely	Slightly Likely	Moderately Likely	Likely	Very Likely

Please share your reasons or comments below

Q18 Which specific tool or feature within the DSS, as demonstrated in the video, do you find most promising or valuable? Please specify and provide reasons for your answer.

Q19 Were there any specific aspects or functionalities of the DSS that you believe require further improvement or clarification based on the video demonstration? If so, can you provide reasons for your answer.

Q20 Are you aware of any similar decision support tools or systems currently used in your sector or within your company? If so, how do they compare to the DSS from the demonstration?

Q21 Is there any additional feedback or comments you would like to provide regarding the DSS and its features, based on the video demonstration?

Thank you for taking the time to participate in this questionnaire. Your input is invaluable to our research and the development of the DSS. Your insights will contribute to the improvement and refinement of our system for the future

Appendix 6-2 DSS Validation Distributed Email

CS

Craig Smith

To: Angus Liu (Galliford Try) <angus.liu@gallifordtry.co.uk>

Tue 03/10/2023 16:20

Cc: Andy Wong

Hi, Angus.
As requested, I have created a link which should be accessible for the video, and I have created an online active survey with the second link. These should be fine for distribution as I have tested them.

Video Demonstration
https://strath-my.sharepoint.com/:v:/g/personal/craig_smith_strath_ac_uk/EZU3OAcwQX1Bowkx8ss13K0BcXFNGDskxAUIXVU8Jjyu5g?e=kVt35B

Link to Questionnaire
https://docs.google.com/forms/d/e/1FAIpQLSeditGwNCcie71_16SwNezwz-69f1CJhIE2fpV0x0jR6-m-Xw/viewform?usp=pp_url

Kind regards
Craig

Appendix 6-3 Validation Closed Question Responses

	Numbered Likert Scale					Summary	
Description	1	2	3	4	5	Average	Standard Deviation
What is the overall usefulness of the DSS based on the video?			1	4	2	4.14	0.64
After watching the video, how well do you feel you understand the key features and functionalities of the DSS?			2	3	2	4.00	0.76
How clear and easy to follow were the explanations and demonstrations in the video?			2	2	3	4.14	0.83
How would you rate the ease of use of the DSS's user interface based on the video demonstration?			1	5	1	4.00	0.73
Did you find the DSS's user interface visually appealing and well-organised based on the video demonstration?			4	1	2	3.71	0.88
Did the video demonstration provide a clear understanding of how the Feature Importance Tool works within the DSS?			3	2	2	3.86	0.83
How well do you understand the role and significance of the Feature Importance Tool based on the video?			2	3	2	4.00	0.76
How valuable do you consider the Tool's functionality for optimising project performance in construction projects?		1		4	2	4.00	0.93
How valuable do you consider the Tool's functionality for optimising for each performance metric?			1	4	2	4.14	0.64
How valuable do you consider the adaptability of the Feature Importance Tool for future datasets and metrics?			1	2	4	4.43	0.73
How confident are you in the accuracy of the Feature Importance Tool's assessments based on the video demonstration?			4	2	1	3.57	0.73
To what extent do you believe the tool can help project managers make informed decisions for project performance?		1	1	4	1	3.71	0.88
Based on the demonstration, how well do you think the DSS can assist in improving construction project sustainability?		2	1	3	1	3.43	1.05
How relevant do you find the DSS's functionalities to the construction industry's needs and challenges?			1	4	2	4.14	0.64
Based on the demonstration, how well do you think the DSS integrates various tools to support decision-making for construction projects?		1	1	4	1	3.71	0.88
How likely would you be to recommend the DSS, based on the video demonstration, to colleagues or peers in the construction industry?		1	1	3	2	3.86	0.99

Appendix 6-4 Validation Questionnaire Q17

Participant	Score	Reason for Response
Senior planner	4	It was a good starting point towards our Source to Sea goal
Data & Systems Manager	3	A possible way to enhance the effectiveness of the DSS is to involve the delivery team in the data analysis process from the beginning of the project. By doing so, the delivery team can gain a better understanding of the data sources, quality, and limitations, as well as the objectives and expectations of the DSS. This can help them to align their work with the data-driven decision-making framework and to identify and address any potential issues or gaps in the data. Furthermore, this can foster a collaborative and transparent culture among the delivery team and the stakeholders, which can improve the trust and acceptance of the DSS outcomes.
Data Analyst	4	Individual feature analysis allows for consideration metrics for long term goals
PMO Manager	5	This tool takes away perception which the construction industry is still largely guided by.
Technical Manager	4	I feel the tools show here are very important to progressive improvement. But feel that I would need to understand the data sets as used and I could not see any risk data being used to influence or validate the costs and time-based changes.
HS&E Systems Manager	2	The concept based on black and white answers and when undertaking a project there are so many variables, the hazards, the skillset of those undertaking the works, the workplace itself, from HS&E perspective the impact of the hazards can change, so how can these be factored in - it is making decisions on generic information and generic can be dangerous; it concerns me that people will not 'think' to add in the other considerations. I am concerned people will stop thinking and reliance totally on the DSS
Innovation and Research Lead	5	This tool has the clear capability to support better decision making in the industry. In my role, I look forward to working with Craig and others to harness the tool.

Appendix 6-5 Validation Questionnaire Q18

Job Title	Which specific tool or feature within the DSS, as demonstrated in the video, do you find most promising or valuable? Please specify and provide reasons for your answer.
senior planner	Prediction of projects outcome
Data & Systems Manager	One possible application of the project nature tool within the DSS is to enable the users to import their existing projects and use the selectors to fine-tune them. This way, the users can leverage the DSS's capabilities to improve their project performance, quality and efficiency. The project nature tool can also provide feedback and suggestions on how to optimize the projects based on the selected criteria.
Data Analyst	The feature importance and performance trends will allow for a degree of anticipation and decision making for project timelines
PMO Manager	Future forecasting, as this will help inform cost forecasts and spend profiles which are key to running frameworks
Technical Manager	Mass collection of the data and the analytics allows greater understanding. The best value here is the great start that can be built upon
HS&E Systems Manager	Trends that can be considered when planning the works; however, would people not start a project in December because of trends? An interesting concept similar to that used for 'Intuity' - where predictions made from previous data added; but consistency of data entry is also something than could alter the value.
Innovation and Research Lead	All looks useful but the project nature element has clear use cases.

Appendix 6-6 Validation Questionnaire Q19

Job Title	Were there any specific aspects or functionalities of the DSS that you believe require further improvement or clarification based on the video demonstration? If so, can you provide reasons for your answer.
senior planner	More external factors that may affect company performance, such as clients, 3rd parties, over economic in the industry, etc.
Data & Systems Manager	At this time, I do not think there are any additional features that could enhance the DSS. However, I suggest that we update our dataset with the latest data to gain a better understanding of the current situation and trends.
Data Analyst	While the project nature page allows for deep comparisons, there are too many definable parameters for an overview over projects of such scale for easy and definable comparison within the current layout. The requirement to find these optimal areas within definable parameters adds another degree of complexity
PMO Manager	None
Technical Manager	Risk overlayed with both time and money
HS&E Systems Manager	
Innovation and Research Lead	It would be useful to incorporate additional inputs such as type of work, form of contract and procurement methodology.

Appendix 6-7 Validation Questionnaire Q20

Job Title	**Are you aware of any similar decision support tools or systems currently used in your sector or within your company? If so, how do they compare to the DSS from the demonstration?**
Senior Planner	Yes, there are many and very powerful such as the Oracle analytic cloud, 4D-BIM system etc
Data & Systems Manager	No
Data Analyst	Utilising expert knowledge and experience with PMO tools such as Primavera. We are able to confidently manage projects. The DSS does allow however for previous insight in detail, especially in the characteristics of a project.
PMO Manager	Not aware
Technical Manager	Engineering is in development of a process of time, resource and cost management for Design stage.
HS&E Systems Manager	
Innovation and Research Lead	None that I am aware of.

Appendix 6-7 Validation Questionnaire Q20

Job Title	Is there any additional feedback or comments you would like to provide regarding the DSS and its features, based on the video demonstration?
senior planner	
Data & Systems Manager	
Data Analyst	You said consideration was placed on sustainability, but no metrics were given on how sustainability can be achieved.
PMO Manager	Excellent presentation, very clear and informative
Technical Manager	I think I would like to assist if possible
HS&E Systems Manager	
Innovation and Research Lead	Excellent work so far. I am happy to support any additional work in this area.