Machine Learning and Case-Based Reasoning for Damage Stability Decision Support

PhD Thesis

Panagiotis Louvros

Maritime Safety Research Centre
Department of Naval Architecture and Marine Engineering
University of Strathclyde, Glasgow

April 23, 2025

Abstract

This thesis investigates the integration of Machine Learning (ML) and Case-Based Reasoning (CBR) to enhance decision support for real-time damage stability assessment in passenger ships under dynamic flooding scenarios. Despite advances in maritime safety regulations, flooding incidents remain a leading cause of catastrophic vessel loss, necessitating innovative approaches to assess and mitigate risks. The research presents a novel framework that combines probabilistic and case-based methodologies, leveraging simulations to predict outcomes such as time to capsize (TTC) and critical survival factors.

Key contributions include the development of a probabilistic decision support system that utilises Bayesian and Dempster-Shafer approaches to fuse predictions from ML and CBR, effectively managing uncertainty and enhancing situational awareness during emergencies. This system incorporates real-time monitoring, feature engineering from historical incident data, and dynamic simulations to account for complex flooding patterns. The methodology is validated through case studies and a Monte Carlo-based uncertainty analysis, demonstrating its efficacy in improving accuracy and reliability.

The research highlights the implications of these advancements for passenger ships, offering insights into their survivability under extreme conditions. It aligns with industry trends and regulatory objectives, contributing to safer ship designs and operational strategies. Recommendations for future work include exploring advanced computational models, integrating human factors, and expanding the framework to other ship types, emphasising the adaptability and scalability of the proposed solution.

Contents

List of Figures

List of Tables

Nomenclature

CBR Case-Based Reasoning

CFD Computational Fluid Dynamics

DST Dempster-Shafer Theory

EMSA European Maritime Safety Agency

GUI Graphical User Interface

GZMAXR GZ Maximum Range

IMO International Maritime Organization

MAE Mean Absolute Error

MC Monte Carlo (simulation)

ML Machine Learning

MSE Mean Squared Error

RMSE Root Mean Square Error

RMT Risk Modelling Tool

Ro-ro Roll-on, Roll-off vessel

SFAC Survival Factor also found as SFACSOL

SOLAS Safety of Life at Sea

SRtP Safe Return to Port

TTC Time to Capsize

WT WaterTight

Acknowledgements

Throughout my PhD and research career, there have been various people who have helped and supported me in carrying on and completing it. I would like to sincerely thank my supervisor, Evangelos Boulougouris for the years of mentorship, advice and research direction but also friendship. My fellow PhD students, researchers and staff have also been there for discussions, support, sharing of ideas and comradeship. I am thankful to have interacted with fellow MSRC colleagues like: Charalampos Tsoumpris, Alexandros Komianos and Paul Lee, Georgios Atzampos, Alistair Murphy, Romanas Puisa, Joao Dantas, and others

I am very grateful to my friends Fotios Stefanidis and Evangelos Stefanou with whom we worked and lived many years during our PhD journeys together. Their emotional and material support throughout all those years together will never be forgotten.

To my family and specifically my mother Natasa, and father Nikos I owe the unconditional, consistent and persistent love and support for me to attempt and persevere with my studies, finally culminating in this academic level. Thank you also little sister, Vera, for your constant encouragement and love.

My editor-in-chief and partner, Mersene, has also been there for me and helped me finish this work also having reviewed and checked my rumblings which are hopefully made more manageable to read by her efforts.

The author's work has been indirectly and directly supported by two externally funded projects. The author would like to acknowledge and identify the Norwegian Research Council funded project SEAMAN (269467) and the European Commission funded project SafePASS (Grant number:815146) as crucial in the delivery of this work.

Chapter 1

Introduction

1.1 Maritime safety & Flooding

Maritime safety is at the crossroads of technology, regulatory frameworks, and human ingenuity, where every component of the design and operation of a vessel is carefully calibrated to meet the demands of a challenging environment. As the shipping industry evolves, so do the complexities of ensuring safety against a backdrop of expanding global trade, climate variability, and technological transformation. Although maritime operations have seen significant improvements, the fundamental risk posed by flooding remains an ongoing focus due to its severe implications for life and property.

Flooding incidents arise from various causes, including structural failures, heavy weather, and even design flaws. Although historical responses to these incidents have been largely centred on improving material strength and emergency equipment, recent advances have introduced sophisticated monitoring and predictive capabilities. Technologies such as real-time flood detection and autonomous response systems are promising tools for the future of maritime safety. Yet, the efficacy of these technologies heavily relies on their integration within robust regulatory structures and adherence to guidelines set forth by bodies like the International Maritime Organization (IMO).

The evolution of maritime safety regulations is punctuated by significant milestones, each catalysed by high-profile maritime tragedies that brought attention to vulnerabilities in ship safety. As regulatory bodies and classification societies developed frameworks such as SOLAS, these events changed public and industry expectations toward a preventive approach to safety. However, despite these advances, certain limitations persist within current regulatory frameworks, particularly regarding damage stability criteria, which often fail to account for complex, dynamic scenarios like multi-compartment flooding or varying sea conditions.

1.1.1 Introduction

Maritime safety has been a critical concern since the early days of seafaring, with the primary objective being the protection of lives, cargo, and vessels in one of the most inherently hazardous environments. Among the many risks faced by ships at sea, flooding poses one of the most serious threats to vessel stability and survival. Whether caused by hull breaches, extreme weather conditions, or operational mishaps, flooding can lead to the loss of buoyancy, capsize, and ultimately, the sinking of a vessel. Over the centuries, the maritime industry has developed and refined various safety practices, technologies, and regulations aimed at mitigating these risks.

This chapter provides a comprehensive background on maritime safety, focusing on the specific dangers posed by flooding. It explores the historical evolution of safety regulations, the role of international frameworks such as the Safety of Life at Sea (SOLAS) convention, and the influence of classification societies. It also examines how technological advancements and modern ship design have improved the ability to prevent and manage flooding incidents, significantly reducing the risks associated with maritime operations.

1.1.2 Historical Overview of Maritime Safety

The history of maritime safety is marked by significant disasters that have spurred the development of more stringent safety measures. Early ships were vulnerable to a wide range of hazards, including storms, poor navigation, and structural failures. Shipping has always been a very important aspect of human activities, namely trade, warfare, cultural dispersion, and communication. hence the loss of vessels at sea is a tale as old as seafaring itself. In fact, in an age where even the knowledge in the inherent safety of the vessel was non-existent, the operation of ships was largely a result of intuition and the safety measures were few and far between. The combination of the importance of shipping and the dangerous nature of it, motivated various technological, societal and cultural innovations. One of the earliest examples is perhaps several "maritime laws" - including shipbuilder's warranty, insurance, wage and outfitting standards., found within what is commonly refereed to as "The code of Hammurabi". These "laws" are found in cuneiform text on decorated basalt steles ordered by the ancient Babylonian King Hammurabi composed during 1755-1750 BC. Some relevant ones are shown below:

Law 235:

> *If a shipbuilder builds a boat for someone, and does not make it tight, and during that same year that boat is sent away and suffers injury, the shipbuilder shall take the boat apart and put it together tight at his own expense. The tight boat he shall give to the boat owner.*

Law 236:

> *If a man rents his boat to a sailor, and the sailor is careless, and the boat is wrecked or goes aground, the sailor shall give the owner of the boat another boat as compensation.*

Law 237:

> *If a man hires a sailor and his boat, and provides it with corn, clothing, oil, dates, and other things of the kind needed for fitting it: if the sailor is careless, the boat is wrecked, and its contents ruined, then the sailor shall compensate for the boat which was wrecked and all in it that he ruined.*

Of course various other civilizations similarly advanced the legal aspect of shipping before finally culminating to the modern rules of the 19th century. For example the first mention of a "Load line Convention" can be found in the Venetian & Genoese republics [Boisson, 1998].

In terms of technological advancements, safety invariably improved. Initially, with more sturdy constructions starting with copper sheathing (which preserved the hull from marine life induced degradation), iron fasteners and later whole iron and steel hulls. With modern times came better metallurgy and materials, along with formal structural assessment with computational tools, as well as inspections and strict maintenance schedules provide a great degree of confidence in modern vessels. Navigational hazards have also been combated to a large degree with modern satellite localization, communications and charting advances. As ships became larger and more complex, the need for formal safety regulations became evident. The sinking of the RMS Titanic in 1912, which resulted in the loss of over 1,500 lives, was a turning point in maritime history. It highlighted the limitations of existing safety protocols, particularly regarding lifeboat capacity, watertight subdivision, and communication during emergencies. Where previously sparse rules existed that mandated few aspects of the ship design, construction and operation, the sinking of the Titanic brought forward a new era of consolidated, international rules as the "base minimum" level of safety required for ships belonging to the signatory parties. The first International Convention for the Safety of Life at Sea (SOLAS) was adopted in 1914. This treaty established minimum safety standards for the construction, equipment, and operation of ships. Over time, SOLAS has been amended and expanded to incorporate lessons learned from subsequent maritime incidents and technological advances, making it the cornerstone of modern maritime safety regulations. More discussion on these can be found in Chapter 2.

Nowadays, maritime safety is still improved upon. Certainly, maritime accidents never stopped occurring, but the impetus that instigated the original SOLAS as well as the amendments of the 60s & 70s is much diminished. On one hand, better tools and technological advancements have made the vessels inherently safer, on the other hand, the application of the refined rules and regulations has obviously also had the intended effect in reducing marine casualties. Maritime accidents and associated loss of life have indeed decreased significantly since 1900. This trend is supported by several key findings:

According to a study examining British military, mining, and maritime accidents since 1900, there have been widespread reductions in fatal accident rates across various hazardous occupations, including merchant shipping. The research found that fatal accident rates in merchant shipping were highest from 1900 to 1914, ranging from 400 to 600 deaths per 100,000 workers. However, these rates have dramatically decreased over time, with the greatest reductions occurring in recent years [Roberts et al., 2021].

The global shipping industry has seen a notable improvement in safety over the past decade. Improved ship design, technology, regulation, and risk management systems have contributed to a 70 % drop in reported shipping losses [Dominguez-Péry et al., 2021]. This significant reduction in accidents and losses indicates a substantial decrease in maritime incidents and associated fatalities since the early 20th century.

While specific long-term data on passenger deaths is limited, a study examining crew and passenger deaths from vessel accidents in the United Kingdom since 1900 also supports the overall trend of declining maritime fatalities [Carter et al., 2019].

Looking at the EMSA statistics [EMSA, 2023] there is a clear decline in fatalities related to flooding incidents but also interestingly the number of accidents that relate to damage stability are consistently the biggest contributing case. Note that these accidents refer to EU flagged vessels, vessels with EU interests or accidents that were reported within EU territorial or internal waters (UNCLOS definition).

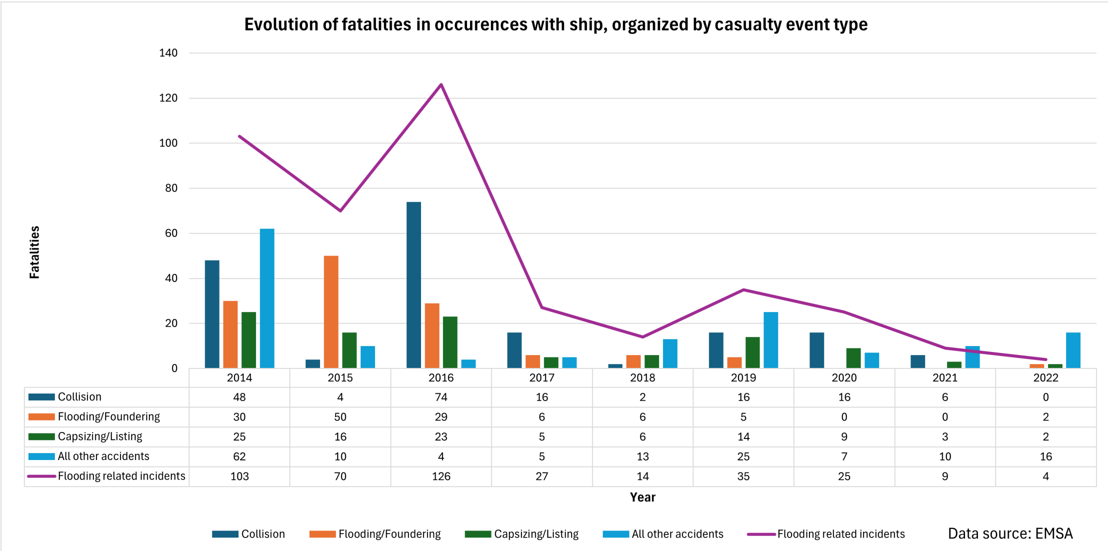

Figure 1.1: Fatalities per event occurrence type from EMSA statistics [EMSA, 2023]

Considering the EU fleet as being an indicative sample of the world fleet but also representing more advanced and strictly regulated ships (see also figure below), it becomes clear that flooding incidents have not been eradicated and continue to claim vessels and souls.

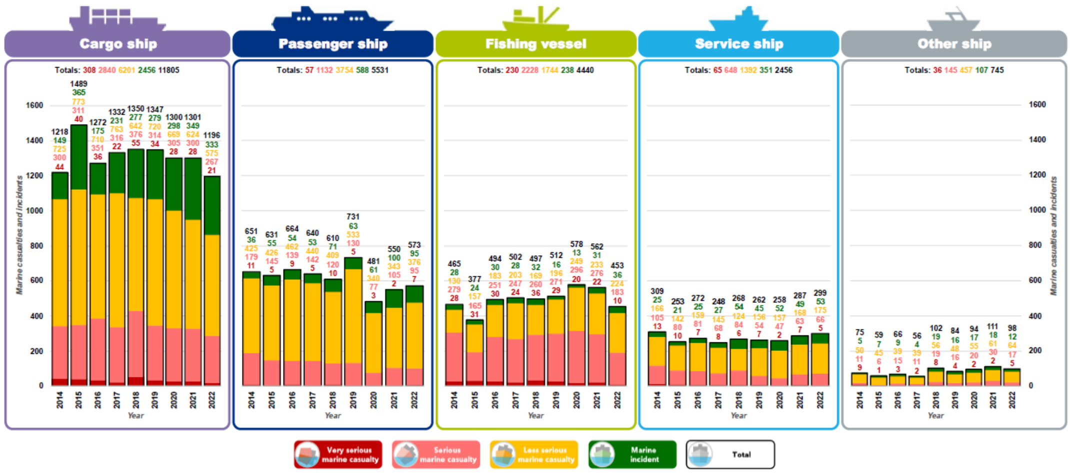

Figure 1.2: Evolution of number of marine casualties and incidents, organized by severity and ship type [EMSA, 2023]

Foundering and capsize are also implicitly linked with either significant loss of life in case of passenger ships and especially ferries (due to large amount of passengers carried) or marine pollution in all cases of ships, but especially crude or product tankers. Due to the nature of the transport medium, a single such accident can have large and lasting effects as evidenced in the aftermath of the Exxon Valdez or Estonia sinkings.

Despite the overall improvement in maritime safety, flooding incidents remain a significant concern, particularly for passenger ships and ferries. The SOLAS 2009 amendments marked a shift from prescriptive requirements to a probabilistic approach in damage stability regulations. This change aimed to provide a more comprehensive and risk-based assessment of a vessel's survivability in case of flooding; however, the effectiveness of these regulations in addressing all aspects of damage stability has been subject to ongoing scrutiny and research.

Several studies have indicated that while the probabilistic approach offers advantages, there may be more efficient methods for establishing the safety level of passenger ships.

Time-domain numerical simulations have been proposed as a potentially more accurate tool for assessing damage stability. This approach allows for a more dynamic evaluation of a vessel's behavior in various flooding scenarios, potentially capturing complex phenomena that might be overlooked in static calculations.

The Joint Industry project eSAFE (enhanced Stability After Flooding Event), funded by the Cruise Ship Safety Forum in 2016, represents a significant step towards improving damage stability assessment. This project aimed to enhance the damage stability of cruise ships by incorporating modern first-principle tools into the early design process. Such initiatives demonstrate the industry's commitment to continually improving safety standards beyond regulatory requirements.

Another notable effort in this direction is the FLARE (Flooding Accident REsponse) project, which commenced in 2019. This project seeks to develop a novel risk-based methodology for 'live' safety assessment, aligning with the IMO's high-level goals for passenger ship safety. The FLARE project recognizes the need for a more dynamic and responsive approach to safety assessment, particularly in crisis situations.

The s-factor, a key component in the SOLAS 2009 survivability calculations, has been a subject of particular interest. Based on the findings of the HARDER project, this factor has been scrutinized for its accuracy in predicting real-world survivability. A study led by EMSA in 2009 highlighted discrepancies between calculated and experimentally observed critical sea states, suggesting that the current formulation might underestimate survivability in certain cases. To address these concerns, recommendations have been made to adjust the SOLAS targeting values for GZmax and Range. However, subsequent analysis by [Tsakalakis, 2012] indicated that these adjustments had minimal impact on the Attained index, highlighting the complex interplay of factors in damage stability calculations.

The persistent challenges in accurately assessing damage stability underscore the need for continued research and development in this field. As ship designs evolve and new technologies emerge, particularly in the context of decarbonization efforts, the maritime industry must ensure that safety standards keep pace with these changes.

Future research directions may include:

1. Advanced computational methods: Developing more sophisticated numerical models that can accurately simulate complex flooding scenarios, including the effects of internal arrangements and dynamic sea states.

2. Real-time monitoring and assessment: Exploring technologies for continuous, onboard assessment of a vessel's stability status, allowing for proactive measures in potentially dangerous situations.

3. Integration of human factors: Investigating the role of crew response in damage control situations and incorporating these factors into safety assessments.

4. Novel design approaches: Exploring innovative hull designs and internal arrangements that inherently enhance damage stability without compromising operational efficiency.

5. Risk-based methodologies: Further developing and refining risk-based approaches to damage stability assessment, potentially allowing for more tailored and effective safety measures.

In conclusion, while significant progress has been made in maritime safety over the past century, the persistent threat of flooding incidents necessitates ongoing vigilance and innovation in damage stability regulations and assessment methods. The maritime industry's continued focus on this area, as evidenced by projects like eSAFE and FLARE, demonstrates a commitment to enhancing safety beyond mere regulatory compliance. As we move forward, the integration of advanced technologies, improved computational methods, and a more holistic understanding of ship behavior in damaged conditions will be crucial in further reducing the risks associated with flooding incidents and ensuring the safety of passengers and crew at sea.

1.2 Motivation

The motivation behind this research stems from the persistent risks associated with maritime flooding incidents and the need for robust, proactive safety methodologies in an industry that continues to evolve. Despite advances in ship design, regulatory frameworks, and technological support systems, maritime flooding remains one of the leading causes of catastrophic vessel loss and human casualties. In particular, issues such as dynamic stability in complex flooding scenarios, the use of improper damage stability tools, and absence of dedicated decision support systems to advice crew during these emergencies are identified as pertinent to this work. The implicit focus of this research is on passenger ships whose arrangements and inherent potential fatalities increase the need for an improvement in their survivability. All the methodologies are applicable to any ship but with reduced urgency and less improvement compared to conventional means. Several accidents of the past 20 or so years have served as an inspiration for this research, guiding the research questions and methodology development in such a way so as to address the leading causes of those tragedies. Several of them are analyzed in subsequent chapters.

1.2.1 Challenges

Below are the main challengers identified and addressed throughout this work.

1. **Real time damage stability assessment** The whole work is predicated on the presence of a real-time, fast and accurate method of predicting the damage stability

characteristics of the vessel in the future. That includes expected roll motions & pitch motions, flooded rooms and others but most importantly the time to sink or capsize (usually referred to as TTC). TTC is generally thought of as the time available for any remedial actions to take place. It is also assumed that "sinking or capsize" coincides with a survival factor of 0. Estimating the time to capsize is a challenging task, as the method needs to account for internal progressive flooding, external environmental conditions and also be inherently probabilistic as various processes are not deterministic - such as unexpected openings or errors in sensed data used to make the predictions.

2. **Simulations and data handling** To properly capture the flooding event, numerical simulations are required. Moreover, since this approach has to happen in real-time these simulations cannot be run ad hoc; due to the stochastic nature of the phenomena multiple simulations are required. The number of simulations needs to be warehoused, accessed and used appropriately. This requires a scheme to store and extract the relevant information when needed.

3. **Decision support** Finally, the objective of this methodology is to arrive at solutions or information that can readily be shown to the crew or decision maker on board and during an emergency. This requires it to be accurate in terms of correctly predicting the outcome but also transparent in terms of the uncertainty of this prediction. Furthermore, there should be a relationship between depicted values and suggested actions on board in a clear and actionable manner.

1.3 Thesis Structure

This thesis is structured in a way that allows the reader to follow each constituent part of the methodology; in the beginning, as separate entities that finally are integrated together. To achieve this, separate chapters are used or others are internally subdivided. The two methodologies are first introduced with relevant literature review which explains their historical origin and use -especially in the sectors similar to the present work. Then for each methodology their "architecture" and high level application rationale is presented separately in the "Methodology" chapter 4. Lastly, each method's implementation is presented in a separate chapter. Their integration and joint evaluation is then done in chapter 7. Chapter 8, "Decision Support System" further elaborates on the use the two methods to create the decision support elements of the methodology discussing also the appropriate use of the tool during an emergency.

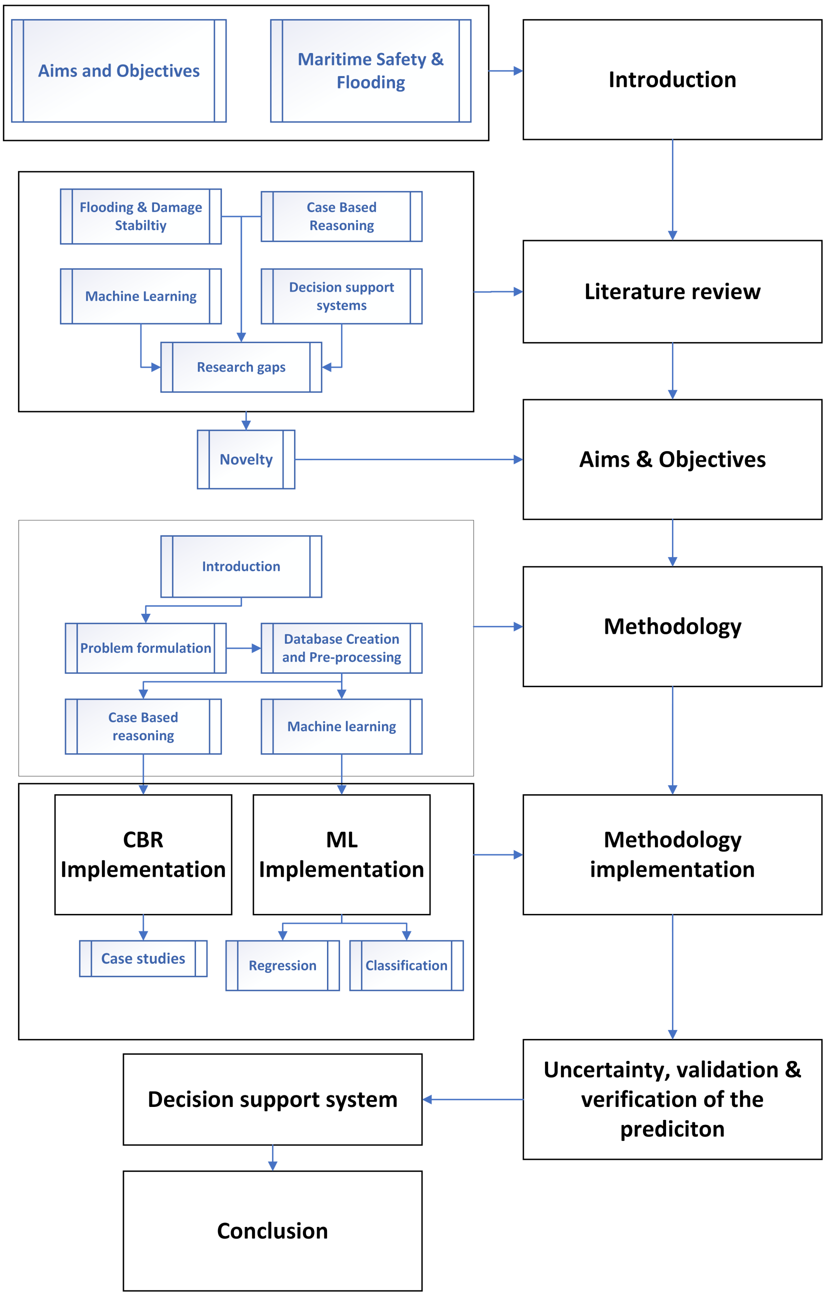

Figure 1.3: Thesis structure as a flowchart: Bold black text stands for the main 9 Chapters of the thesis and arrow connections depict the flow between chapters and subchapters

Since the methodologies of machine learning and case-based reasoning are not normally found within the naval architecture field but have been increasingly becoming relevant, the author elects to introduce their theory and basic properties, including ways to gauge performance as well as the rationale and process behind them so that readers less experienced in those subjects can have a good understanding of their ideas and most importantly how they integrate within this work.

Chapter 2

Literature Review

2.1 Flooding

2.1.1 Introduction to Damage Stability and Flooding

Flooding refers generally to a significant lapse in the vessel's watertight integrity that allows water to enter, burdening the vessel. If allowed to continue and progress it can lead to the ship foundering in various ways. Typically, for most ships as the floodwater mass increases, the vessel's displacement and waterline rise, usually leading to more openings going underwater, accelerating the inflow of floodwater until all buoyancy is lost and the vessel sinks. If this happens without the attitude of the vessel changing much, it is generally referred to as a "gracious sinking" as it is not characterized by large motions and is usually the slowest the vessel can sink. For example, the sinking of the Titanic fits that description. Another possibility is that the presence of the water inside can quickly alter the stability of the vessel leading to capsize. This event is usually catastrophic as it allows no time for remedial or life-saving actions to take place. For example, the sinkings of Estonia and Herald of the Free Enterprise are characterized by large sudden roll moments and subsequent overturning (partial or complete). In other cases, the floodwater affects the structure of the vessel and leads to catastrophic structural failure such as in the case of MV Derbyshire or SS El Faro [National Transportation Safety Board, 2015].

Linked to flooding is the concept of damage stability. This entails the ability of the vessel to withstand flooding phenomena based on first principles evaluation. Damage stability is heavily regulated and primarily is "baked into" the design by virtue of the arrangements, materials and equipment used in the construction of the ship. Naval architects (following relevant regulations) effectively simulate scenarios of damage and evaluate the ship's survivability in each case. Through this process, the stability characteristics of the vessel under damage are understood and quantified, which leads to an estimation of risk and potential for loss of life. Most ships must meet minimum levels of safety in regard to the potential for loss of life in such events. In short, the process nowadays is characterized by a probabilistic approach that takes into account all potential damages -no matter how improbable- but weighs their outcome according to their probability of occurring. Thus, the final result is a ship that can survive all highly possible damage conditions and only sink when the damage is exceedingly rare.

2.1.2 Historical Overview

Unfortunately but not uncharacteristically for humankind, major accidents and incidents incite and motivate organizations and countries to pursue ways to ensure that the accident in question *"never happens again"*. Of course, this is primarily an emotional response that drives the need for improvement, growing more intense in relationship with lives lost. However, accidents are also crucial data points that simultaneously evaluate and test all safety measures that have already been applied. During the accident sequence, technological and organizational measures of safety are tested and should they fail, criticized. Safety is not achieved only by technology means or training or some other approach but is an ecosystem that invariably contains:

- Ships designed with safety in mind.
- Technological means that increase the survivability of the vessel or passengers.
- Crew that is trained appropriately to react and save lives on board.
- Organizations with a safety culture embedded that actively protects against accidents and correct lapses in safety records.

Many times the complicated nature of this network of functions, responsibilities, and prerogatives is hard to evaluate and can only be made transparent during an accident.

2.1.3 Accident Analysis

2.1.3.1 TITANIC (1912)

The White Star liner RMS Titanic was built in Belfast by Harland & Wolf with "no restriction as to the cost" [Committee on Commerce United States Senate,] placed upon the builders, eventually leaving the yard at 46,328 register tonnes and as the largest, at the time, vessel by gross tonnage. The owning and operating companies both placed significant emphasis on safety and luxury over speed. The UK report [British Government, 1990] quotes on page 24 the procedures of the shipping company when first appointing masters to command. The masters received a letter that they had to sign and return. The letter stated that You are to dismiss all idea of competitive passages with other vessels and to concentrate your attention upon a cautious, prudent and ever watchful system of navigation, which shall lose time or suffer any other temporary inconvenience rather than incur the slightest risk which can be avoided. [Schröder-Hinrichs et al., 2012] Titanic and its sister vessel Olympic were either on purpose or inadvertently referred to as "unsinkable". The president of the owning company later testified that "[she was] especially constructed to float with her two largest water-tight compartments full of water". At the time of Titanic's construction, the Board of Trade did not specify specific requirements regarding damage stability or emergency response. The main tool for protecting from flooding was subdivision while there was no requirement for Titanic to carry life-saving appliances for all passengers. In fact the number of lifeboats that would have to be carried to accommodate all passengers would have used a lot of space and made the ships less stable (due to the raising of the center of gravity). The recommendations and expected emergency response by the Board of Trade was for the vessel to buy time for the present lifeboats to ferry the passengers of the foundering vessel to assist ships. The busy liner schedule of the era along with the adoption of wireless radio made the assumption of prompt response by neighboring ships a valid one. In fact the main accident scenario that was thought at the time was the collision scenario with a colliding vessel hitting any bulkhead directly being deemed the critical "worst-case" scenario (in case of the bulkhead delimiting the largest watertight compartments). This is an example of a deterministic damage scenario that is in line with the experience of the era as ships have always been involved in collisions and several vessels managed to survive after such incidents or afford the time for an evacuation. Titanic could also survive with the four foremost compartments flooded, in line with a collision scenario.

Famously Titanic hit an iceberg while at high speed $\geq 20kts$ which damaged 5-6 compartments that slowly started filling with water. It was relatively quickly figured out by the chief designer onboard that the vessel would eventually sink while the time to foundering was given as 01h30m. Actually, Titanic slipped under the waves after 2 hours and 20 minutes. The conservative time to sink given could have hastened certain actions that led to a larger number of people being lost such as the hasty filling of lifeboats. Neighbouring ships were contacted -largely in time- and steamed at full steam to the Titanic's location with the exception of the Californian who due to a breakdown in communication was delayed in understanding the Titanic's condition.

As a result of the accident around 1500 would lose their lives while around 700 passengers and crew were rescued, mostly retrieved by rescuing ships from the lifeboats. The large loss of life in conjunction with the publicity surrounding the Titanic's maiden voyage prompted various public enquiries in both UK and US. Importantly, the tragedy of the Titanic led to the 1st International Convention for the Safety of Life at Sea which produced the Text of the Convention for the Safety of Life at Sea that was signed in London, on January 20th, 1914. The convention went largely unratified and not applied due to the breakup of WW1 shortly after. The convention sets out is purpose as , "[...] en vue d'assurer la sauvegarde de la vie humaine en mer, dicter tous rglements et prendre toutes mesures propres lui faire produire son plein et entier effet. Translated: "with a view to ensuring the safety of human life at sea, to issue all regulations and to take all measures to ensure that it produces its full and complete effect. " The convention sets out which vessels it affects, navigational guidelines surrounding proper navigation of ice fields, use of radio or signals, lights on ships, communication as well as emergency procedures at sea among others. In further chapters construction regulations are given, involving the compartmentation (subdivision) of vessels, cases of fire, double bottom extents, astern power, inspections, lifeboats and others. Also, instructions on the maximum length of watertight compartments, evacuation procedures for affected compartments, openings and other items are referred to, whose existence is still present in SOLAS regulations, albeit changed drastically and expanded. [International Maritime Organization., 2009]

2.1.3.2 Herald of the Free Enterprise (1987)

On March 6, 1987, the roll-on/roll-off ferry Herald of Free Enterprise capsized shortly after leaving the Belgian port of Zeebrugge, resulting in the deaths of 193 passengers and crew members. The cause was determined to be a combination of human error and design flaws - the ferry's bow doors were left open, allowing water to flood the vehicle deck, leading to a rapid loss of stability and capsizing. Also, the bow wave was increased in height due to the shallow water effect or "squat". The accident prompted the adoption of the International Safety Management (ISM) Code by the International Maritime Organization (IMO), requiring safety management systems on ships to address human error. New regulations under the Safety of Life at Sea (SOLAS) convention were implemented, including improved monitoring of watertight doors and cargo areas, better emergency lighting, and increased stability requirements for damaged conditions. This tool is often used to evaluate new regulations or to compare proposed changes to existing standards. The accident highlighted the inherent vulnerability of roll-on/roll-off (ro-ro) ferry designs to rapid capsize if the vehicle deck is breached, leading to calls for improved damage stability. Amendments were made to SOLAS regulations to increase the number of emergency exits and revise the method for assessing a ship's stability in a damaged condition. The accident prompted a re-evaluation of the "safety culture" in the maritime industry, with a greater emphasis on safety management systems and corporate responsibility. The IMO introduced Formal Safety Assessment (FSA) as a tool to evaluate risks and assess the costs/benefits of proposed safety regulations. FSA consists of five steps: [IMO, 2013]

1. identification of hazards;
2. risk analysis;
3. risk control options;
4. cost-benefit assessment;
5. recommendations for decision-making.

FSA became the main tool with which new and existing regulations are evaluated in a holistic manner, moving away from prescriptive and reactive measures to accidents. Importantly, the FSA methodology allowed for regulatory changes to be developed in a clear and efficient manner.

2.1.3.3 Estonia (1994)

The sinking of the M/V Estonia in 1994 remains one of the deadliest maritime disasters in recent history and serves as a critical case study for ROPAX (Roll-On/Roll-Off Passenger) vessel safety, particularly regarding damage stability and watertight integrity. Like the 1987 Herald of Free Enterprise tragedy, the Estonia disaster was triggered by the loss of watertight integrity after the bow visor detached in rough seas, allowing seawater to flood the car deck [Almeida Santos Neves et al., 2011]. The car deck design in ROPAX vessels poses unique stability challenges, as its broad, open spaces allow water to shift freely, drastically reducing stability when flooded.

On the night of September 28, 1994, Estonia was en route from Tallinn, Estonia, to Stockholm, Sweden, carrying 989 passengers and crew. In heavy seas, the bow visor, designed to shield the car deck from water ingress, detached after its locks failed under the immense force of the waves. The visor was then torn off completely, and seawater rapidly flooded the car deck through the bow ramp, creating a free-surface effect that severely compromised the vessels stability.

As water continued to flood the deck, the ship developed a list, which progressively worsened until the vessel capsized and sank within approximately 30 minutes. The speed of the sinking left little time for an organized evacuation, contributing to the high loss of life, with 852 people perishing in the disaster. Only 137 people were rescued.

The Estonia disaster highlighted several regulatory and operational shortcomings:

The failure of the bow visor and subsequent flooding of the car deck underscored a critical vulnerability in the watertight design of ROPAX vessels. The open car deck design, combined with ineffective securing mechanisms, made the vessel susceptible to rapid loss of stability in case of visor failure. As the water moved unrestrainedly across the car deck, the shifting weight exacerbated the list, demonstrating the severe stability risks posed by the free-surface effect in open-deck designs. This phenomenon can destabilize a vessel quickly, as seen in both the Estonia and Herald of Free Enterprise incidents. Due to the rapid capsize, the crew and passengers had limited time to respond, and many were trapped below decks. The disaster led to increased scrutiny of the importance of clear and robust evacuation plans, as well as training to handle rapid stability loss scenarios.

The Estonia sinking prompted the International Maritime Organization (IMO) and European regulators to tighten safety regulations for ROPAX vessels. These included strengthened bow visor requirements, improved securing arrangements, and stricter stability criteria. The sinking also led to the Stockholm Agreement in 1996, which set additional damage stability requirements for ROPAX vessels operating in Northern European waters, where rough sea conditions heighten risks.

The Estonia tragedy thus exemplifies how design limitations, structural vulnerabilities, and inadequate emergency preparedness can combine to magnify risk in the event of a stability breach. This disaster, alongside similar incidents, has driven forward the development of stricter damage stability standards and more resilient designs, particularly in addressing the unique hazards posed by ROPAX vessels. For the research themes of this PhD, Estonias sinking reinforces the critical need for rigorous probabilistic safety models, real-time stability monitoring, and robust decision-support systems that can guide crew actions during rapidly evolving flooding scenarios.

2.1.3.4 F/B Express Samina (2000)

On Tuesday, September 26, 2000, the Greek passenger ferry Express Samina made contact with rocky outcrops outside Paros port. The ship suffers three raking damages of which one is below the waterline and therefore serious. The ship will founder about 50 minutes later due to a large extent to progressive flooding through open watertight doors. It was found later by divers and through testimonies that most, if not all, of the watertight doors were open at the time of the accident. A simulation of the flooding event by [Papanikolaou et al., 2003] reveals that if most of the WT doors were closed the ship would most likely have survived.

2.1.3.5 M/S Costa Concordia (2012)

The cruise ship M/S Costa Concordia made contact with rocks close to the island of Giglio on the 13th of January 2012. The report published by the Italian Ministry of Infrastructure and Transport is illustrative of events that day [Concordia, 2012]. A long and extensive penetration was sustained that extended 5 consecutive watertight compartments and ultimately led to the vessel resting on the bottom at a final heel of 80 degrees with the ship attaining large angles of heel quite rapidly after the event. In this accident, all relevant WT doors were closed at the time of the accident and there were no significant findings related to their use. However, the nature of the accident and its progress can signify the importance of WT doors. Costa Concordia had a very fast evolution of the emergency, 15 degrees of sustained heel 1 hour after the contact, because of the compartments affected. The free surfaces created in the machinery spaces were very quickly overwhelming the reserve stability of the vessel. Simultaneously, all the power-producing and distributing equipment fail very quickly, making damage control and general actions much harder. This was compounded by a faulty emergency generator.

Figure 2.1: View of WT compartment 6 and 7 from compartment 5 (from a sister vessel of Concordia [Concordia, 2012]

Figure 2.2: View of WT comp. 5 from comp. 6 from Concordias sister ship [Concordia, 2012]

2.1.3.6 HNOMS Helge Ingstad

The Norwegian frigate ”Helge Ingstad” collided with the tanker ”Sola TS’ outside the Sture terminal in Hjeltefjord. Upon impact, the Helge Ingstad suffered extensive damage, particularly along its starboard side, leading to flooding of several key compartments. This flooding resulted in the loss of stability, and despite the crews efforts to manage the situation, the ship sank. The rapid sequence of events, coupled with inadequate preparedness, exemplifies the failures in both the vessel’s design and the operational response to damage, issues directly linked to the themes explored in this research.

The design of the vessel revealed significant flaws in watertight integrity. Parts of the vessel like the quarter deck were not properly included in the watertight envelope while flooding through the hollow shafts represented a serious lapse in watertight integrity that was not known to crew and shouldn’t have been present altogether. These weaknesses contributed to the rapid escalation of the flooding, highlighting the importance of rigorous damage stability assessments during the design phase. Moreover, the lack of detailed consideration for all possible damage scenarios in the ships design exposed it to vulnerabilities as the static damage stability information could not be adapted to the real openings and flooding points.

The vessels bilge system, which is designed to pump water out of flooded compartments, was found to be ineffective in this case. The failure of the bilge system to handle the volume of water in a timely manner contributed significantly to the worsening flooding and ultimately the ship’s sinking. This issue points to the critical need for systems that can efficiently handle the dynamic nature of flooding and the rapid spread of water across a vessel, ensuring that floodwater is expelled in time to prevent loss of stability.

One of the most striking factors in the accident was the crews limited training in damage control and their inadequate knowledge of stability principles in an emergency. The officers were not fully equipped to assess the damage or execute the necessary procedures to stabilize the ship. Additionally, their understanding of the vessel's damage stability performance was incomplete, which impaired their ability to make timely and informed decisions. This incident underlines the need for advanced training programs and real-time decision support systems to guide the crew through emergency scenarios.

During the emergency, the crew did not have access to functional decision support tools to guide their response. The onboard tools, which were meant to provide real-time stability assessments and risk analysis, were either inadequate or not operational. The paper damage stability plans were of little use in a situation of discontinuous damaged areas (exacerbated by the permeable shafts). Similarly, the shore-side support team, which could have provided additional expertise and guidance, failed to offer the necessary assistance. This highlights the importance of having reliable and intuitive decision support systems that integrate with real-time ship data to help officers make informed decisions. If the officers on board had information regarding the expected time to sink they could have attempted to stabilize the situation knowing that there was ample reserve stability, buoyancy and time.

The decision to evacuate the vessel was made under pressure, and crucial mistakes were made during the process. Doors and hatches, which should have been sealed to maintain buoyancy and prevent further flooding, were left open during the evacuation. This was identified as the decisive factor leading to the ship's sinking. The failure to follow proper procedures under the pressure of the moment underscores the need for comprehensive emergency drills, clear protocols, and systems that ensure critical steps are followed in high-stress situations.

The combination of the rapid escalation of flooding and the overwhelming complexity of the situation led to a psychological burden on the crew, particularly those responsible for making critical decisions. The lack of real-time stability data and the uncertainty surrounding the ships fate likely led to poor judgment calls, such as the premature evacuation order. This underscores the importance of integrating human factors and cognitive support into damage stability systems, as crew members in high-stress environments require not only technical data but also decision-making frameworks that help reduce uncertainty and guide them through complex choices.

2.1.4 A short historical timeline of progressive flooding studies

The flooding rate plays a crucial role in modelling fluid transfer between compartments and through hull breaches. The Bernoulli equation is often applied to estimate flow rates at openings, based on the water levels in the connected compartments. However, this method assumes steady flow conditions and does not fully capture the complexities of real-world flooding dynamics.

Understanding the behaviour of flooding water is essential for advancing predictive tools. Research that explores the intrinsic (hydrodynamic) mechanisms of fluids or investigates fluid mechanics through simulation is particularly relevant in this context.

In scenarios involving cross-flooding, time-domain analysis of fluid movement is necessary to demonstrate how quickly the asymmetric load equalizes. Importantly, the actual flooding timeline may expose hazardous conditions that arise before reaching the steady state, underscoring the need for detailed time-dependent studies.

Subsequent developments in hydraulic flow modelling introduced innovative approaches that enhanced the accuracy of ship motion simulations. In 1991, Vredeveldt and Journe [Vredeveldt and Journee, 1991] advanced the field by including roll-added inertia in their hydraulic models, while holding other motion degrees quasi-stationary, to represent water flow dynamics more realistically. Further advancements came with Vermeer et al. [Vermeer et al., 1994] in 1994, who considered inertial effects of floodwater on roll, sway, and yaw, using convolution integrals to account for added mass and damping coefficients linked to radiation forcesalthough wave excitation and diffraction were not incorporated. This was later improved in [Journee et al., 1997].

Following this, Vassalos and Turan [Vassalos and Turan, 1994] refined these models in 1994 by incorporating interactions between sway, heave, and roll movements while dynamically updating trim to simulate ships under irregular wave conditions. Early models often assumed static flooding rates, but Vassalos and Turan added precision by employing Bernoullis principle for calculating flow rates through openings. The same model was further developed in [Vassalos and Letizia, 1995]. In 2000, [Vassalos et al., 2000] proposed a nonlinear, coupled 6-DOF method that included memory effects from radiation forces. [Zaraphonitis and Papanikolaou, 1997] contributed to this area by modeling floodwater as a concentrated mass at its center of gravity. This approach used the lump mass method for coupled ship and floodwater motion equations, assuming the floodwaters free surface remained level. Shallow water modelling is another technique commonly used to simulate the flooding water movement along with 3D mathematical models for the vessel motions ([Krueger and Nafouti, 2015] [Papanikolaou, 2001b]) Others focus mostly on the event of flooding itself; especially in the initial stages [Santos et al., 2002].

During the last two decades, a lot of research has been conducted in the area of damage flooding. More recently, projects such as eSAFE [Bulian et al., 2020] [Atzampos et al., 2019] [Luhmann et al., 2018] and the EU-funded H2020 project FLARE investigated the complex phenomena occurring during flooding and measures to mitigate them. For a more complete and up-to-date review of the literature, the reader may also consult [Atzampos, 2019].

More categorized literature is found in the following thematic sections.

2.1.5 Regulatory Framework

The regulatory framework governing damage stability in maritime vessels has seen substantial advancements, reflecting technological progress, evolving risk assessment methodologies, and critical insights gained from past maritime accidents. This framework is founded on the International Convention for the Safety of Life at Sea (SOLAS), which serves as the cornerstone of international maritime safety regulations, setting standards that are universally recognized and enforced.

2.1.5.1 The Shift from Deterministic to Probabilistic Damage Stability Assessment (SOLAS 2009)

The International Convention for the Safety of Life at Sea (SOLAS), administered by the International Maritime Organization (IMO), has long-established standards for maritime safety, including regulations on ship stability. Traditionally, SOLAS relied on a deterministic framework for assessing damage stability, focusing on specified prescriptive flooding scenarios. This deterministic approach, dating back to regulations such as SOLAS 1974, considered only a limited number of hypothetical damage cases, providing rigid criteria for ship survivability based on assumed worst-case breaches. While this method set an important baseline, it did not account for the wide variability in real-world damage scenarios, leading to potential gaps in safety assessment for complex or high-risk operational environments [Papanikolaou, 2007]

In response to these limitations, the 2009 amendments to SOLAS introduced a probabilistic approach to damage stability. This shift marked a significant regulatory advancement, moving from predefined scenarios to a risk-based method that calculates the likelihood and consequences of various flooding events. The probabilistic model in SOLAS 2009 utilizes "Attained Index" (A) and "Required Index" (R) values, derived from a ships characteristics and damage probability calculations. The Attained Index represents the calculated probability of surviving multiple potential damage conditions, while the Required Index is a target value reflecting minimum acceptable safety levels for specific ship categories. A ship is deemed compliant if its Attained Index meets or exceeds the Required Index, thus achieving an adaptable measure of safety tailored to each vessels unique design and operational profile [International Maritime Organization (IMO), 2009].

The probabilistic approach aligns with modern risk-based design principles, enabling naval architects to evaluate stability under a broad spectrum of potential damage conditions. This methodology acknowledges the complex nature of real-world incidents, allowing for a comprehensive safety assessment that goes beyond the rigidity of deterministic scenarios. As a result, designers and engineers can pursue innovative configurations while ensuring compliance with safety standards, supporting the evolution of ship design that enhances survivability without compromising operational flexibility.

The origins of this probabilistic framework trace back to IMO Resolution A.265(VIII) from 1973 [Inter-governamental Maritime Consultative Organization, 1973], which introduced concepts for probabilistic stability evaluation for passenger ships. Resolution A.265 was an early attempt to incorporate probability-based assessment in damage stability by proposing a survivability index and an assumed probability of damage for different compartments. While limited in scope and applied only to certain vessels, A.265 laid the groundwork for SOLAS 2009s more comprehensive framework, fostering gradual acceptance of probabilistic stability in regulatory contexts. Over time, lessons from marine accidents and advancements in computational modelling underscored the need for an adaptable approach that could accommodate different vessel types and operational environments, which SOLAS 2009 ultimately addressed [Vassalos Dracos et al., 1997].

The probabilistic method introduced by SOLAS 2009 not only strengthens ship safety standards but also integrates damage stability into the overall design process. By considering the varying likelihood and impact of damage across different compartments and zones, this approach allows for a nuanced view of ship survivability. This integration of probabilistic assessment into early design stages promotes optimization for both stability and operational efficiency, aligning with the industrys shift towards holistic design philosophies that prioritize both safety and performance [Papanikolaou, 2007] [Vassalos and Mujeebahmed, 2021].

2.1.5.2 The 2020 Amendments and Increased Focus on Ro-Ro Vessel Stability

Subsequent updates to the regulatory framework have further refined damage stability requirements. Notably, the 2020 amendments to SOLAS introduced a revised Required Index (R-index) for damage stability, applying stricter stability standards for vessels carrying ro-ro cargo. This revision was implemented in response to specific vulnerabilities identified within ro-ro vessels, particularly the risk of rapid water ingress onto vehicle decks, which can significantly impact stability. By addressing these vulnerabilities, the 2020 amendments represent an important step toward enhancing the safety of ro-ro vessels, whose unique design characteristics and operational requirements necessitate tailored safety standards [International Maritime Organization (IMO), 2019].

2.1.5.3 Regional Frameworks: The Stockholm Agreement & others

While SOLAS establishes global standards, regional agreements complement the international regulatory framework by addressing localized operational challenges. A prime example is the Stockholm Agreement, which was later incorporated into EU Directive 2003/25/EC. This directive mandates additional stability requirements for ro-ro passenger ships operating in the rough waters of Northern Europe, specifically to mitigate the risk of water accumulation on vehicle decks. Such water accumulation has been linked to stability failures in ro-ro vessels, highlighting the importance of region-specific stability regulations [Vassalos, 2014].

In the United States, the US Coast Guard (USCG) enforces damage stability regulations under the Code of Federal Regulations (CFR), with additional requirements for certain types of vessels, especially those carrying passengers or operating in sensitive environments. For example, USCG regulations include specific criteria for "subdivision and damage stability" for passenger vessels, similar to SOLAS but with variations to account for local operating conditions.

2.1.5.4 Goal-Based and Performance-Based Standards

The shift from prescriptive regulations to goal-based and performance-based standards represents a broader trend within maritime regulation, with damage stability considerations now forming an integral part of the ship design process. Goal-based standards enable designers to achieve regulatory compliance through innovative solutions that meet or exceed safety levels, as opposed to adhering strictly to prescriptive criteria. For instance, the Goal Based Damage Stability (GOALDS) project, initiated by the IMO, introduced performance-based standards for passenger ships, fostering flexibility in design while maintaining rigorous safety benchmarks [Papanikolaou et al., 2013]. This integration of damage stability within the design phase encourages more holistic safety evaluations, allowing ship architects to balance survivability with other operational and economic considerations.

2.1.5.5 Operational Standards and Human Factors

Damage stability regulations extend beyond design criteria to encompass operational standards that recognize the critical role of human factors. These regulations require that ships carry up-to-date stability information for use by the master and crew, providing them with the tools necessary to make informed decisions during emergencies. Additionally, protocols for maintaining watertight integrity are strictly enforced, as watertight integrity is essential for mitigating flooding risks. This focus on human factors acknowledges that the effectiveness of damage stability regulations relies not only on design but also on the competency and decision-making abilities of the crew during emergencies.

2.1.5.6 Safe Return to Port (SRtP)

The "Safe Return to Port" (SRtP) regulation, introduced under the 2009 SOLAS amendments, addresses the need for passenger ships to retain essential systems and structures for safe navigation back to port following significant damage, such as fire or flooding. This concept is designed to enhance survivability by mandating that critical systemssuch as propulsion, steering, fire detection, and essential electrical systemsmust remain operational in the aftermath of a casualty. If these systems are compromised, alternative systems and procedures must be available to maintain functionality long enough for the ship to either reach port or, if necessary, safely evacuate all passengers and crew.

SRtP requirements apply mainly to large passenger vessels, where evacuating passengers at sea may pose significant risks. The regulation emphasizes redundancy and compartmentalization, ensuring that damage to one part of the ship does not necessarily compromise the vessel's entire operational capability. Specifically, systems are often divided into "redundant groups" located in different areas of the ship, minimizing the likelihood that a single incident affects all critical systems. For example, duplicate engine and power systems are commonly placed in separate compartments, enabling a ship to maintain propulsion and manoeuvrability even if one system is lost.

An important aspect of SRtP involves the application of "system performance standards," which define acceptable levels of system functionality under damaged conditions. This allows shipbuilders and operators to implement innovative designs and redundancy features that meet SRtP goals without adhering to prescriptive layouts or equipment configurations. By focusing on performance outcomes, SRtP aligns with goal-based standards, promoting design flexibility and supporting the industry's shift toward performance-based regulations.

Another critical component of SRtP is crew training and preparedness, as effective damage control relies heavily on personnel's ability to manage and operate redundant systems during an emergency. Regular drills and training on SRtP systems are mandated, enhancing crew competency and preparedness. This operational focus underscores the importance of human factors in implementing damage stability measures, highlighting that safety depends not only on design and equipment but also on the actions and expertise of the crew in critical situations.

Through SRtP, SOLAS has established a comprehensive framework for managing damage control on passenger vessels, aiming to increase survival rates and prevent tragedies by ensuring ships can, when feasible, return safely to port after sustaining damage. This integration of system redundancies, performance-based criteria, and operational preparedness reflects the IMOs commitment to continuous improvement in maritime safety regulations.

2.1.6 Fundamental Principles

The flooding process can be subdivided into three phases: transient, progressive and equilibrium/steady state. ([Ruponen, 2007] [Zhang et al., 2019])Following a hull breach of a ship in a seaway that leads to flooding of internal ship spaces, the ship will first undergo a transient response as a result of the inrush of floodwater, which can lead to capsizing without equilibrium being restored (transient asymmetric flooding) [Begovic et al., 2013]. Such response normally results in the ship heeling towards the oncoming

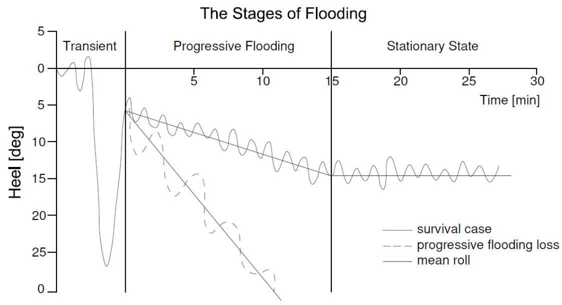

Figure 2.3: Overview of flooding stages based on diagram shown in [Ruponen, 2007]

waves but depending on the latter and the ship response, the vessel may heel towards the lee-side. This, in turn, may be beneficial depending on the size of the breach and the internal ship arrangement. As a general conclusion, there are so many parameters in large passenger ships affecting this initial response, the floodwater evolution and the ship behaviour that one cannot address these prescriptively.

Damage stability has been steadily moving from prescriptive to performance-based safety regulations [EMSA, 2015] [Vassalos Dracos et al., 1997] [Vassalos and Guarín, 2009]. Adding to that, the elucidation of certain phenomena and the mechanisms of loss from operational experience (including accidents) and improved simulation tools, the industry is now at a stage where greater certainty can be attached to the damage stability calculations done and therefore to real, practical safety on board. The phenomenon of progressive flooding is one of those matters that can be further studied.

Several damage stability & flooding reviews have been published beginning with [Papanikolaou, 2007] with multiple ITTC-led research efforts being recounted [ITTC, 2002] [Papanikolaou and Spanos, 2005] [Papanikolaou, 2001a]. Also, initial benchmark studies on different numerical codes proposed for modelling and simulating the flooding phenomenon [Papanikolaou and Spanos, 2008] [van Walree and Papanikolaou, 2007] [Spanos, 2008] are published showing a broad interest in the sector from many research and academic institutions. In many cases these "bouts" of interest are fueled by recent accidents involving damaged stability elements, usually sinking such as Estonia (1994), Samina Express (2000) or earlier Herald of the Free Enterprise (1987). Later reviews such as [Santos and Soares, 2010] examine the traditional approaches to damage stability making the distinction between deterministic techniques and probabilistic approaches clear. The transition from deterministic to probabilistic assessment methods is further elaborated in the below subchapter. In [Manderbacka et al., 2018] a selection of papers from STAB & ISSW related to damage stability are reviewed by the authors. Most recently [Rodrigues, 2024] included the latest advancements.

2.1.6.1 Large open spaces (RORO)

Large open spaces on board ships are generally characterized by the absence of primarily longitudinal and secondly transverse subdivisions usually found in Roll-on & Roll-off (passenger) vessels as exemplified in their car decks. These spaces, when flooded can amplify the effects of the floodwater's free surface and lead to a virtual loss of stability capable of capsizing the vessels minutes after the damage occurrence. In such cases, the final equilibrium position is not the proper way to ascertain stability in damaged conditions. [Santos et al., 2002] They are of special interest also under the effect of waves, causing a capsize in the presence of damage ([Kat and Otto, 1988] [Letizia, 1996] [Zaraphonitis and Papanikolaou, 1997] [Papanikolaou and Spanos, 2008]). The "water on deck" phenomenon had been identified as a crucial area for improvement quite early on, being made blatantly obvious by the tragic loss of Estonia but also SS Heraklion [Papanikolaou et al., 2014], Al Salam Boccaccio and Herald of the Free Enterprise [Department of Transport, 1987] who all featured rapid capsize due to water on deck. The Estonia disaster led to the Stockholm agreement that requested passenger ships travelling into European waters to have sufficient stability even with floodwater on the vehicle deck as well as higher freeboard. The accidents also spurred various studies into the dynamics of floodwater, seaway and passenger ship arrangements. Namely, Spanos & Papanikolaou [Spanos and Papanikolaou, 2001] used a flooded ship simulation code to examine the roll motion of the vessel, also in [Papanikolaou et al., 2000] where the mass of the floodwater is restrained on predetermined path using the lump mass concept and [Papanikolaou, 2001a]. Other researchers also used the lumped mass such as ([Acanfora et al., 2019] and [Acanfora and Cirillo, 2017]) Lumped mass simplifies the problem by describing all the floodwater as a point mass and its movement is similarly restrained across planes. [Manderbacka et al., 2011]. Other approaches also exist, for example, see [Lee et al., 2023b]. Ruponen [Ruponen, 2007] also creates a numerical time domain simulation to capture the progressive flooding on a passenger ship especially focusing on the pressure correction technique applied iteratively in a hydraulic model to model the water movement from compartment to compartment.

Initially, passenger ships were excluded from the initial implementation of the probabilistic framework known as SOLAS B1 [Sonne Ravn, 2003] which later became harmonized and applicable to all ships with updated probability factors and damage distributions.

2.1.6.2 Probabilistic assessment

The development of probabilistic assessment frameworks for damage stability in maritime vessels has been advanced by a series of European research projects, each contributing critical insights, methodologies, and regulatory foundations that enhance maritime safety. Here is an expanded overview of the listed projects:

1. **HARDER (1999-2003):** The HARDER project (Harmonization of Rules and Design Rationale) played a pioneering role in refining probabilistic damage stability standards. Building on IMO's SOLAS Chapter B1 and the probabilistic concepts from Resolution A.265, HARDER reviewed and validated damage probability fac-

tors and distributions, introducing revisions aimed at a harmonized approach for stability regulations across ship types. The findings emphasized the need for realistic risk quantification in stability assessments and set the groundwork for the probabilistic standards adopted in later SOLAS regulations. The project culminated in recommendations that led to new international standards in damage stability, which were subsequently integrated into IMO regulations.

2. **SAFEDOR (2005-2009):** SAFEDOR (Design, Operation, and Regulation for Safety) extended the probabilistic stability concepts introduced in HARDER, applying them within a broader risk-based design framework. This project sought to balance innovation in ship design with safety standards by developing tools and methodologies that integrate safety as a core design consideration. SAFEDOR produced guidelines for risk-based design and operation, including probabilistic approaches for damage stability. Its goal was to encourage flexibility in design while meeting safety requirements, which has been instrumental in enabling designers to optimize ships with increased safety through probabilistic criteria.

3. **EMSA (2009):** The European Maritime Safety Agency (EMSA) supported efforts to strengthen damage stability requirements, particularly in the wake of the Estonia disaster. EMSA's work focused on supporting the implementation of probabilistic methods through guidelines and tools that assist regulatory authorities in applying probabilistic models. This support has enabled smoother adoption and standardization of probabilistic stability assessments across European maritime operations.

4. **GOALDS (2009-2012):** The GOALDS project (Goal-Based Damage Stability) aimed to enhance damage stability for passenger ships by developing goal-based standards that prioritize survivability. GOALDS conducted extensive flooding

simulations and accident analyses, which informed probabilistic models tailored for passenger vessels. By proposing amendments to the SOLAS probabilistic regulations, GOALDS contributed to more resilient designs for large passenger ships, accommodating the complexity of damage scenarios in vessels with large, open deck spaces like ro-ro passenger ships.

5. **EMSA III (2013-2016)**: Building on prior EMSA initiatives, EMSA III focused on refining the application of probabilistic methods in damage stability and emergency response protocols. It provided additional support for implementing the probabilistic regulations across EU member states and conducted research on practical applications, including real-time damage assessment tools. EMSA III highlighted operational readiness, including crew training and decision support for emergency response, aligning with EU maritime safety policies.

6. **eSAFE (2018-2019):** The eSAFE project developed enhanced probabilistic damage stability assessment tools that incorporate new safety metrics and criteria for ro-ro passenger vessels. The project tackled specific stability issues related to vehicle decks, addressing rapid flooding risks. eSAFE introduced dynamic stability assessment tools that consider transient flooding and progressive water accumulation on decks, offering a more comprehensive approach to ship stability that accounts for operational vulnerabilities.

7. **FLARE (2019-2022):** The FLARE (Flooding Accident Response) project aimed to advance risk-based damage stability methodologies, specifically enhancing the assessment of passenger ship survivability under flooding scenarios. A key focus was refining the s-factor formulation used in SOLAS, utilizing large-scale experimental data and computational fluid dynamics (CFD) simulations to improve accuracy in predicting ship survivability. FLARE introduced real-time flooding as-

sessment techniques by integrating sensor-based monitoring and machine learning-driven predictive models, significantly enhancing decision support for emergencies. By evaluating SOLAS probabilistic models and addressing challenges related to progressive flooding, FLARE provided recommendations for regulatory updates to improve safety standards.

Together, these projects illustrate the evolution of probabilistic damage stability assessment from deterministic to dynamic risk-based frameworks. They highlight an industry-wide shift toward real-time damage response, enhanced modelling accuracy, and crew-centred decision support systems. This body of work underscores the importance of probabilistic assessment as a core tool for modern maritime safety, providing a regulatory framework that adapts to varying ship types and operational contexts.

2.1.6.3 Opening identification & size

Accurately identifying damage openings and determining their size is a crucial step in damage stability analysis, as these parameters significantly influence flooding rates and subsequent ship stability. The size, location, and shape of the openings determine the amount of water ingress and the dynamics of the flooding process. Damage openings may result from various scenarios, such as collisions, grounding, or structural failures, each presenting unique challenges for assessment.

The identification of opening characteristics typically involves input from sensors, usually water-level sensors. Such approaches have been theorized and implemented in various cases [Penttilä and Ruponen, 2010] [Ruponen et al., 2017] The geometry of an opening greatly influences flooding rates. For example, irregularly shaped or partially obstructed openings create turbulence, reducing effective flow rates, while clean, unobstructed openings allow water to ingress more freely (see section above on discharge coefficient).

2.1.6.4 Flooding risk assessment methods

Flooding risk assessment methods have evolved significantly to address the complex challenges of ship safety and stability in dynamic and emergency scenarios. These approaches increasingly rely on forensic investigations, probabilistic modelling, and quantitative analyses to identify vulnerabilities and guide design or operational improvements.

Forensic investigations, such as those highlighted in FLARE (Flooding Accident Response)'s deliverable D5.9 (Forensic Investigation of critical scenarios), play a pivotal role in understanding critical scenarios. By analyzing past incidents, reconstructing events, and extracting actionable lessons, these studies provide valuable insights into the mechanisms of flooding progression. Vassalos et al. [Vassalos and Paterson, 2021] emphasize the importance of incorporating these insights into the design of inherently safer vessels. This integration allows for the mitigation of flooding risks through advanced structural solutions and refined operational risk assessments. As forensic methods and tools have become more accessible and precise, they enable a deeper understanding of flooding phenomena. This, in turn, fosters innovation and improved measures for risk mitigation.

Probabilistic assessments and advanced numerical simulations are also reshaping flooding risk evaluation. Taimuri et al. [Taimuri et al., 2023] demonstrate the application of these techniques in predicting flooding dynamics and assessing ship stability following hard groundings. Their methodology incorporates hydrodynamic topologies, ship dynamics, operational conditions, and structural properties. This holistic approach creates a more robust framework for evaluating damage extent and survivability under challenging conditions. By considering the progression of flooding and its interactions with design features, such assessments provide critical insights into safety margins and operational strategies.

Overall, the integration of forensic and probabilistic approaches into flooding risk assessments enhances the maritime industry's ability to prevent incidents, improve survivability, and achieve the twin goals of safety and operational efficiency.

2.1.6.5 Decision Support systems

A decision support system refers to usually an informatic system or tool that is designed to be used by users with decision-making agencies. These are sophisticated, interactive and computer aided techniques for aiding the decisions [Turban, 1995]. These systems contain processes and methodologies to evaluate data and produce information that helps the decision-maker make the right choice. Usually, decision support systems are viable in applications where human decision-makers are unable to analyze all data to make an informed decision due to the:

1. vast amount of data
2. fast pace of the event
3. dynamic conditions
4. complex, stochastic uncertain events.

Also, DS systems are valuable in cases where decisions are high stakes. Decision support systems can be broken down into three main parts:

1. **Data entry interface** This is the mechanism through which data enters the system and may include filtering, sensor data or large databases. Data may be entered manually or be fed directly from sensors or other means.
2. **Analysis process** This usually entails a tool to evaluate the data according to the use of the system. The processes happening in this tool are usually too intensive to be done by the human decision-maker directly. Examples of these are mathematical calculations, simulations and others.

3. **Output to user** Lastly, the result of these calculations has to be formulated in a way that is useful to the decision-maker. In most cases, the result of the analysis generates a lot of nuanced data that is often not needed in the decision-making process. The DSS then has to summarize and refine this data to arrive at simple instructions or data to be presented to the human user. This last step is what differentiates a DSS from a generic analysis tool.

Decision support systems have found use in many sectors and especially with the advent of sophisticated analysis tools have greatly increased their presence. DSS have found use in business settings [Arnott and Pervan, 2016], healthcare [Khalfallah et al., 2023], supply chain management [Gunasekaran and Ngai, 2012], agriculture [Manos et al., 2006], environmental issues [Matthies et al., 2007] and many more. Below are some examples of decision support systems used in the naval architecture sector and especially surrounding vessel damage stability. In [Jasionowski, 2011] the author presents a decision support system that aims to help the crew on board in a flooding emergency. The author focuses on the concept of life-cycle risk formulated as a ”vulnerability log” that captures the vulnerability of the vessel to flooding, from the opening of watertight doors for example. Following that metric the crew can appreciate the capacity of the vessel to withstand damage or to improve the watertight integrity during an accident. In [Pennanen et al., 2015] the authors introduce the concept behind a decision support tool for flooding emergencies. Further in [Pennanen et al., 2015], the author demonstrates a complete methodology that also incorporates a vulnerability metric but also uses sensors to determine breach location and do time domain predictions of the survivability of the vessel. The decision support aspect of it is presented to the decision maker through the NAPA emergency computer and its functions are certainly helpful in understanding the gravity of each damage scenario. Further exploration of this approach is found in [Ruponen et al., 2019]. Similar work was carried out earlier by Varela et. al. [Varela et al., 2014] which incorporated a hydraulic model progressive flooding algorithm and a 3D representation of the vessel and floodwater. Another approach by ler & Majumder [Ölçer and Majumder, 2006] uses Cased-based reasoning (which is also used in this work) to access a database of pre-run damage stability calculations and extract the relevant information to be given to crew as a decision support function. Decision support also exists in other naval architecture sectors such as design, for example, [Ölçer et al., 2006].

2.1.7 Experimental and Computational Methods

There is a large variety of numerical hydrodynamic methods available to study the behaviour of a damaged ship in waves, ranging from Time-Domain Simulations (TDS), to CFD using RANSE and SPH solvers [Cao et al., 2019] [Cheng et al., 2017]. TDS offers low computational cost at the expense of accuracy, while CFD solvers offer accuracy at the expense of much higher computation cost [Ruth and Rognebakke, 2019]. Both are much less expensive than the model tests particularly when the number of damage scenarios increases [Lee et al., 2012]. In [Ruth and Rognebakke, 2019] it is mentioned that 30 minutes of simulation requires the use of 200 CPUs for 50 days (!). A comparison can be seen at [Niotis et al., 2019]. Especially for ships with complex internal geometries, the problem of accurately and dynamically describing the survivability is of compounded difficulty [Vassalos et al., 2016].

Furthermore, specialized tools have been proposed for crisis management during emergency flooding scenarios [Jasionowski, 2011]; [Pennanen et al., 2015]; [Varela et al., 2014]) but also as a way to dig deeper into the hidden dangers of flooding [Gao et al., 2004].

2.1.7.1 Time domain simulations

Three sub-problems contributing to the entire ship damage stability problem are identified:

1. Ship at zero forward speed moving on the free surface of the sea under the excitation of waves.

2. The behaviour of accumulated floodwater inside the ships compartments and its interaction with the ship.

3. The flooding phenomenon itself, namely the procedure of water inflow and outflow through the outer shell damage (and existing internal) openings.

Simplified computational methods, compared to CFD are the methods implemented in the time domain, typically using impulse response functions derived from linear diffraction codes in the frequency domain for the motion of the ship while the progressive flooding is modeled using hydraulic methods such as resorting to the Bernoulli equation. Calculated hydrodynamic properties in the frequency domain are transformed into the time domain by use of the impulse response function concept. By application of Newtons 2nd Law, a system of nonlinear ordinary differential equations is formulated and integrated numerically in the time domain to determine the ship motions in response to the exciting waves.

The degree to which the progressive flooding, flow at the openings, free surface effects, and ship dynamics are accurately modelled, and the extent to which all these are coupled, varies between codes. Hydraulic models based on the Bernoulli equation have been used successfully in [Santos et al., 2002] and augmented and adapted in [Papanikolaou et al., 2000] [Ruponen, 2007], to name a few indicative ones. The same is also true in terms of introducing aspects such as air entrapment, tank venting, and waves.

As regards the hydrostatic and undisturbed wave exciting (Froude-Krylov) forces/-moments acting on the ship, they are commonly taken into account by integration of the hydrostatic, incident wave pressures over the instantaneously wetted surface of the moving ship body. Damping effects that play a substantial role in the prediction of roll motion greatly depend on the viscous flow and the hull appendices moments that act on the vessel. Viscous effects that are not included in the potential theory modelling might be introduced with simplified, semi-empirical viscous models [ITTC, 2021,Petacco et al., 2024].

PROTEUS is a ship hydrodynamics and dynamics software suite, developed by the University of Strathclyde [Jasionowski, 2001] along with Brookes Bell Safety at Sea Ltd and marketed by the latter as a seakeeping and stability software with roots going back four decades. The software is capable of modelling the dynamic behaviour of intact and damaged vessels in the time-domain when exposed to wind and wave effects, in addition to performing several other functions. Over the years, the software has undergone a series of developments and versions. In addition, the software has been validated against numerous model experiments and benchmark tests on various vessel types [van Walree and Papanikolaou, 2007] [Papanikolaou and Spanos, 2007] [Spanos and Papanikolaou, 2009] [Ruponen et al., 2022]. PROTEUS follows the approach (regarding time domain formulation) explained above; deriving vessel RAOs in frequency domain and then using impulse response functions to decompose any incoming compound waves to regular waves that can be all integrated in terms of vessel motions by calculating the Froude-Krylov and restoring forces. Also PROTEUS simulates internal flooding using a simple hydraulic model while the code is inherently using the strip theory for computing the geometry entered. Viscous effects are finally added empirically.

In fact many centers and academic institutions have developed codes to simulate ship motions and progressive flooding using various techniques and focusing on different aspects.

Other numerical codes are:

1. wDamstab by China Ship Scientific Research Center CSSRC
2. SMTP by Korea Research Institute of Ships & Ocean Engineering (ROK)
3. Fredyn by MARIN
4. NAPA by NAPA
5. LDAE by University of Trieste
6. E4 Flooding by University of Applied Science(Kiel)

These codes adapt different ways of:

1. Treating the floodwater, especially regarding to the free surface(horizontal or momentarily inclined surface).
2. Vessel motions (DOFs, techniques etc.)
3. Viscous effects like damping, particularly roll damping
4. Forward speed
5. Seaway modeling(calm vs waves) including forces (radiating, diffracting etc)
6. Hydraulic model formulation
7. Geometry handling (strip or panel theory, or mesh)

More information on those techniques and numerical methods can be found in ITTC Quality System Manual Recommended Procedures and Guidelines [ITTC, 2021].

2.1.7.2 Sloshing effects & CFD

In the presence of flooding water, the motions of the vessel are strongly coupled with the movement of the water, leading to a highly complicated non-linear set of forces and vessel responses [Gao et al., 2010]. The formulation and solution of the coupled ship-floodwater motion problem were elaborated by Spanos[thesis] et al. (1997) and [Papanikolaou et al., 2000] and the effects of sloshing investigated as in [Gao and Vassalos, 2015]. In order to simulate this behaviour, numerical tests or model tests can be undertaken [Lee et al., 2012].

Several methods have been developed to model flooding in damaged ships, including field methods, such as computational fluid dynamics (CFD), and simplified methods, such as the Bernoulli equation. Field methods, like unsteady Reynolds, averaged Navier-Stokes (URANS), can accurately predict the transient flooding phase but are typically not appropriate for practical tools due to their complexity and computational requirements. Simplified methods, on the other hand, can be used to calculate the flow at openings between compartments and are often implemented in time-domain procedures. Nowadays, it is widely acknowledged that CFD methods are the only viable approach to deal accurately with the coupled motion of a ship and flooding water since the numerical simulation is more advantageous to tackle this complex problem efficiently and economically than the model tests [Lee et al., 2012]. Especially for ships with complex internal geometries, the problem of accurately and dynamically describing the survivability is of compounded difficulty [Vassalos, 2015]. Some approaches focus on the lost stability by the effect of water present inside the vessel (sloshing) [Gao and Vassalos, 2015] [Zaraphonitis and Papanikolaou, 1997].

2.1.7.3 Hydraulic Model & Air Compressibility

Hydraulic models are central to damage stability analysis, simulating water ingress and predicting ship behaviour under progressive flooding. These models replicate the intricate dynamics of water flow in damaged areas, which is crucial in assessing how effectively a ships design and compartmentalization respond during real-time flooding scenarios. Bernoulli's equation, a fundamental principle in fluid dynamics, serves as the basis for hydraulic modelling. In these scenarios, it determines flooding rates by relating the speed, pressure, and energy of the water flow through hull breaches. The accuracy of these predictions is often refined using a discharge coefficient, traditionally fixed at approximately 0.6 to account for flow losses due to turbulence and friction.

$$\frac{p_B - p_A}{\rho} + \frac{1}{2}\left(u_B^2 - u_A^2\right) + g(z_B - z_A) + \frac{1}{2}k_L u_B^2 = 0$$

where p is the air pressure, u is the flow velocity and k_L is the openings pressure loss coefficient.

Other commonly used equations are the mass balance and momentum equations [Braidotti and Mauro, 2020].

Various studies, usually utilizing experimental data have attempted to better determine the discharge coefficient specifically, see [Stening et al., 2011] or as part of a larger flooding validation exercise, see [Ruponen et al., 2010] [Ruponen et al., 2007] [Tian et al., 2024].

Incorporating air compressibility in hydraulic models is another important aspect [Lee et al., 2023a], particularly for unventilated or partially ventilated compartments. When water floods a compartment, it compresses any trapped air, generating a cushioning effect that can significantly alter flooding progression. Air compressibility plays a crucial role, particularly in confined or isolated compartments where pressure changes can either accelerate or slow flooding rates [Ruponen et al., 2013].

One of the critical challenges in modelling air compressibility is its nonlinear thermodynamic behaviour. When air compresses, the pressure response deviates from simple linear models, requiring advanced algorithms to capture these dynamics accurately. This complexity increases computational demands, as simulations must resolve nonlinear equations associated with compressing air. Physical testing also struggles with this factor due to scaling differences; air volume and pressure relationships in model-scale tests do not always translate perfectly to full-scale scenarios, introducing uncertainties in model predictions [Taveira Pinto, 2020] [Benreguig and Murphy, 2019].

Research on multi-compartment flooding, particularly [Ruponen et al., 2013], emphasizes that trapped air pressures can greatly impact the outcome, with air acting as a temporary barrier that might prevent water from entering certain compartments. Alternatively, in some situations, compressed air can drive water into adjacent areas, altering the flooding sequence and ultimately the ships stability. Modelling these dynamics is particularly challenging in multi-compartment scenarios, where the interaction of compressed air, structural gaps, and varying water levels affects not just individual spaces but the vessels overall stability.

To address the computational complexity of modelling these factors, advancements like the Pressure-Correction Method (PCM) have been developed. PCM streamlines calculations by approximating pressure and volume changes without adding substantial computational overhead, enhancing the model's efficiency in real-time applications [Ruponen, 2006]. Future improvements in hydraulic modelling might incorporate adaptive meshing and hybrid simulation methods, which can dynamically adjust the model's detail according to the flooding scenario's progress, improving both accuracy and speed.

2.1.8 Uncertainty

Uncertainty plays a pivotal role in the risk assessment and survivability analysis of damaged ships, as it influences the reliability and accuracy of predictive models and decision-making frameworks. The sources of uncertainty are diverse, ranging from variability in environmental conditions to inaccuracies in model parameters and inherent randomness in flooding events. The sources of uncertainty can broadly be categorized into *aleatory* and *epistemic* types. Aleatory uncertainty, inherent in the stochastic nature of flooding events, includes unpredictable factors such as the location and size of damage, sea state variability, and the interaction between ship motion and floodwater. Epistemic uncertainty arises from limitations in knowledge, including inaccuracies in ship geometry modelling, hydrodynamic assumptions, and material properties [Karolius et al., 2018, Chen, 2013]. For example, variations in discharge coefficients used in hydraulic models or errors in compartment permeability estimates can significantly alter flooding predictions. Also, the seaway is inherently stochastic requiring various runs to control for outliers. Probabilistic frameworks are widely used to address these uncertainties by incorporating statistical distributions and Monte Carlo simulations into survivability models. These methods evaluate a range of possible outcomes, providing insights into the likelihood of extreme events and the effectiveness of risk mitigation strategies [Chen, 2012, Karolius et al., 2018]. Incorporating probabilistic methods allows for the estimation of confidence intervals in predictions, aiding in decision-making processes. For instance, Monte Carlo simulations applied to time-domain flooding models can generate distributions of survivability metrics, such as critical heel angles or time to capsize. Uncertainty is also present in the validation of flooding and survivability models which is critical to ensure their reliability under uncertain conditions. This often involves comparing simulation outputs against experimental data or full-scale trials to quantify discrepancies and refine model parameters. Techniques such as sensitivity analysis and uncertainty propagation modelling have been developed to systematically investigate how input variability affects the outputs of flooding simulations [Chen, 2013, Braidotti et al., 2019]. For example, sensitivity studies on the influence of compartment permeability or floodwater flow dynamics can highlight key parameters contributing to uncertainty. By identifying these critical parameters, efforts can be concentrated on improving data quality or refining assumptions to enhance model accuracy. Modern survivability frameworks increasingly integrate uncertainty analysis into decision support systems for crisis management and design evaluation. These systems use probabilistic risk assessment to prioritize actions based on the likelihood of various scenarios and their associated consequences. For instance, a decision support tool could use uncertainty-aware flooding simulations to rank damage scenarios by their risk levels, guiding resource allocation during emergencies [Chen, 2012, Karolius et al., 2018].

Advances in computational methods and data assimilation are expected to further reduce epistemic uncertainty, while improved statistical methods will enhance the robustness of probabilistic analyses. Hybrid models combining deterministic and probabilistic approaches may offer a balanced solution, capturing both the detailed physics of flooding and the variability inherent in real-world scenarios.

2.1.9 Research Gaps in Decision Support for Damage Stability

The primary area where this research identifies significant opportunities for advancement is in the development of decision support tools for damage stability assessment. While numerous techniques have been proposed to estimate and mitigate flooding risks, no existing methodology effectively supports real-time decision-making regarding whether to evacuate a vessel based on an accurate probabilistic assessment of imminent sinking.

The need for such a tool may not seem immediately obvious, yet several high-profile maritime accidents demonstrate how the absence of reliable stability information contributed to poor decisions and catastrophic outcomes. A notable example is the HNOMS *Helge Ingstad* incident, where it has been argued that a more comprehensive understanding of the vessel's progressive stability loss could have prevented its sinking and reduced the risks faced by onboard personnel.

Existing approaches to damage stability assessment rely primarily on quasi-static calculations, which provide fast but oversimplified estimations, failing to capture the dynamic nature of progressive flooding and the influence of external environmental conditions. Although more advanced time-domain simulations and computational fluid dynamics (CFD) models offer higher accuracy by incorporating seaway interaction and transient effects, their computational cost renders them impractical for real-time use. Furthermore, traditional methods tend to be deterministic, failing to adequately account for uncertainty in damage extent, flooding rates, and ship behaviour, leaving decision-makers without a clear understanding of risk levels during an emergency.

To bridge these gaps, this research proposes a novel decision-support methodology that integrates machine learning (ML) and case-based reasoning (CBR) within a probabilistic framework. Unlike conventional physics-based models that require extensive computations for each scenario, ML models trained on precomputed time-domain and quasi-static simulations can generate rapid and reliable predictions, making real-time decision support feasible. By incorporating CBR, predictions are further contextualized through the retrieval of historically similar cases, ensuring that outputs remain interpretable and rooted in real-world experience.

One of the most significant contributions of this work is the explicit incorporation of uncertainty quantification in damage stability prediction. While existing methodologies provide single-point estimates, this research applies Bayesian inference, Monte Carlo simulations, and Dempster-Shafer theory to offer confidence intervals and probabilistic assessments of outcomes. This approach not only enhances the reliability of predictions but also ensures that decision-makers are aware of the level of uncertainty associated with each recommendation, allowing for more informed actions under crisis conditions.

Beyond methodological advancements, this research also aims to improve the practical applicability of decision support systems in maritime safety. Existing solutions often lack usability considerations, presenting results in a manner that is difficult for ship crews to interpret under time-sensitive conditions. By designing an intuitive interface that translates probabilistic outputs into actionable insights, this work ensures that the proposed methodology is not only theoretically robust but also operationally viable.

In summary, the integration of ML and CBR into maritime safety decision support represents an underexplored yet highly promising avenue for improving damage stability assessment. This research demonstrates how a hybrid data-driven approach can overcome the limitations of traditional methods, providing a scalable, real-time decision support tool that enhances situational awareness and facilitates safer evacuation decisions during maritime emergencies.

Chapter 3

Aim & Objectives

The primary aim of this research is to develop a novel methodology for real-time damage stability assessment and decision support. This methodology will respond to the multifaceted challenges of emergency maritime safety management by integrating advanced simulation techniques, probabilistic analysis, and the combined use of Case-Based Reasoning (CBR) and Machine Learning (ML). The goal is to generate actionable, reliable decision support for the crew under high-stakes conditions.

To achieve this aim, the following objectives are defined:

1. **Develop a Flexible, Real-Time Damage Stability Prediction tool** This objective focuses on designing and implementing a model that can provide accurate, real-time predictions of a vessel's stability in varied damage scenarios. This framework will integrate both quasi-static and time-domain simulations to address computational demands. Core stability indicatorsincluding roll, pitch, flooding progression, and Time-to-Capsize (TTC)will be embedded in the model, with a probabilistic component to capture uncertainties due to environmental and structural factors.

    - Simulation Software Integration: Quasi-static simulations will be performed using NAPA, while time-domain simulations will utilize PROTEUS. This combination enables flexible analysis, with the time-domain approach providing temporal accuracy for complex flooding progression.

2. **Establish a Comprehensive Simulation and Data Management System** The objective is to create a scalable, high-performance system to manage the extensive simulations and data necessary for accurately representing a wide array of flooding scenarios. This system will include a robust data handling solution, enabling rapid access and analysis to support real-time operational needs. The data management framework will also accommodate stochastic modelling to capture the randomness of damage scenarios, providing a more realistic foundation for predictions.

    - Data Processing: Implemented in Python, the system will leverage the pandas library for efficient data organization and flow management, allowing seamless integration with downstream predictive models.
    - Monte Carlo Simulation Framework: To assess model robustness, a Monte Carlo method will be employed by adding controlled noise to inputs, enabling uncertainty quantification through the spread of prediction outcomes. This approach will gauge how sensitive predictions are to minor input variations, enhancing confidence in the models resilience.

3. **Develop a Decision Support Interface Tailored to Crew Needs** A user-friendly interface will be designed to convey essential predictive information in a format optimized for rapid crew comprehension during emergencies. The interface will communicate uncertainty levels transparently, empowering the crew to evaluate prediction reliability and take informed action. The interface will map predictive outputs to actionable recommendations to facilitate intuitive decision-

making. The combined use of CBR and ML will improve the interpretability and precision of the support system, enhancing safety outcomes in real-world conditions.

- Input Design and Crew Interaction: Inputs will be selected and structured to be easily recognizable by the crew, with sensor data compatibility. The system will then rapidly generate predictions with corresponding uncertainty indicators. Bayesian integration with the interface will support dynamic adjustments in evacuation and response planning.

3.1 Novelty

This research introduces a unique methodology by combining Case-Based Reasoning (CBR) and Machine Learning (ML) within a probabilistic framework to enhance maritime safety through real-time decision support. This approach departs from traditional static damage stability plans and enables a more dynamic, adaptable response to unfolding incidents.

1. **Ship agnostic prediction of time to capsize** : Use of time domain simulations will improve the accuracy of the predictions (compared to quasi-static approaches). It will also allow for the dynamic motions of the vessel to be captured, used and then be predicted in an emergency such as maximum roll. The use of time domain simulations is possible due to the fact that they are pre-run and not executed in real time which would be impossible. The inherent stochastic nature of them is also addressed somewhat since the overall framework relies on multiple of them or otherwise their statistic significance. Finally, the approach does not apply to any specific ship type specifically and can be used for any vessel with a sufficient database.

2. **Novel CBR-ML Integration**: Although CBR has been used in related applications and ML is broadly applied in predictive modeling, their combined application in this context is unprecedented. CBRs interpretability pairs with MLs ability to process complex datasets, reducing uncertainty and enabling more reliable damage stability assessments. The dual approach minimizes single-model biases, improving transparency and robustness in predictions.

3. **Real-time Decision Support in emergency situations**: This tool directly supports critical processes such as damage control, mustering, and evacuation. By providing real-time probabilistic insights, the system informs the crew of incident evolution, supporting informed, safety-centered decisions. This aspect is particularly valuable given that unnecessary evacuations, especially in adverse weather, can pose increased risks to life.

4. **Integrated Uncertainty estimation**: This methodology includes a systematic approach to uncertainty management. Initial uncertainties in input data (e.g., damage extent, breach characteristics) are quantified, and subsequent predictions are treated probabilistically to account for variations in evolving scenarios. The prediction distribution offers a clear representation of certainty, while CBR further reinforces reliability by correlating with historical cases that yield similar outcomes. In cases of high divergence among similar cases, the system flags these as high-uncertainty scenarios, allowing the crew to weigh decisions with additional caution.

This research thus contributes a real-time, adaptive damage stability assessment and decision support tool that aligns with the IMOs evolving standards on probabilistic safety. This work advances theoretical understanding in maritime safety management but also provides a practical framework for operational use, capable of improving situational awareness and emergency response.

3.2 Knowledge Generation & Impact

The new knowledge afforded by this research revolves around the application of data-driven methodsspecifically, the integration of Machine Learning (ML) and Case-Based Reasoning (CBR)to address the challenge of damage stability assessment in ships. While CBR has previously been used in this domain, its combination with ML for real-time predictive modelling represents a novel advancement. This research expands the theoretical foundation of damage stability analysis by demonstrating how data-driven approaches can systematically capture the complex interactions between ship hydrodynamics, structural vulnerabilities, and environmental conditions.

A key contribution of this work is the development of a framework that allows the real-time application of **time-domain simulations**, which have historically been computationally prohibitive except in post-incident forensic investigations. By leveraging precomputed simulation databases and probabilistic modelling, this research shows that time-domain methods can be repurposed for **real-time decision support**, significantly broadening their applicability in maritime safety. This opens the door for computationally expensive techniquessuch as high-fidelity hydrodynamic modelsto be incorporated into operational decision-making, a capability that was previously impractical.

Beyond the methodological advancements, this research also contributes new insights into **uncertainty quantification in maritime decision support**. Traditional damage stability assessments provide deterministic outcomes, but this study incorporates probabilistic modelling through Monte Carlo simulations, Bayesian inference, and Dempster-Shafer theory to provide a **confidence interval for predictions**. This uncertainty-aware approach enhances the reliability of predictive models, allowing ship operators to make better-informed decisions under uncertain conditions.

3.2.1 Impact and Practical Applications

The impact of this research extends beyond academia to **practical maritime safety applications**. The development of a real-time decision support system, integrating CBR and ML, has significant potential to improve emergency response efficiency by providing dynamic, **data-driven stability assessments** tailored to evolving damage scenarios. This capability directly supports critical safety operations such as:

- **Damage Control:** Enabling ship crews to assess progressive flooding scenarios in real-time and take preemptive actions.
- **Mustering & Evacuation Planning:** Providing probabilistic guidance on the safest evacuation strategies based on predicted ship stability conditions.
- **Regulatory Compliance & Safety Standards:** Supporting maritime organizations, such as the **International Maritime Organization (IMO)**, in evolving probabilistic safety regulations by providing a framework that aligns with the latest risk-based approaches.

This research also impacts the development of **next-generation shipboard decision support tools**, which could be deployed as onboard software solutions integrated with vessel monitoring systems. By demonstrating the feasibility of real-time damage stability predictions using ML and CBR, this work lays the foundation for future advancements in **autonomous vessel safety**, where decision-making must be rapid and data-driven.

Furthermore, the scalability of the methodology ensures that it is **ship-agnostic**, meaning it can be applied across various vessel types without requiring extensive customization. The ability to generalize across different ship designs enhances its potential adoption by ship operators, classification societies, and regulatory agencies seeking to implement more **resilient, data-informed safety protocols**.

In summary, this research contributes both **new theoretical knowledge**by integrating ML, CBR, and probabilistic uncertainty modelling for damage stabilityand **practical impact**by making real-time decision support feasible in emergency maritime safety scenarios. This work not only advances computational maritime safety science but also provides a practical foundation for improving shipboard operational safety and regulatory frameworks.

3.3 Technical requirements specification

Considering this research as leading to the development of a "tool" for damage stability, the following technical requirements can be set:

1. Develop a large database of simulations (quasi-static & time domain) that is properly indexed and captures most damage stability cases the vessel may face.

2. Develop a methodology based on CBR & Machine Learning in order to create models that use the database to predict an outcome based on initial damage conditions.

3. Using random noise (applied on the initial inputs) to address the (un)certainty of the prediction and derive a confidence interval of predictions instead of a singular value.

4. Finally, wrap the prediction appropriately to be used by the onboard crew in the event of an emergency.

Except for achieving the above, further improvements to the methodology will also be presented. A demonstrative vessel and database are also created to evaluate the methodology, mainly addressing how it would work as well as demonstrating roughly the expected accuracy of the methodologies.

The above technical ”tool” component is also created by the author as a standalone Python script.

Chapter 4

Methodology

4.1 Introduction

In this chapter, two main methodologies will be presented that were used to analyse and process the data included in the aforementioned flooding database. Both these approaches make use of the stored data to inform future opinions based on the assumption that the result of a casualty that has the same initial parameters as one in the database will be the same. In fact, case-based reasoning is quite similar to machine learning. In fact, both methods are generally implementations of procedures and protocols utilizing statistics. The distinction made in this document is based on the models used in the machine learning approach, which is commonly used in ML while in the case-based reasoning approach, the methodology is developed explicitly by the author following the general guidelines.

At several parts of this methodology, several steps or items are common for both methods or equivalent. Importantly, the pre-processing and final results are similar for both methodologies.

4.2 Problem formulation

Defining a problem to be solved by a machine learning model should follow a formal process that can ensure that proper methods are used and the data and results are properly entered and extracted. It also helps to go through the process to check if the problem has been set up correctly and catch any logical errors and inconsistencies at the start.

From [Cherkassky and Mulier, 2007];

1. State the problem
2. Formulate the hypothesis
3. Design the experiment/generate the data
4. Collect the data and perform pre-processing
5. Estimate the model
6. Interpret the model/draw the conclusions

Step 1: Statement of the Problem

The initial step in any data modelling endeavour involves articulating a clear problem statement. This is a common first step in multiple disciplines, including optimization. The problem statement usually comes as a direct consequence of a stated objective or after the identification of a gap. Often, data modelling studies are situated within specific application domains, necessitating domain-specific knowledge and expertise to formulate a meaningful problem statement. Unfortunately, recent studies have tended to emphasize the application of learning methods, such as neural networks, at the expense of explicitly defining the problem. This oversight can lead to ambiguity and hinder the effectiveness of subsequent modelling efforts. In machine learning circles, sometimes the problem statement is not based on a necessity or a need but can be more exploratory in nature. This can be more easily understood considering the pattern search and finding aspect of machine learning models. A researcher could simply attempt to seek relationships between data that is not conventionally known. Of course, the final outcome is that this relationship can be used to improve understanding of a phenomenon.

Step 2: Hypothesis Formulation

Once the problem is defined, hypothesis formulation entails specifying the underlying dependencies to be estimated from experimental data. This step involves identifying sets of input and output variables, along with a general form for the unknown dependency. Multiple hypotheses may be formulated to address different aspects of the problem, requiring collaboration between domain experts and statistical modellers to ensure accuracy and relevance. This dependency is crucial in establishing the function that the machine learning model will have to emulate. For example, this dependency between input and output could be explained by a set of physics laws. In many cases, the exact relationship between input and output may already be known, and in that case, the machine learning model is expected to find and emulate that (mathematical) function. In other cases, the relationship is more convoluted but demonstrably exists. If a relationship between the input and output is not proven to exist, then it is possible that a machine learning model will fail to accurately predict the target variables given the input. Lastly, as this dependency approaches a clear mathematical formula then it is expected to be easier to train a model on it, while for more chaotic, noisy or stochastic systems, a direct relationship might be harder to uncover and more stochastic methods, such as neural networks, may need to be used.

Step 3: Data Generation/Experiment Design

The process of data generation or experiment design involves determining how data are collected. Two distinct scenarios may arise: designed experiments, where the modeller controls the data generation process, and observational settings, where data are obtained from existing sources without intervention. Most applications utilize data that may have been collected for unrelated reasons. Understanding the underlying sampling distribution is crucial for modelling and interpreting results, as it affects both model estimation and prediction accuracy. In the case of a classification task, for example, to accurately train a model to recognize both binary classes, it is important that there is sufficient data for both classes in the database.

Step 4: Data Collection and Preprocessing

Data collection and pre-processing encompass the acquisition and preparation of data for analysis. In observational settings, data are typically sourced from existing databases. Pre-processing tasks include outlier detection/removal and feature engineering, such as encoding and selection. Outliers, aberrant data points that deviate significantly from the majority, must be addressed to prevent their adverse impact on model performance. Additionally, pre-processing steps like variable scaling and feature selection optimize data representation for subsequent modelling. This step is essential to convert the physical data into a format that is better utilized within the model formulation. modelling **Step 5: Model Estimation**

Model estimation involves constructing predictive models to accurately capture the dependencies identified in earlier steps. Various modelling techniques, ranging from traditional statistical methods to modern machine learning algorithms, are employed to estimate these dependencies using available data. Many machine learning models are based on mathematical formulations resembling regression, while other models are more stochastic in nature, encompassing more random and flexible structures that are constructed on the go while training them. Some models might be better suited for different tasks, and they also differ in terms of computational complexity, hyperparameters and therefore time to train.

Step 6: Interpretation of the Model and Drawing Conclusions

Once models are constructed, they must be interpreted to facilitate decision-making. Interpretable models are essential for human understanding and trust, particularly in contexts where complex "blackbox" models are impractical. Balancing predictive accuracy with interpretability is a common challenge, with traditional statistical methods favouring structured parametric models for their simplicity and transparency. In contrast, modern approaches prioritiZe prediction accuracy and treat interpretation as a separate task, leveraging the flexibility of advanced modelling techniques. Most models are now mostly interpreted through the proxies of their performance metrics or test cases with different databases and compared with other iterations. This is especially true for the stochastic models, such as neural networks or decision trees.

4.3 Database Creation and Pre-processing

Both methods require a well-crafted database exhibiting features such as indexing, headers and labelling of every row as a unique entity, the case. The case base is created by combining various data sheets, applying rules and transformations and packaging all data in an efficient and accessible format.

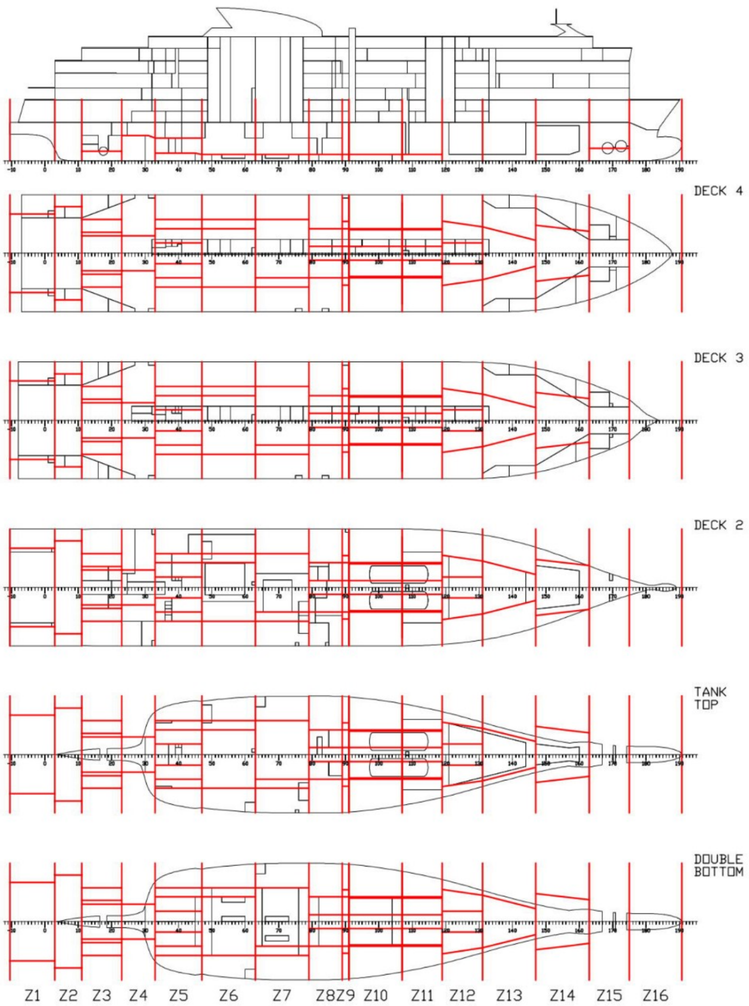

Figure 4.1: FLARE no.6 ship subdivision

The database used here consists of a range of sample damage cases prepared within the project FLARE for a ROPAX vessel of 28500 GT and 2000 passengers. Moreover, this vessel (FLARE Ship no 6) is compliant with SOLAS 2020 and uses LNG as fuel.

Charasteristic	**Value**
Gross Tonnage	28,500
DWT	3800t
Service Speed	17 kt
Car lane meters	1050
Trailer lane meters	800
LOA	162 m
Beam	28.0
Subdivision draught	6.30 m
Bulkhead deck	9.2 m
Attained Index	0.8892

Table 4.1: Characteristics of FLARE Ship no 6

The process begins when the data stored in Excel sheets is accessed and the static and numerical simulations are merged. For the testing of the above methodologies, a small database of 2518 unique cases is used. Some $\approx$ 250 cases are also evaluated in the time domain using the software PROTEUS. The merge of these two sources creates a sparse database. A case ID is given to each case, and consistent feature headers are created and maintained throughout.

The sorting, merging and processing operations are carried out in a Python script heavily utilizing the Pandas library methods.

At this stage, other than administration and organization processes, several operations are carried out on the data to align it more with the problem case. These operations include capping values to max values, as in the case of capsizing data stored is physically absurd. For example, the max roll angle is capped at 60 degrees.

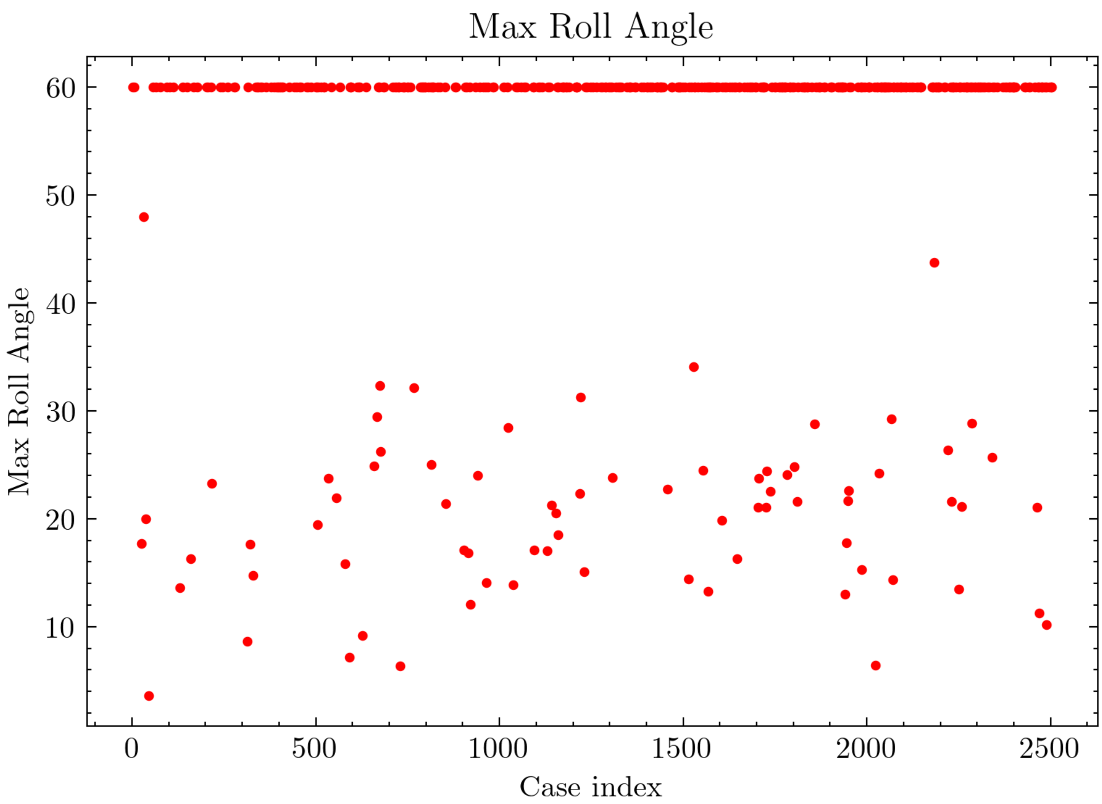

Figure 4.2: Max roll angle (numerical simulations) after capping

4.3.1 Feature rationalization & engineering

Feature engineering is a common pre-processing step of an ML pipeline where the feature variables are transformed in various ways with the expectation that the resultant features are better used and understood in the context of a model. This sometimes entails reducing the "complexity" of the given data or aggregating many different features into one. It is recognized as an area that can greatly influence the quality of the results of an ML project.

Along those lines in the following pre-processing step, a copy of the input variables is prepared in a way that:

1. Improves the performance of the ML models
2. Enables better training results
3. Can be expected to be provided in the projected use case.

One way this is done is by binning. Quantization or binning aims to reduce the nature of a variable from continuous to binary or discrete bins. For example, the location of the breach is transformed to 10 bins, each covering a tenth of the length of the vessel. So, instead of the feature vector containing the exact location of the damage, it now contains the general location. Physically, this means that damages that are close together are presented as having occurred in the same space. Of course, this is not always accurate and introduces errors. However, this is still useful to do for the following reasons. As with any such feature engineering process, this process reduces the complexity of the data and can promote better model accuracy. Secondly, it aligns with the expected use case. If the model was trained on the raw location, then it would expect the exact coordinate of the damage to work in a real-time environment.

As is easily discerned, several continuous variables, namely: Location, Penetration, Z centre and Z height, are adapted to distinct values based on a scale of numbers. This, in effect, clusters all different values into distinct bins or groups. This is done to emulate the fact that in the real case, we will never know exactly the value of those parameters. In fact, breach estimation is an area of research due to this fact. It is deemed more likely however, that a member of the crew could piece together a general idea of the damage through various sources of information such as sensors, cameras, known obstacles (in navigation, e.g., sailing close to land or in shallow water) and experience. The

Table 4.2: Main feature variables used after pre-processing

Feature	Adapted range	Comments
Length	As is	
Location	-10 - +10	1 to 10 according to location on vessel with 5 being midships, 1 stern and 10 bow. Positive for starboard and negative for port
Penetration	1-3	Incremental increases in penetration up to a maximum of 3 which corresponds to B/2 penetration.
Z centre (height of damage centroid)	1-4	Distributed evenly across the depth of the ship
Z height (height of damage opening)	1-5	According to limits (0, 2, 4, 6, 8)
Rooms	N/A	Names of rooms known to be open to the sea in the simulation.
Motions		
Heel (static & time-domain)	0-30	Capped to prevent erroneous data.
Roll	0-60	Capped at 60 to prevent large numbers due to capsizing.
Draft	As is	
Trim	As is	
Performance		
Survival factor	0, 1	Adapted to reduce the problem to classification.Threshold used ranges from 0.7 to 0.9.
GZMAXR	As is	Max GZ from equilibrium to downflooding angle.
Range effective	As is	Effective range of positive stability from equilibrium to downflooding angle.

quantification, then, on a scale is much easier and avoids errors by simply requiring approximately how bad the damage is. For example, if a ship receives a head-on collision at midships with a large, fast-moving containership then it is safe to give a high penetration number (owing to the high kinetic energy and bulbous bow), low Z centre (due to bulbous bow) and it is visually clear what a tenth of the ship is affected and side. The length of the damage is left as a continuous variable as it is generally easier to guess, and a range would not be as intuitive. The above process, while it reduces the accuracy of the models as there is no distinction between close cases, is necessary for this approach to be able to be directly applicable in a scenario where information is not exact.

4.4 Case-based reasoning theoretical background

Case-based reasoning, or CBR, is a problem-solving approach that relies on relevant past cases to find solutions to emerging situations. In this process, similar situations, which happened in the past, are searched for, and the experience gained in those situations is used. The problems and their solutions are represented by cases, and these cases are generally stored in a dynamic database (or case base).

From [Kolodner, 1992]:

1. Case-based reasoning allows a reasoner to propose solutions in domains that he/she/it does not understand completely.
2. Case-based reasoning gives a reasoner a means of evaluating solutions when no algorithmic method is available for evaluation.
3. Cases are particularly useful for use in interpreting open-ended and ill-defined concepts.
4. Remembering previous experiences is particularly useful in warning of the potential for problems that have occurred in the past, alerting a reasoner to take action to avoid repeating past mistakes.

5. Cases help a reasoner to focus their reasoning on important parts of a problem by pointing out what features of a problem are the important ones.

Case-based reasoning revolves around the idea of experience. Experience in this sense is stored in cases, hence the term case-based. This experience, if devolved or deconstructed, is simply a combination of inputs and outputs; how a system responds to specific inputs. Using this black box approach, many problems can be solved based on past experience.

CBR is based on 4 basic assumptions [Kolodner, 1996] :

1. Regularity: the same actions executed under the same conditions will tend to have the same or similar outcomes.

2. Typicality: experiences tend to repeat themselves.

3. Consistency: small changes in the situation require merely small changes in the interpretation and in the solution.

4. Adaptability: when things repeat, the differences tend to be small, and the small differences are easy to compensate for.

This is a function that humans frequently use and is the basis for many sciences, as it is directly linked with concepts such as empiricism. For example, medicine is largely based on using the symptoms (evidence/input) to discover what condition afflicts a patient and then using an appropriate set of treatment options to cure that condition based on what has been shown to work. Doctors routinely reach a diagnosis by recollecting a past case that had similar symptoms. They can then rapidly apply what worked or avoid what did not using their experiences. Just from this example, it is obvious that case-based reasoning is an important tool, especially in highly complicated cases such as medicine, where a causal relation or a transparent understanding of function is hard to attain. It is also easy to see how this process can lead to accurate conclusions, skipping otherwise necessary steps of due diligence and investigation.

Several characteristics and advantages of CBR systems are [Watson and Marir, 1994]:

1. They can be built when no model exists.

2. Can be flexible and easy to implement, considering that features need only be selected and can be further changed, added or improved.

3. The computation time for CBR systems to provide an answer is generally very fast.

4. New cases can very easily be added to a CBR to increase its performance without needing to re-engineer parts of it.

5. Due to the fundamental knowledge base being past cases, it is very straightforward to interpret or explain the operation of a CBR system. Similarly, failures and inaccuracies can be made evident, while the result can be validated if compared, for example, with similar cases identified.

The main reason why CBR is considered in the problem of damage stability is the speed of execution. The processes and tools to conduct a detailed simulation of a damaged ship in waves exist and can provide adequate answers for the task at hand. However, the time and resources needed for those to be effective are well beyond what is available for an emergency or what can practically be provided in an uncertain environment such as a nautical accident. CBR could be the answer to providing swift answers that can be constantly updated with new information to be used to predict to a large degree, the behaviour of the vessel in the future. Such information is of paramount importance for decision support. Namely, evacuating or not evacuating the vessel could mean the difference between mass casualties or no casualties. The time to capsize or sink, if known, can give rescuers and the ships crew a time window to plan operations. For example, if sinking is imminent, passengers should evacuate immediately, while if the time to sink or capsize is large then safer rescue operations could be planned such as towing the vessel to port or waiting for external assistance (using rescue boats or helicopters) to transfer the passengers to land or to another vessel. Making such decisions has been a judgment call so far, which, even though it is found on human-based CBR lacks the epistemic certainty that can be provided by analysing the ships behaviour and environment through first-principles calculations.

Furthermore, the interpretability and experience-based framework of CBR is quite easy to understand in operational terms and for companies and organizations to build and maintain. The inherent ease of confirming the result and deciding if the result should be dismissed is also a critical item for a decision support provider system.

The approach of CBR for the case of a damaged ship will be based on pre-calculated cases similar in detail to what naval architects are accustomed to in case studies, but through the process of CBR, a multitude of cases will allow for the extrapolation of a new, emerging case from the calculated ones. This will necessitate a large amount of pre-calculated cases to be run along with an appropriate organizational and procedural regime (indexing).

The structure of this approach will be presented in detail in the next sub-chapters.

4.4.1 Layout/Structure

Case-based reasoning requires a structured approach to information and reliable data. It is imperative that the knowledge stored in the cases is indexed properly, easy to recall and classified correctly according to its features. The process also requires a well-thought-out approach to selecting which cases to consider when comparing to a new one, how to deal with deviations, and lastly, how to update the solution based on a constant stream of new data. All these elements are presented below in detail.

4.4.2 Indexing and Features

Indexing or classifying each case by features that are of importance is of paramount importance, whilst also being unclear how it can be achieved. It is certain that each problem will require a custom regime of indices to outline and focus on specific features. Indexing is necessary to be able to recall previous cases (from the case base) through some identifying features. Features can be divided into surface and derivable features [Kolodner, 1992]. An example given by Kolodner [Kolodner, 1992] is that of predicting the winner based on comparative strength. One needs to focus on the ratio of attackers to defenders (derivable) rather than the actual numbers (surface). Similarly, the selection of features needs to focus on what is linked with the objective at hand. Specifically, for the task at hand, the features that should be selected need to have a direct correlation to the dynamic damage stability of the vessel. The exact relation between those features and the behaviour of the vessel does need to be epistemically known, but such a relationship needs to exist. This unknown relationship is the reason this approach is used; instead of predicting this behaviour with a first-principles calculation. The number of these features is also something that needs to be carefully decided. Ideally, one would use all relevant variables and features in creating the case database and indexing. However, we should also consider the need to have an adequately populated case base across all these features, meaning that increasing the number of features will exponentially increase the number of cases that need to be run, evaluated against, and lastly considered. This is also compounded by the fact that the allowable values for many of those features will be found on a range of numbers rather than Boolean/digital predetermined values. As an example, 5 features where each of these has 10 possible values, theoretically lead to $10\hat{5}$ possible combinations, while doubling the feature number of the same possible values will increase that number by 100,000 %. Features are also easier to constrain to smaller numbers than the range allowed for possible values, since the latter is necessary to capture adequately the usually non-linear behaviour those features have on the end response of the vessel.

Thus, the features used should be:

1. Derivable, in the sense that they represent the part that directly correlates with the phenomenon.

2. Affect the behaviour of the vessel straightforwardly and clearly. E.g. KG, Length of damage.

3. Limited in number to the most important & have a limited range if possible.

4.4.3 Retrieval of cases and Interpretation

The objective of the retrieval part of the process is to select the most relevant cases from the case base. This requires good indexing beforehand, as explained. In addition to that, the comparison between the variables needs to be evaluated properly. For the specific problem at hand, most features are expected to have a range of values such as draft, KG, and length of damage, while other features, such as size of damage, watertight door status or what rooms are damaged, are more of a dual nature. A good example of a very similar approach in ([Ölçer and Majumder, 2006]) is seen below:

	Case in the case base
Damaged rooms	R7025 R7054 R7055 R7006 R7024 R7077 R7058 R7069 R7070 R7073
Transversal location	Port
Longitudinal location	Forward
Door status	Closed
Trim	0
KG	19.2
Draught	7.45
Severity index	0.497 86
c coefficient	1

Figure 4.3: Example of a case with features from [Ölçer and Majumder, 2006]

In [Kolodner, 1992], they use a scoring system to evaluate the closeness of the cases to reality. Each feature is also scored according to weight, allowing great flexibility for users to set priorities among the features. Such an approach seems necessary as it is not straightforward what features are more important than others or how significant the percentage difference in their values. It is expected that an iterative process will produce the best weights to be used for this task. A closeness score, being the sum of the products of each feature by its weighting factor, will allow a simple ranking of cases that are the best to consider, making it easy to select a predetermined number for further processing or all cases within some limit of closeness.

4.4.4 Adaptation

Once the case selection has been completed, along with the necessary scoring, the procedure carries on with the adaptation step, where useful information is inferred from the stored case so that it can be applied to the emergent case. It is generally inevitable that the solution sought through past cases can only be approximate in nature. In the best-case scenario, the new case should have feature values that are totally encompassed by the cases already evaluated. In that case, we can postulate with some certainty that the behaviour of the vessel, or rather the maximum values characterizing it, will be within those of the stored cases. Essentially, a range of values is more appropriate to be used as the result of a median value with a known deviation. If the uncertainty is captured within this range, then the result retains its accuracy and practicality. According to the mentality of prepare for the worst, if the range of values exceeds set limits, that should trigger a response to mitigate that danger as is the case generally. Hence, the expected output from this step will be mostly a range of expected values. It is possible to link some of these values with expected behaviour, for example: GM and GZ curve; however, that will require further analysis of whether the relationship between those values is significant and can be captured without knowing other values. This approach can reduce the number of features that need to be indexed, hence reducing the number of precalculated cases required in the case base. It is generally true that many variables in damaged condition are interconnected; it remains to be seen if a simple correlation can be used.

4.4.5 Evaluation and Repair

Lastly, since this approach focuses on predicting an emergency case rather than building a database, as is the case in many CBR applications, the last two steps (Adaptation & Evaluation) should have an iterative nature. Instead of building a database, the process is required to constantly update the prediction, utilizing new data fed into the system. At the beginning of an emergency, most features will be unknown, but as time passes, hopefully, the case(s) closest will start to become more and more apparent, especially in cases where the condition starts to stabilize. The new information that is fed back can be used to eliminate possible cases from a selection of close cases or reinforce the belief in the matching of others. It is also possible that the new data will invalidate all predictions, in which case a new one should be made, or the system should alert the user that the emergency case appears to be found beyond the scope of the calculated ones, hence the result is inaccurate. The above addresses the concept of evaluating the fit, but also repair. Repair in this sense is the effort of the system to choose the best case and improve its original guess through new information. This process can also be augmented by individual simple calculations, such as a survival factor calculation that can provide extra information beyond that stored in the features of the cases. Similarly, to the adaptation step, this process needs to be straightforward, and the relationship needs to be represented accurately in a mathematical manner.

4.4.6 Advantages, Disadvantages & Considerations

The main advantages of this approach have been mentioned or inferred already. Mainly, and especially for the task at hand, CBR systems can be created without understanding or solving the first principles expressions that describe the underlying relationship between the input and output (characterized by the feature variables of the case base). In addition, the speed of execution is very fast, allowing this technique to be available in time-critical situations. Lastly, the reasoning operation is highly transparent and explainable.

CBR, however, is only as efficient and effective as the case base. Since there is no inherent understanding of the problem, the methodology can only be safely used in a space defined clearly by similar cases. In other words, the abilities of the methodologies to extrapolate are lacking compared to those of interpolation. Hence, this may require an extensive database if a sufficiently large amount of feature variable permutations are present.

Lastly, recalling the 4 assumptions inherent to CBR according to Kolodner presented earlier, when the underlying event or problem deviates from those qualities, the reasoning loses fidelity. For example, when the underlying event exhibits non-linear, dynamic or largely chaotic or noisy properties, any space left between previously evaluated cases is not accurately represented by assuming an interpolation of closest values. If, for example, the same inputs exhibit different responses in different cases or if some influencing factor is not captured in the feature variables, then the methodology will obviously suffer. It is conceivable, however, that increasing the case number and hence density and resolution will alleviate these problems to an extent through the application of the central limit theorem and by cumulatively reducing the "distance" between a case in the case base and the given one that needs to be "interpolated".

4.5 CBR implementation in this study

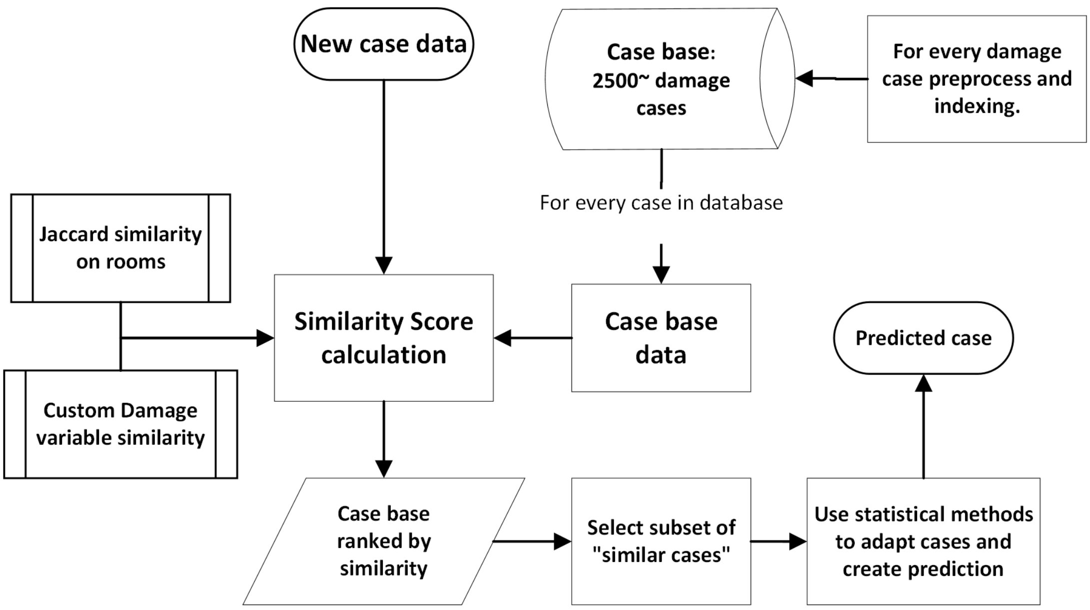

Figure 4.4: CBR flowchart as implemented

4.5.1 Introduction

Case-based reasoning, or CBR, is a problem-solving approach that relies on relevant past cases to find solutions to emerging situations. In this process, similar situations, which happened in the past, are searched for, and the experience gained in those situations is used. The problems and their solutions are represented by cases, and these cases are generally stored in a dynamic database (or case base). The theoretical background and layout are introduced above in 4.4.

4.5.2 Similarity functionality

The similarity calculation is a process done on pairs of emergent case data/database case data for the purpose of determining which cases from the database are closest to the emergent one. The similarity score is broken down into two individual scores:

1. Rooms similarity
2. Damage similarity

Each of the individual scores is added to create the similarity aggregate.

$$Similarity_{agg} = w_{1room} + w_{2damage} \tag{4.1}$$

where w_1,w_2 are weighting factors. The similarity scores for the rooms are based on Jaccard similarity. Jaccard similarity measures the similarity between two sets of data and is often used in that function.

$$J(A, B) = \frac{\mid A \bigcap B \mid}{\mid A \bigcup B \mid} \tag{4.2}$$

In the case of the room similarity, it is immediately obvious how Jaccard similarity is applied.

In the case of damage, a custom metric is used that works as follows. For each different damage variable, there is a different function that gives a closeness metric ranging from 1 for total similarity to 0 for complete dissimilarity. The function is different each time as the physical meaning of the damage variables is different. Each variable also has a distinct weight attached to it:

$$Similarity_{damage} = \sum_{i=0}^{n} Sm_{var_i} \cdot w_i \tag{4.3}$$

Where n is the number of damage variables. See below the length similarity function. Several points are chosen arbitrarily, and a function is fit over them. The points are selected based on intuition for now. For example, a length percentage difference of 10 % is evaluated as a similarity of 0.8 while at 60 % it is 0.

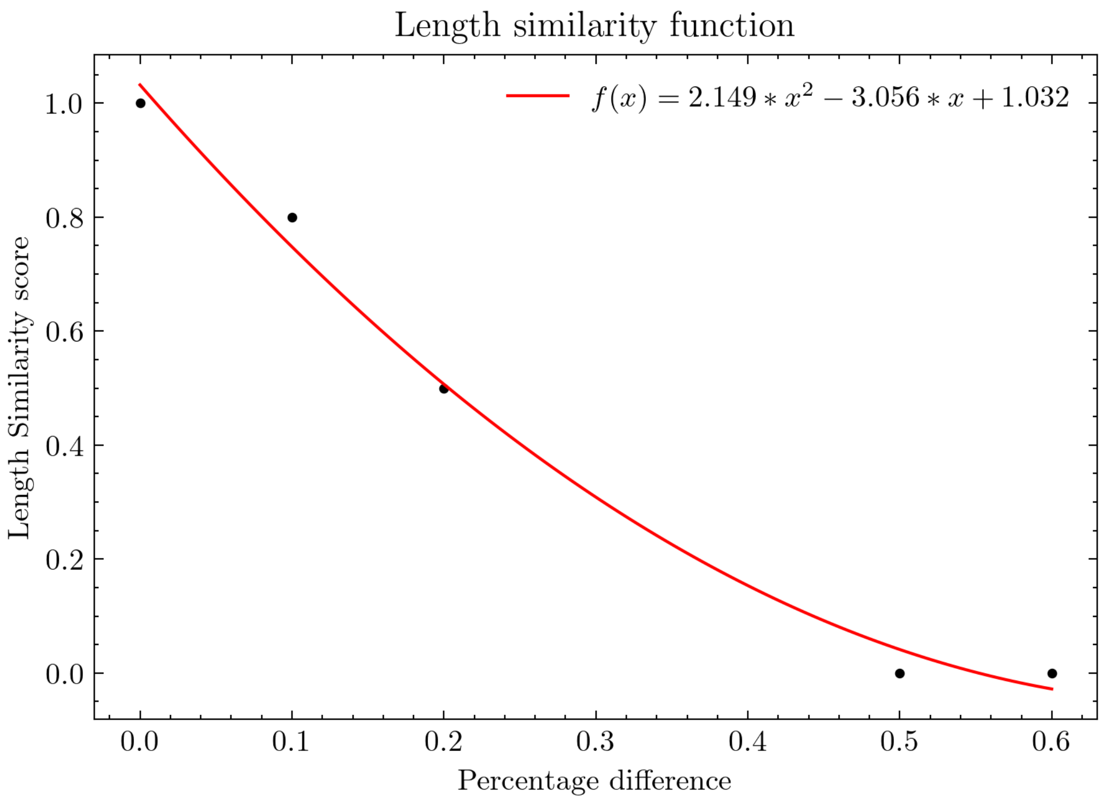

Figure 4.5: Length similarity score as a 2nd order polynomial.

In cases where the function ceases to follow logic, limits are set internally. For example, for length % difference values of over 60%, the result is automatically zero. For variables where the values are integers, the difference is just the integer distance. For the location, if values do not have the same sign, the result is total dissimilarity (opposite side of damage).

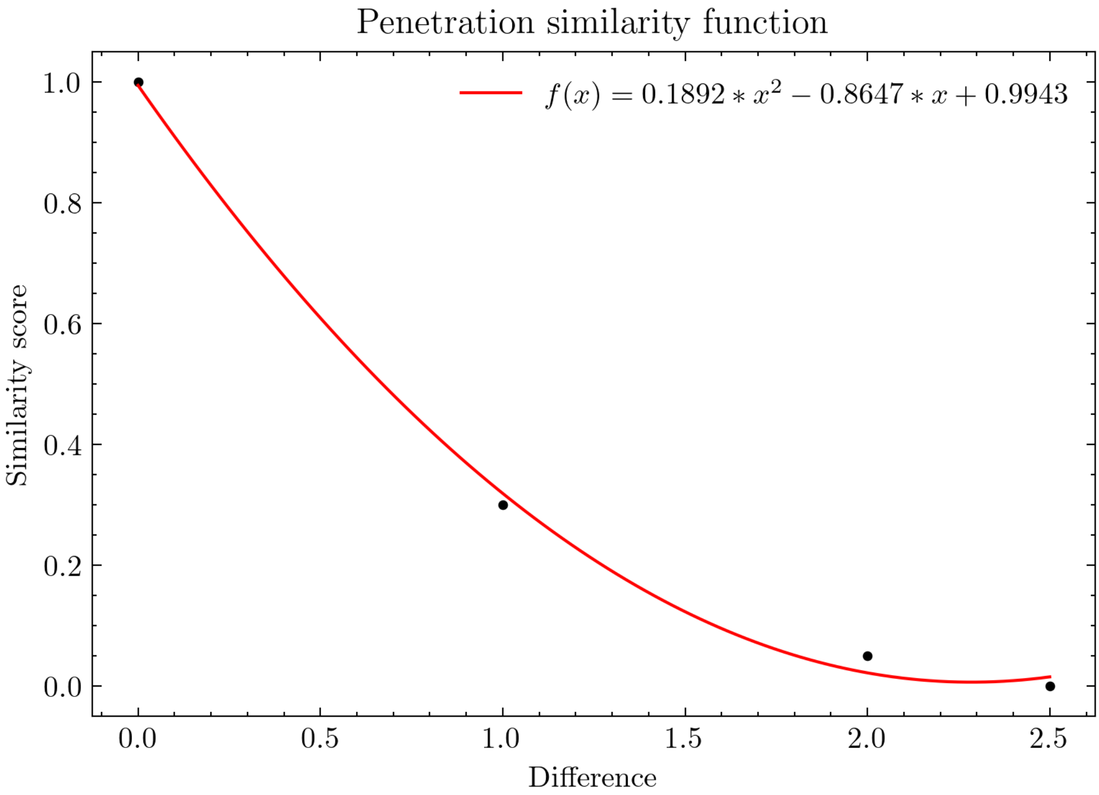

Figure 4.6: Penetration similarity as a polynomial

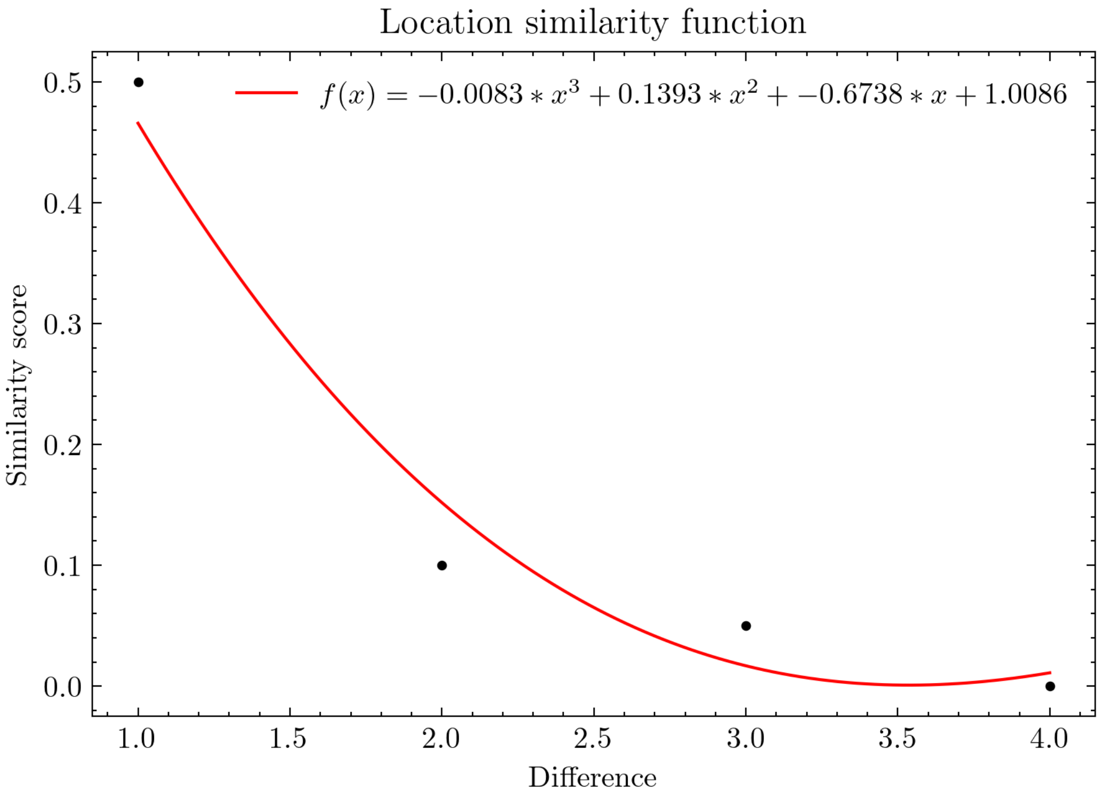

Figure 4.7: Location similarity as a 3rd order polynomial

The weights used, w_i, are needed since some damage variables are more important than others in determining the outcome. Traditionally, length and location have been used for the assessment of damage stability (e.g. SOLAS). Through a process that will be introduced later in the Machine learning part of the methodology, the importance of these features or damage variables can be calculated. Feature importance is a process that makes use of a model that is trained on the data. The input features are then changed, and the result on the outcome is stored. Through this process, in the end, the importance of each feature can be calculated, usually compared to others. This is a very useful technique that provides insight into the data and how each variable affects the final result. It can point to the most important features or to features that are superfluous and can be dropped. Below is the feature importance for a Random Forest classifier fit on the data for predicting the survival factor (SFAC).

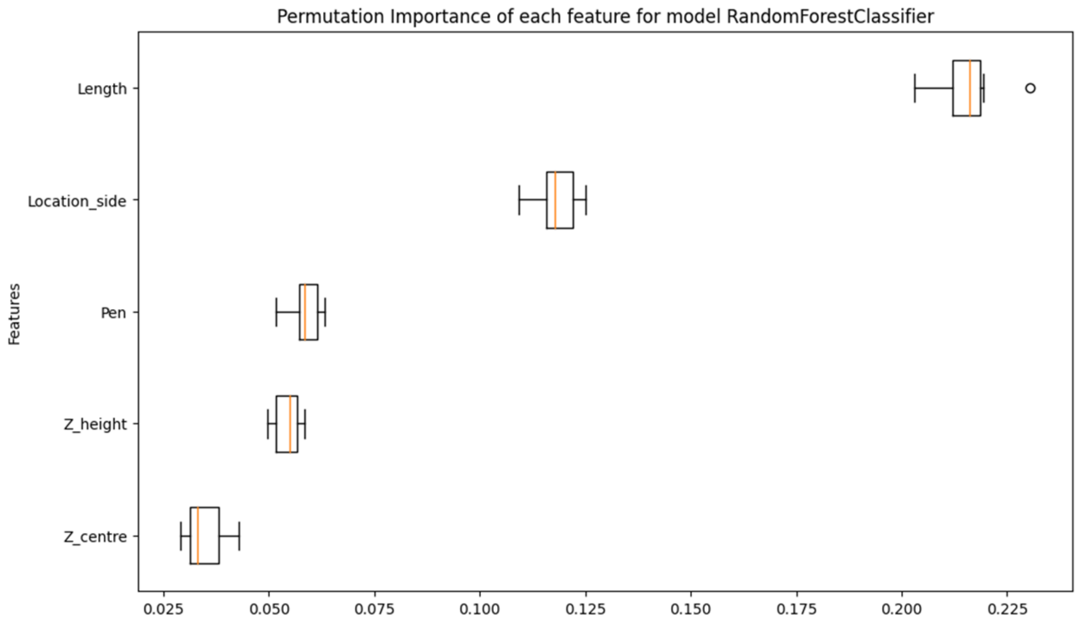

Figure 4.8: Permutation feature importance for predicting SFAC

As can be seen, length and location are very important in predicting the outcome. The weighting factors are determined accordingly.

Table 4.3: Weight factors for similarity function

Feature	Weighting factor
Length	5
Location	3
Pen	1
Z_height	1
Z_centre	1

4.5.3 Data random case generator

At this stage, a function for generating data is also created based on the fact that the data received by the tool will be information about the damaged and affected rooms. A random element is also introduced here to create noisy data. First, a case is selected to be simulated. Then the data of the case is affected by random noise around its true values. The amount of noise is directly controlled and can be altered.

```
Case selected 22
Noise is 10% and 0% of the rooms are wrong
Most likely cases are:
22 with a similarity score of: 0.99  SFAC: 1.00
5663 with a similarity score of: 0.96  SFAC: 1.00
8643 with a similarity score of: 0.85  SFAC: 1.00
9548 with a similarity score of: 0.84  SFAC: 1.00
```

Figure 4.9: Example of case (noisy) generator output 1

```
Case selected 22
Noise is 0% and 6% of the rooms are wrong
Most likely cases are:
22 with a similarity score of: 0.98  SFAC: 1.00
5663 with a similarity score of: 0.96  SFAC: 1.00
8643 with a similarity score of: 0.84  SFAC: 1.00
9548 with a similarity score of: 0.84  SFAC: 1.00
9581 with a similarity score of: 0.84  SFAC: 1.00
```

Figure 4.10: Example of case (noisy) generator output 2

In the example above, case 22 is selected, and the random noise added is printed.

The percentage signifies the amount of noise compared to the range of the variable. The similarity score, which will be introduced later, is then computed. Case 22 is the top selection for small amounts of noise. For moderate to large amounts of noise, case 22 is ranked 3rd as the shift in data points to other cases. However, the similarity is still large, and case 22 will be considered.

```
Case selected 22
Noise is -30% and 8% of the rooms are wrong
22 with a similarity score of: 0.85  SFAC: 1.00
2662 with a similarity score of: 0.77  SFAC: 1.00
9548 with a similarity score of: 0.76  SFAC: 1.00
9898 with a similarity score of: 0.76  SFAC: 1.00
9537 with a similarity score of: 0.75  SFAC: 1.00
```

Figure 4.11: Example of case (noisy) generator output 3

4.5.4 Retrieval of cases

By using the similarity function, the cases closest to the emergent case can be found. Actually, the similarity function is applied to all cases, and each is furnished with the similarity score. The resultant database is also sorted based on descending similarity score. To reduce the computational time, only the highest-performing cases are considered. Setting an arbitrary number of cases to consider or a cutoff limit of the minimum score is not appropriate. Instead, a percentile value is better used that will shift the cutoff limit of similarity based on the distribution of the similarity scores. An example of the function can be interpreted in 4.12.

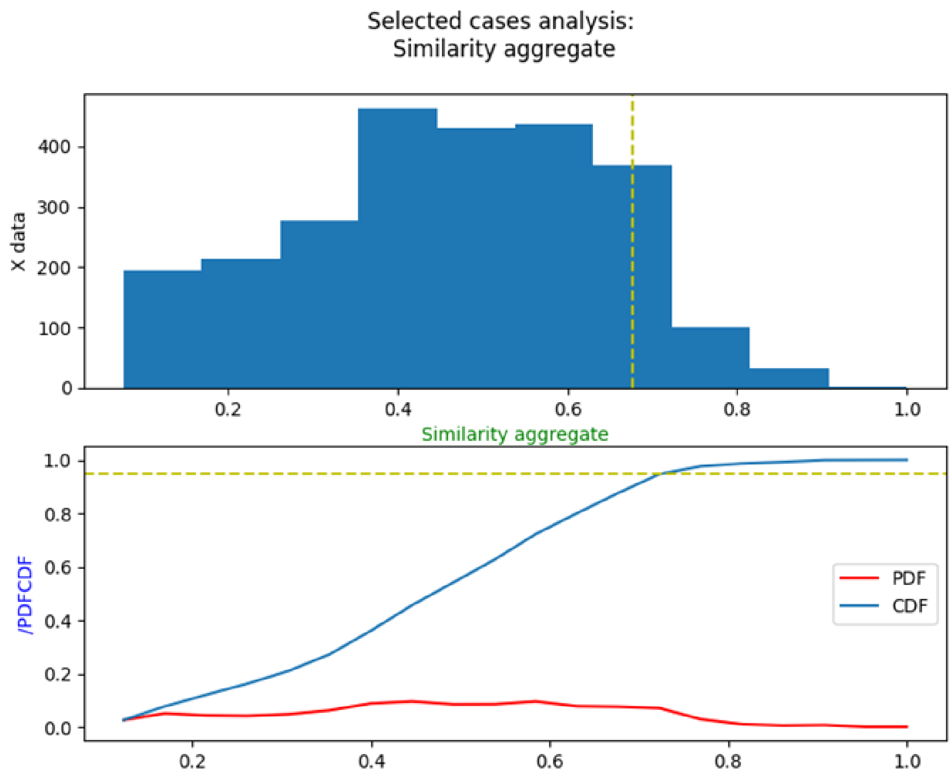

Figure 4.12: Subplot(upper): Histogram of similarity scores across database. Subplot(lower): Probability & Cumulative density function of similarity aggregates. The yellow line is the similarity value according to the percentile selected and the percentile for the lower subplot.

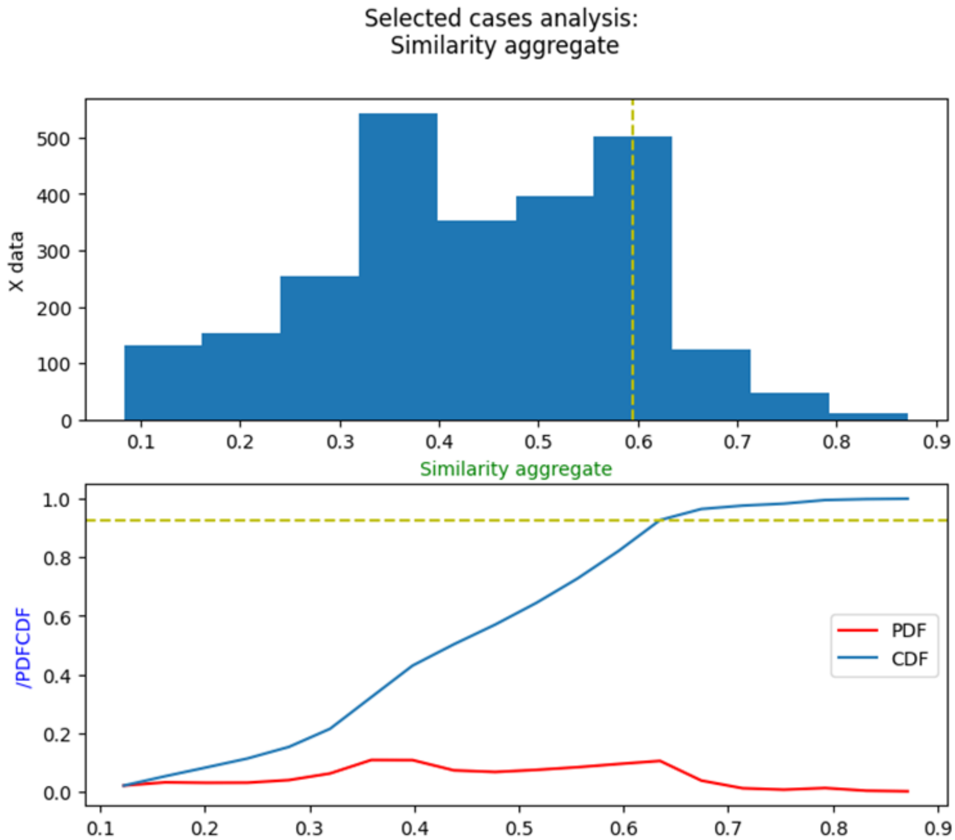

Figure 4.13: Same plot as above with different random data.

The percentile used in this case is the 92nd. It is easy to see that the selection makes sense as it keeps the highly similar cases specifically while avoiding clusters of numerous but less probable cases. In Figure 32, the larger number of high scores shifts the cut-off even higher as expected. The next step is to extract a useful value from this database of probable cases. The values extracted can be any of the values present in the database. We can also include further knowledge in the probability of predicting the correct case. The database already contains the a priori probability of each case in the form of the p factor. In short, some cases are more probable to happen as a result of the distribution of damages considered in SOLAS. Hence, this knowledge can be used to enhance the similarity factor and form a probability that the specific case is happening. To achieve that, Bayesian inference is used, utilizing Bayes rule similarly to a sensor fusion problem. Bayes rule lies at the heart of most data fusion methods. In general, it provides a means to make inferences about an object or environment of interest described by a state x, given an observation z. In this case, the state x, is which case from the database is the one evolving while the observation, z, is the similarity metric. The joint probability is obtained from Bayes rule.

$$P(\mathrm{x} \mid \mathrm{z}) = \frac{P(\mathrm{x} \mid \mathrm{z}) \cdot P(\mathrm{x})}{P(\mathrm{z})} \tag{4.4}$$

The conditional probability $P(\mathrm{x} \mid \mathrm{z})$ serves the role of a sensor model as it describes the likelihood that a specific z observation is obtained conditioned on the state of x. We have evaluated this value for each case (x) by using the sensor similarity score applied to variables of x. This term is also denoted as the likelihood function in literature. The denominator, $P(z)$, is the marginal probability that, in this case, simply serves to normalize the posterior probability $P(\mathrm{z} \mid \mathrm{x})$. It will be computed so that the sum of all conditional probabilities will sum to 1. Finally, $P(x)$, contains all prior beliefs on the state of x irrespective of observation. The a priori distribution is specifically the p factor as calculated from NAPA based on the SOLAS formulation. According to the above formulations, the probability of a case being the one really happening is given as the posterior probability $P(\mathrm{x} \mid \mathrm{z})$. Where $P(\mathrm{x} \mid \mathrm{z})$ is:

$$P(\mathrm{x} \mid \mathrm{z}) = C \cdot Similarity_{agg_i} \cdot pfac_i \tag{4.5}$$

For:

$$C = \frac{1}{\sum_{i=0}^{n} P(\mathrm{x} \mid \mathrm{z})_i} \tag{4.6}$$

where n is the number of cases in the database.

To conclude, by considering the a priori distribution of probability found in the p factor, the algorithm will boost cases that are more frequent compared to rare ones.

4.5.5 Adaptation

The result of the previous step is now a probability density function (that integrates to 1) for not only the similarity of the cases but for every value linked to the cases. For example, see below Figure 4.14 & Figure 4.15.

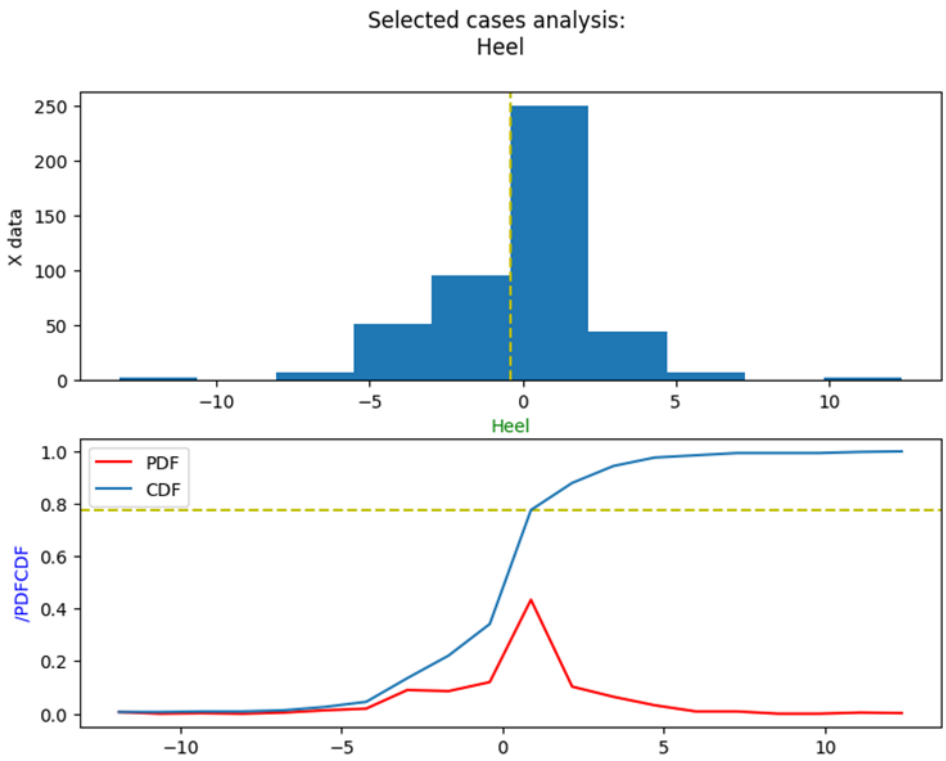

Figure 4.14: Analysis of heel data from selected cases.

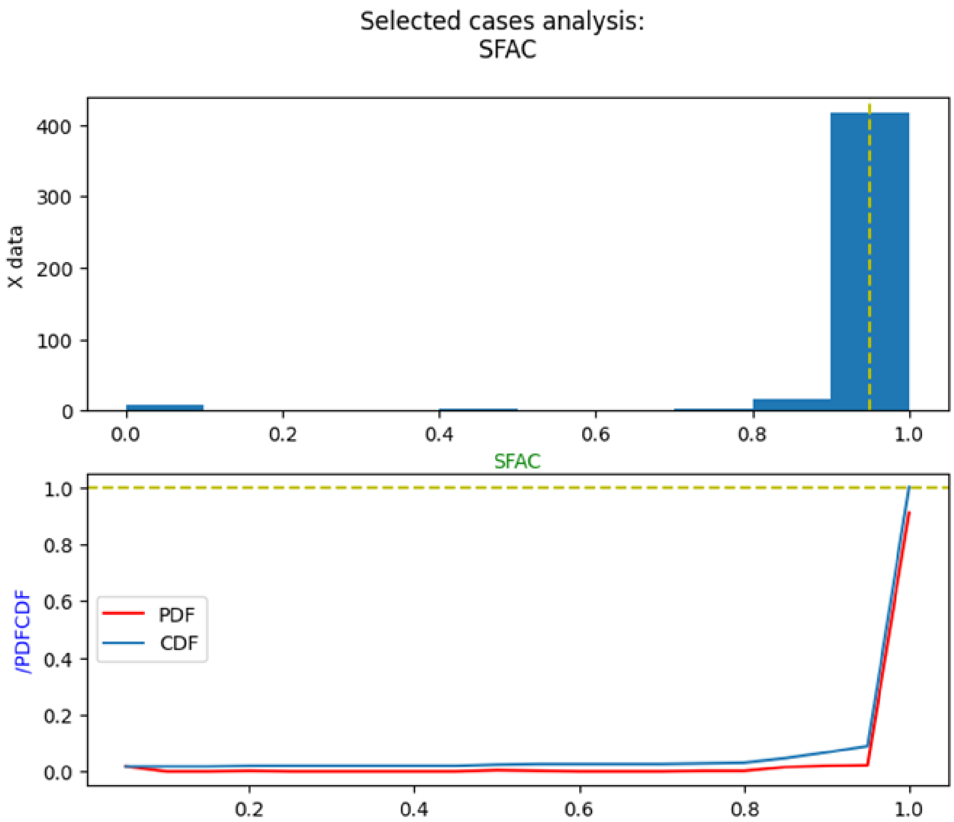

Figure 4.15: Analysis of SFAC data of selected data

The yellow line in the case of variables is defined by the mean value and the corresponding cumulative density function (CDF) of that value in the bottom graph. Note that the average extracted here assumes all the selected cases are equal in weight. Finally, to make use of the different probabilities of each possible case as calculated by the conditional probability of the a priori distribution and the likelihood function, a weighted prediction is returned, formulated as:

$$Prediction_{var} = \frac{\sum_{i=0}^{n} \cdot var_i \cdot P(\mathrm{x} \mid \mathrm{z})_i}{\sum_{i=0}^{n} \cdot P(\mathrm{x} \mid \mathrm{z})_i} \tag{4.7}$$

Where n is the number of selected cases based on 4.5.4. Using this value, more probable cases will have a larger contribution towards the prediction. Compared to the simple average, this value will be more specifically conditioned on the fact that the cases selected have high scores. The similarity aggregate can be used instead of the probability for an uninformed a priori prediction. The simple average will be less specific and more conservative. The same effect is replicated if the percentile selected is lowered or higher. A higher percentile will narrow the scope to the most similar cases, while a lower one will encapsulate more cases, thus being a more conservative and less specific prediction.

4.5.6 Evaluation

The last step of the process is to evaluate the prediction and decide on future steps. It is also important to decide how time enters into the equation of the CBR methodology. Namely, as time passes, new data is generated as a function of:

- Flooding propagation: most of the data we have revolves around the final equilibrium of the vessel, if present (sinking, capsizing) thus it will take time for the flooded rooms to be reported either through technical or human means.

- Better information, as more time is made available for crew actions, such as surveys of the damage or reports coming in, will paint a better picture of the situation. It is expected that several damage variables can be confirmed given sufficient time to survey the damage. These include the location of the vessel, penetration, and height. An estimate of the length could also be made.

Owing to the above, later predictions are expected to be more accurate. Since the data available refers only to one state (final state equilibrium) the previous predictions do not carry any implicit information forward (in time). In other words, the initial prediction will not progressively change into a new one but rather it will be constantly replaced by evaluating the most recent data. In the case where data is present that is time-dependent, the posterior evaluation can make use of the previous time step.

4.6 Machine learning approach

4.6.1 Introduction

Machine learning (ML) is a field of computer science that uses specialized algorithms, techniques, and processes for generating solutions to complicated problems that are hard to tackle with conventional programming approaches. Most importantly in machine learning the programmer does not design the program to solve a problem explicitly in predefined steps but instead sets up a framework of rules and functions that can achieve that by learning. This learning is possible given a labelled dataset that essentially contains examples of input-output pairs that the algorithm then learns to reproduce for new input data and provide correct results. This is called supervised machine learning. ML algorithms can solve many complex problems using this generic approach.

Preface: *The following section shortly introduces machine learning while focusing on elements pertinent to the present methodology and implementation. The following is in no sense a complete or perfectly accurate representation of machine learning but a useful introduction aimed at engineering applications and audience.*

> Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed. Arthur Samuel, 1959

Machine learning (ML) is a field of computer science that uses specialized algorithms, techniques, and processes for generating solutions to complicated problems that are hard to tackle with conventional programming approaches. Most importantly in machine learning the programmer does not design the program to solve a problem explicitly in predefined steps but instead sets up a framework of rules and functions that can achieve that by learning.

Machine learning, within the expansive realm of artificial intelligence (AI), represents a pivotal subset dedicated to the development of algorithms and models designed to confer upon systems the capability to learn and refine their performance through experiences. Unlike traditional programming paradigms where explicit instructions dictate a system's response to various inputs, machine learning operates on the principle of enabling machines to autonomously discern patterns and improve their decision-making processes over time. This subset within AI not only emphasizes the capacity for autonomous learning but also the ability to adapt to new information and optimize performance without explicit human intervention.

Arthur Samuel's definition of machine learning in 1959 defined a subset in computing & programming. In [Samuel, 1959] he demonstrated that machines can "learn" given only general directions and rules; in this instance the game of checkers. The author concludes prophetically:

> *As a result of these experiments one can say with some certainty that it is now possible to devise learning schemes which will greatly outperform an average person and that such learning schemes may eventually be economically feasible as applied to real-life problems.* [Samuel, 1959]

The ability of machine learning to solve problems that it is not explicitly programmed to solve, transcends the conventional approach of explicit programming for specific tasks.

Unlike traditional programming, where rules and instructions are meticulously crafted to govern a system's behavior, machine learning introduces a dynamic and adaptive framework that can be adjusted to provide results and solutions to various problems of a multitude of natures. For example, natural language, regression, pattern recognition, clustering and many more. In every case, the systems are not explicitly programmed for a predetermined task; instead, they learn from data, identifying patterns, and making informed decisions based on experience. This departure from explicit programming introduces a more flexible and responsive approach to problem-solving.

This flexibility essentially "unlocks" different possibilities and functions. For example, in the presence of a data set of seemingly disjoint variables a machine learning model might be able to figure out a pattern. This demonstrates a significant shift from traditional statistical approaches or generally how systems are interpreted and understood. As systems are better understood, the interdependencies and aleatoric and epistemic uncertainties sometimes frustrate attempts to model them with strictly first-principles tools. In many cases the underlying physical laws are well understood but the size, complexity and random factors negate their use. Similarly, some events. while explainable, are not easily modeled and instead have to rely on assumptions and simplifications. Consider the following examples:

- Modelling power consumption of a region
- Weather prediction and forecasting

Starting with the task of modelling the electricity consumption of a region over a time period-this is obviously a worthwhile task as if consumption can be predicted, the network operator can pre-allocate resources and optimize power production from cheap and renewable resources consistently. It is also clear that modelling this is not overly complex; residential, industrial and other consumers will turn on devices and power will be consumed. The physics are not complicated, however predicting the behaviour of every consumer is impossible as it is completely aleatoric. Some days some workers may stay overtime and factories might use more power, some people may leave the country in the summer months for vacations. Not all days are the same; one night there is an important football match and everyone is home watching it or the following Monday is a public holiday. All these factors point towards a multitude of periodic and transient events that are impossible to directly predict and integrate in an estimation. Statistics are helpful here. Disregarding the underlying events and blindly collecting data, a data scientist could provide helpful insights on pairs of data that can be easily interpreted. For example, public holidays have, on average 35 % less power than a normal working day or that nights with big football matches on have a 7 % decrease in power used. Along the same lines a machine learning model can be trained on the data collated. The trained model however will "find" all underlying patterns in the data without explicitly being programmed to. This is also not restricted in dimensions. Humans are not able to reason cause and correlation relationships of more than 4 priors [Halford et al., 2005] , while usually most people will struggle to correlate more than 2 variables with a value; for a machine learning model there is no inherent limit.

This is further reinforced in considering a weather prediction system. Initially, weather forecasting systems exist; all significant physical phenomena linked with weather are properly understood and can be modeled in high fidelity. Of course, there are many assumptions involved and the resolution of the answer is again limited by the chaotic nature of the phenomenon and any prediction is unstable when carried much further into the future.

In the study by [Heckscher, 2018], researchers investigated how the veery's reproductive behavior could be correlated with the timing and intensity of the hurricane season. The veery, a migratory bird that travels from North to South America annually, seems to possess an internal mechanism to adjust its reproduction in response to hurricane patternsa phenomenon that researchers were able to model statistically. They observed that in years when hurricanes are expected to be more intense or frequent, veeries produce fewer eggs and begin nesting earlier in the season.

The study found a strong correlation ($R^2 = 0.84$) between veery nesting behavior and the timing of the hurricane season, suggesting that these birds use environmental cues to make accurate predictions without understanding the atmospheric processes behind hurricanes. The high value (where 1 indicates a perfect fit) implies that 84% of the variation in the hurricane seasons timing could be predicted by changes in veery behavior, a level of accuracy comparable to machine learning models that are trained to predict outcomes based on historical patterns.

This research highlights how ML systems, like the veery, can achieve predictive accuracy through pattern recognition, leveraging correlations in data even in the absence of explicit causative knowledge. Heckschers predictive model, based on veery reproductive metrics, mirrors how ML models might identify and exploit complex data correlations to produce accurate forecasts without understanding the physics or biology underlying the data.

To conclude, machine learning and in general artificial intelligence, is very useful in places where there is no known algorithm, or in other words in discerning patterns.

Linked to that capability and as a direct result, ML can provide insights into data and help humans understand and learn themselves about the correlation of events. Applying ML techniques to dig into large amounts of data can help discover patterns that were not immediately apparent. This is called data mining. For example, even with no prior knowledge we can now say that the veery reproduction season is affected by future hurricanes. Similarly, (and as can be seen later in Chapter 4) the relative importance of events in relation to an outcome can be quantified. Data mining or knowledge discovery from data (KDD) comes as an evolution of information and data management technology [Han et al., 2011]. Most specifically, data mining enables us to extract knowledge or information from a seemingly vast and unimpressive collection of data. In the present implementation, the predictive analysis of the data is sought to extract "knowledge" about the ship's survivability. In theory, a sufficiently large and descriptive database of damage stability events (flooding incidents with all possible combinations of seaway, damage locations, loading conditions etc) can be "mined" to derive a comprehensive model that can predict the outcome of the damage scenario given the characteristics of the damage stability events.

Some examples of machine learning implementation in a data mining role include [Elgendi, 2019] where the author by analyzing a database of lowly and highly cited papers was able to identify features of the papers that are either positively or negatively related to the citations they get. In [Bose and Mahapatra, 2001] the authors outline the multiple ways ML is used in the business sector again in the form of "Data mining". The authors in [Dogan and Birant, 2021] list the uses in manufacturing [Bock et al., 2019]

Machine learning, as a subset of data-driven techniques or conflated with artificial intelligence has been seamlessly integrated already in a multitude of ways into daily life. Numerous examples of its use often go unnoticed. One prevalent manifestation is in personalized recommendations on streaming services, where machine learning algorithms analyze user preferences, viewing history, and demographic information to suggest content tailored to individual tastes. Virtual assistants go beyond rigid command-response interactions. They exhibit a nuanced understanding of natural language, thanks to machine learning models that continuously refine their language comprehension through exposure to diverse linguistic patterns. Moreover, predictive text on smartphones exemplifies machine learning's ability to anticipate user input based on context, history, and evolving language trends. Recent natural language machines such as Chat-GPT or Bard seemingly get closer to an artificial general intelligence (not being one) due to the seemingly human-like answers they provide. All of these are examples of learned behaviour on sufficiently large and appropriate data sets. Other recent, high-profile uses of ML can be found in Internet of Things, healthcare and the COVID-19 pandemic, user recommendation and e-commerce, image search and recognition among many others [Sarker, 2021].

To summarize, Machine Learning is great for:

1. Problems for which existing solutions require a lot of hand-tuning or long lists of rules: one Machine Learning algorithm can often simplify code and perform better.
2. Complex problems for which there is no good solution at all using a traditional approach: the best Machine Learning techniques can find a solution.
3. Fluctuating environments: a Machine Learning system can adapt to new data.
4. Getting insights about complex problems and large amounts of data (data mining).

4.6.2 Supervised/Unsupervised Learning

Machine Learning systems can be classified according to the amount and type of supervision they get during training [Pantic,], [Rebala et al., 2019]. There are four major categories: supervised learning, unsupervised learning, semi-supervised learning, and Reinforcement Learning.

4.6.2.1 Supervised learning

Supervised learning requires a database of inputs that are ideally explaining the outputs through some kind of relation. In that sense, a basic understanding of the problem is a prerequisite along with a well labelled dataset. The data included needs to be understood to have a relationship between them, with a subset of them being designated as "input" features or variables while one (usually) or more features being selected as the "output" or target. No matter how efficient and effective a supervised model is, the pre-processing step of selecting the variables and other steps is crucial as the model will be able to find a relationship between "inputs" and "outputs" if one such exists. The involvement of a person in the setting up of the problem makes this approach "supervised" as the algorithm is given explicit information on what variables correspond to which output or what priors describe the target feature. In some cases, this is simply pointing to the data point while in others such as in machine vision, a person might also have manually entered data and labeled them. The act of labelling generally entails the preprocessing step of describing and categorizing the data points. The learning is then possible given a labelled dataset that essentially contains examples of input-output pairs that the algorithm then learns to reproduce for new input data and provide correct results. More specifically, supervised models will alter variables that affect their (labelled) output given the inputs until the two match. Then repeat that for all data available and derive the values that best fit all data.

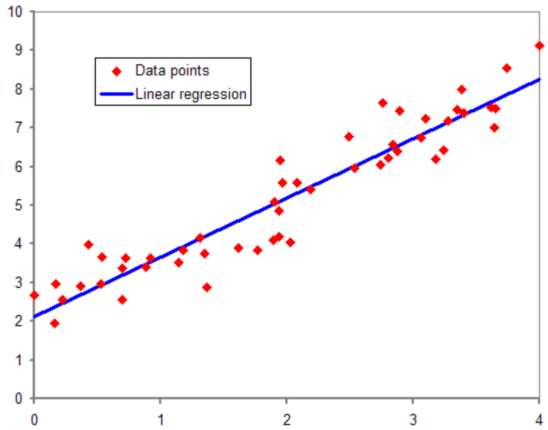

Figure 4.16: Regression line for 50 random points in a Gaussian distribution around the line y=1.5x+2

Another way to consider this to compare it to the typical regression mathematical problem. Given data points, a "best-fit" line is sought that "explains" the relationship between x, y as seen in 4.16.

Now consider the special case of a linear function acting as a ML model:

$$f(x) = \theta^T x + \theta_0$$

where θ and θ_0 are the unknown variables that the algorithm "searches for". Once these values are selected(through a training regimen that gives the best results for the dataset), the function can be completed (trained) and now solved for $f(x_i)$ where x_i is the input given. Note that the training process is significantly more computationally expensive than using the model for predictions.

The above case can be generalized regarding its function to apply to all machine learning models. In fact it is common that ML models are thought of as black boxes with inputs and outputs. Many machine learning models are quite simple in their workings while others are significantly harder to interpret such as an artificial neural network.

A further decomposition of supervised learning is the type of expected output:

- **Classification**
- **Regression**

4.6.2.1.1 Classification A typical supervised learning task is known as classification. A classification problem is one where the ability to classify something into a distinct set or class is sought. The spam filter is a good example of this as it is trained to recognize when an email is legitimate or generally unwanted by the user. Similar systems are used to classify various items and activities in everyday life. Note that while the target feature is returned as a binary data point or otherwise known as a categorical variable, the internal workings of the algorithm do work and use continuous variables.

Supervised learning algorithms for classification encompass a diverse array of methods, each with its own strengths, weaknesses, and underlying assumptions. Some of the most widely used classification algorithms include:

4.6.2.1.1.1 Logistic Regression Despite its name, logistic regression is a linear model used for binary classification tasks. It estimates the probability that a given input belongs to a particular class.

4.6.2.1.1.2 Decision Trees Decision trees recursively split the feature space into regions based on feature values, leading to a hierarchical structure resembling a tree. They are intuitive, interpretable, and capable of handling both categorical and continuous features.

4.6.2.1.1.3 Random Forests Random forests are an ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes (classification) or mean prediction (regression) of the individual trees.

4.6.2.1.1.4 Support Vector Machines (SVM) SVMs find the optimal hyperplane that separates classes in the feature space with the maximum margin. They are effective for high-dimensional data and can handle both linear and nonlinear classification tasks using kernel tricks.

4.6.2.1.1.5 Neural Networks Deep learning techniques, particularly neural networks, have gained prominence in recent years for their ability to learn complex representations from data. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are widely used for image classification and sequential data, respectively.

4.6.2.1.2 Regression Another typical task is to predict a target numeric value, based on features that are hypothesized to influence the target value also known as predictors. This is very similar to the typical mathematical regression as explained above and in 4.16; however the machine learning model is responsible for determining which "line" fits. There are multiple examples of such problems, for example predicting the future power consumption of a city based population, industry intensity, weather etc, or figuring out the stock price based on the movement of other indicators and many more.To train such a system, the basic supervised approach is required. The only difference with classification is that the target feature is a continuous value.

4.6.2.2 Unsupervised learning

In unsupervised learning, the training data is unlabeled meaning there is no information included with the data that points to what data are expected to influence the others. The algorithms basically try to discern any patterns between the data if they exist. This process is easy to find in everyday life as well and unsupervised learning problems are usually allowing us to explore connections that were not before evident or understood. For example, a large shopping mall could utilize such a system to figure out the shopping patterns of visitors and categorize them in groups. Coming back to machine vision, an unsupervised ML model could categorize all photos given to it according to similarities it can discern between them such as splitting animals from humans, or categorizing all same animals together in clusters. In this case, no labelling of the photos is done so that the algorithm can "compare" photos to known classes but instead has to figure out what photos have similarities itself.

4.6.2.3 Reinforcement Learning

Reinforcement learning is one of the most exciting and powerful areas of ML. Generally, it mimics to a large degree the learning process for humans including the underlying understanding of the problem and the adaption to changing circumstances [Rebala et al., 2019]. It can be thought of as the model learning the strategy of solving a problem rather than the exact answer to the problem given the inputs right now. ML models trained in this way have been shown to achieve complicated tasks such as driving cars.

4.6.3 Performance metrics

There are several metrics that can be used to quantify the performance of a machine learning application. Classification and regression problems both also have specific metrics that can apply. While several of them were used to create those models, there will not be extensive discussion on their formulation or other metrics that were not used. Similarly, the implementation of the ML process pertains more to computer science and as such it will not be discussed extensively nor is the author claiming that it is properly optimized. The main objective is to show how this approach works and demonstrate its potential for the particular problem. The metrics as presented below are formulated and used as part of the Python scikit-learn(sklearn) library.

4.6.3.1 Classification Performance Metrics

4.6.3.1.1 Accuracy Accuracy is a simple metric that is defined as the sum of correct predictions divided by the total number of predictions. For example,

$$\begin{bmatrix} Predictions \\ Truth \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{4.8}$$

$$Accuracy = \begin{bmatrix} 0 & 1 & 1 & 1 \end{bmatrix} = 3/4 \tag{4.9}$$

In the above example, we have matching predictions with truth on 3 cases while all the cases are 4 hence the score comes to 75%. It is an easy way to see if the classifier is working and how good it is at identifying correctly the class. However, it can be misleading on its own, especially in cases where the classes are imbalanced. For example, if 90% of the data is positive then a classifier that only returns positive will be accurate at 90%.

4.6.3.1.2 Precision Precision is given by:

$$Precision = \frac{True_{pos}}{True_{pos} + False_{pos}} \tag{4.10}$$

Where $True_{pos}$ is the number of true positives and $False_{pos}$ is the number of false positives. Positive and negative here refer to the possible states of the binary classifier. The precision metric provides insight into the classifiers ability to not generate false positives. This metric can be very important depending on how the problem is formulated. In this present application, the positive label is given to correspond to a survival factor of 1, hence a false positive would mean that a dangerous prediction is returned. The ideal value is 1 while the worst is 0.

4.6.3.1.3 Recall Recall is a metric that captures the ability of the classifier to find all the positive samples. It is given by:

$$Recall = \frac{True_{pos}}{True_{pos} + False_{neg}} \tag{4.11}$$

Where $False_{neg}$ are the number of false negative predictions, or predictions that are positive but are misclassified. The ideal number is once again 1 and worst 0.

4.6.3.1.4 F1 score The F1 score also known as F-score or F-measure combines precision and recall in one metric by taking their harmonic mean. It is useful to compare two different classifiers by evaluating precision and recall together.

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{4.12}$$

Again, the best value is 1 and the worst is 0. The F1 score is useful in cases where the accuracy is not as robust. Namely for imbalanced datasets as discussed. Also, it can better capture the difference in consequences of a false positive/false negative. For example, for a cancer recognition application a false negative is many times more costly than a false positive (considering that follow-up investigations will reveal it to be false).

4.6.3.2 Regression performance metrics

Regression performance metrics revolve around the idea of fitting a curve to a cloud of data points and associated metrics that can be generally found in mathematics and statistics.

4.6.3.2.1 R^2 coefficient of determination In statistics, the coefficient of determination is routinely used to explain how well the statistical model replicates the observed data points. Similarly, in ML it captures how well the model can predict the data given and as such a high value is sought.

Also known as the coefficient of determination, R^2 represents the proportion of the variance in the dependent variable (target) that is explained by the independent variables (features) in the model. Here's a detailed explanation:

Mathematically, it is expressed as:

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

Where:

- SS_{res} is the sum of squared residuals (or errors) between the actual and predicted values.
- SS_{tot} is the total sum of squares, representing the variance of the dependent variable.
- R^2 ranges between 0 and 1, where:
  - $R^2 = 0$ indicates that the model does not explain any variance in the target variable.

  - $R^2 = 1$ indicates that the model perfectly explains all the variance in the target variable.
  - $0 < R^2 < 1$ indicates the proportion of variance explained by the model, with higher values indicating better model fit.

$R^2$is commonly used as a performance metric to assess the goodness of fit of regression models. It provides insight into how well the independent variables predict the variation in the dependent variable. Higher R^2 values indicate that the model captures more variance in the data and is therefore more effective at predicting the target variable. However, R^2 should be used in conjunction with other evaluation metrics to fully assess the model's performance and generalization ability.

- R^2 can be influenced by the number of features in the model, and adding more features may artificially inflate its value.
- It does not indicate whether the model's predictions are unbiased or accurate; therefore, it should be used alongside other metrics such as mean squared error (MSE) or root mean squared error (RMSE) for a comprehensive evaluation.

In summary, R^2 provides a useful measure of how well a regression model fits the observed data, offering insights into the predictive power of the model and its ability to explain variance in the target variable.

4.6.3.2.2 Mean absolute error Mean absolute error is also found in statistics and captures the average of the absolute errors for pairs of prediction-ground truth data.

$$\text{MAE} = \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

Where:

- n is the total number of data points.
- y_i represents the actual target value for the i^{th} data point.
- $\hat{y}_i$ represents the predicted value for the i^{th} data point as produced by the regression model.

4.6.3.2.3 Mean Squared Error (MSE) In the context of machine learning regression, mean squared error (MSE) is a key performance metric used to evaluate the accuracy of a regression model's predictions. MSE quantifies the average squared difference between the actual values (or targets) and the predicted values produced by the model. Here's a detailed explanation:

MSE is calculated by averaging the squared differences between the actual target values (y) and the predicted values ($\hat{y}$) for each data point in the dataset. Mathematically, it is expressed as:

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

Where:

- n is the total number of data points.
- y_i represents the actual target value for the i^{th} data point.
- $\hat{y}_i$ represents the predicted value for the i^{th} data point as produced by the regression model.

Mean Squared Error (MSE) quantifies the average squared difference between the predicted and actual values in regression tasks. By squaring the differences, MSE gives greater weight to larger errors, thereby making the metric sensitive to outliers. Consequently, a smaller MSE signifies superior model performance, with an ideal value of 0 indicating a perfect fit where predicted values precisely match the actual values.

In machine learning applications, MSE serves as a loss function during the training phase of regression models. Throughout training, the model's parameters are iteratively adjusted to minimize MSE, leading to enhanced predictive accuracy. Following training, MSE continues to play a vital role in assessing the model's effectiveness on unseen data. Lower MSE values suggest superior generalization ability, indicating that the model can accurately predict outcomes beyond the training dataset.

4.6.3.2.4 Root mean square error Root mean square error (RMSE) or root mean square deviation is another tool very commonly found in statistics that once again captures errors but in contrast to MAE each error has a different effect, especially for the outliers.

$$\mathrm{RMSE} = \frac{1}{n}\sqrt{\sum_{i=0}^{n-1}(y_i - \hat{y}_i)^2}$$

4.6.4 Advantages & disadvantages of ML

ML has distinct advantages in solving several problems compared to a more traditional approach. Given a dataset of high resolution and sufficient entries ML is much better at discovering patterns in the data than a human programmer. For example, a person would perhaps plot the x, y input and outputs in a graph and then determine their relation by fitting a curve or using statistics to extract some truths about the underlying processes. However, one can easily understand that when the input number starts to increase or generally the complexity increases, it becomes much harder to formulate theories as to their interrelation. Examples of such problems are; the prediction of the power consumption of a city, distinguishing written characters, predicting health outcomes, designing control systems for autonomous vehicles and many more. The main disadvantage of ML methods is inherent to their function as something that can change and adapt itself. This is especially true for artificial neural networks which are almost impossible to interpret why they function the way they do. While the mathematical formulation of these models is well known, the complexity of a trained model makes it very opaque to direct interpretation. In fact, most of the workings of ML models are assumed to be a black box that is then evaluated by changing parameters and observing the results. Of course, there exist multiple procedures and metrics to evaluate the performance of a model and optimize it, but the way to do that may involve heavily trial and error. In a sense, this process bears a great resemblance to tuning a PID system.

Chapter 5

ML implementation

Machine learning, as introduced earlier, is an approach that has been used in this case to get similar results to the case-based approach. More specifically using ML we can train models on the data in the database and then use them to extract predictions.

5.0.1 Methodology

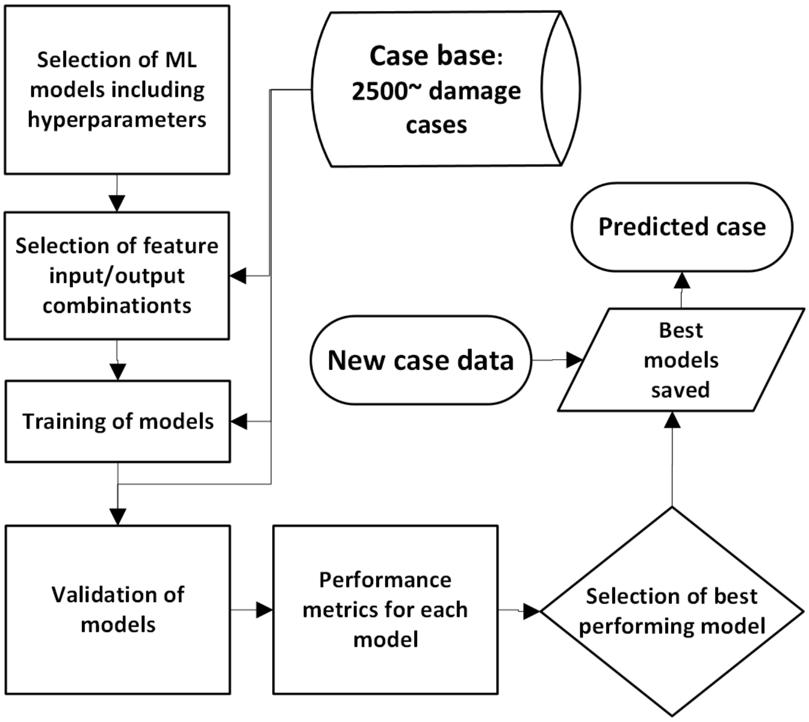

Figure 5.1: ML flowchart as implemented

The implementation of ML methods happens in parallel to the CBR approach in the same script, relying on the Python library, scikit-learn[36]. The objective is to utilize the information found in the dataset to provide predictions for an emergent case. In contrast to the CBR approach, many of the functionalities of how this happens are not distinct and are defined by its models internal logic as well as pre and post-processing steps.

The methodology followed for using ML is as follows:

Firstly, selections are made regarding what parameter should be predicted and the feature variables that will be used to train the model.

An array of candidate models is compiled, and their performance is compared after fitting them to the dataset. In general, the most accurate and fast model is retained. A combination of the models is also evaluated and used if proven better than individual models.

After the selection is made the model is optimized in a process called hyperparameter tuning, where parameters that refer to the models internal operation are optimized.

The resultant model is then trained on the data and used for the predictions.

CBR approach implementation Introduction All the above processes and algorithms are codified using the Python language in an extensive script containing the following parts:

1. Database creation and pre-processing
2. Model comparison & tuning
3. Model fitting & stacking
4. Model evaluation
5. CBR methodology including
6. Similarity calculation
7. Adaption of selected case data to a prediction
8. Prediction using ML
9. Prediction using CBR
10. Time-loop accepting new data for every time step and providing a new outcome.

All the above will be briefly demonstrated below. The application is split into classification & regression processes.

5.1 Machine learning procedure

The steps followed for each ML implementation are:

1. Pass database of cases
2. Select features from database and the target feature (objective to be predicted)
3. Select a machine learning model, e.g. RandomForestClassifier.
4. Choose whether to binarize the target feature
5. Pass the threshold for binarization
6. Choose whether to standardize the data

An example of a wrapper function for the procedure is seen below with passed data

```
def df_to_model(dfNP, feautures_N, target_feat,
                                    RandomForestClassifier(),
    True, 0.7, False)
    ...
    return
```

Figure 5.2: Example of wrapper function for training a model.

The function then manages the steps of splitting the dataset to train and test sets. The split used is 0.3, meaning 30% of data is kept for the testing/validation step. This is done since if the model is trained on the data used to also test it, improved performance will be returned as the cases have been found before. The good results would therefore be misleading. It is hence common practice to take random slices of the data for training and testing. One step further is the process of cross-validation. In cross-validation, the splits of train-test data are done multiple times each time measuring the performance of the model. After a number of runs, the data can be collated. In this way, randomness is further reduced. Below is an example of a sensitivity analysis (introduced later) done on 10 folds of data.

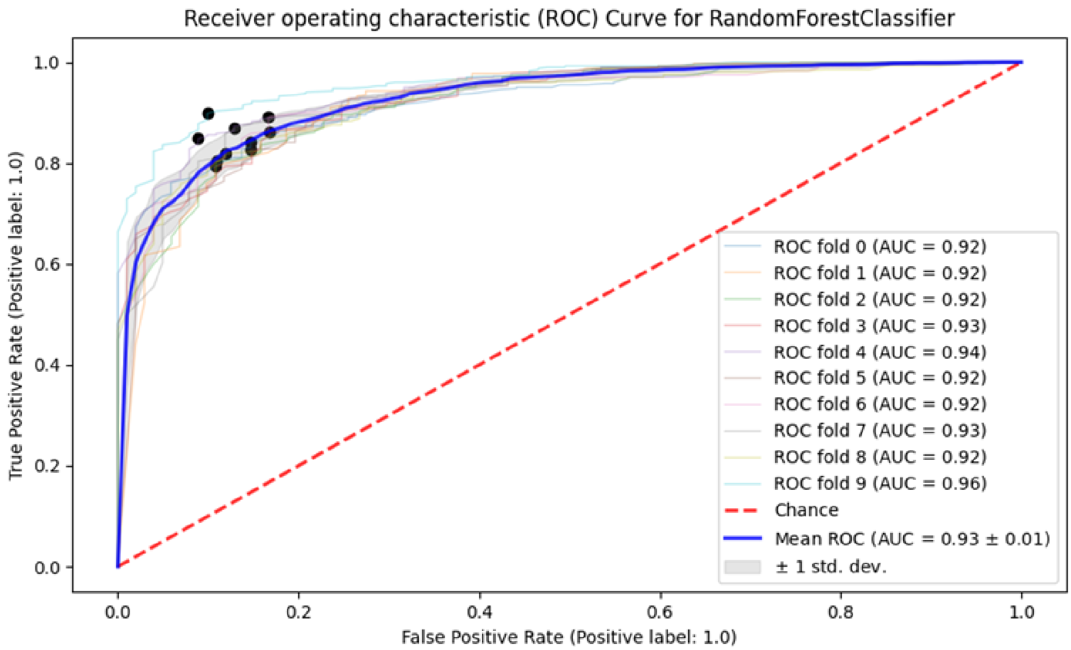

Figure 5.3: Receiver operating characteristic curve for RandomForestClassifier

After the split is done correctly the model is then trained on the data. Lastly, the trained model passes through an evaluation function where several of its performance metrics are computed as well as other outputs. Some are also printed as below:

Metric	**Value (Std. Dev.)**
Accuracy	0.881 (0.016)
Precision	0.917 (0.017)
Recall	0.937 (0.018)
F1 Score	0.927 (0.010)

Table 5.1: Performance metrics for RandomForestClassifier predicting SFAC.

Note that the values are returned as mean and the standard deviation in parentheses since the model is evaluated across all folds or splits of data. The trained model is returned and can then be used for predictions.

5.1.1 Classification

There are no features of the damage cases that fit the description of a binary classifier. However, with some adjustments, several can be created. For example, the most important classifier feature that was adapted is the survival factor. While normally SFAC ranges from 0 to 1, the extremes are normally what is seen. Also, values lower than 1 but not zero usually represent a degraded survival, usually one with a heel angle. Hence, cases with a value larger than 0 are generally survival cases albeit marginal ones. Consequently, by choosing a threshold value in the range of 0 to 1 and then binarizing the SFAC we can convert it to a binary classification output. For example, if we choose 0.7 as the threshold, then all entries with a value higher than 0.7 will become 1 and those lower will become 0. A high threshold will generate more non-survival cases from the marginal ones while a lower one is the inverse. The threshold value used is 0.7 in the code. A Binarizer function carries out the procedure across the database.

Another possible classification target is a new feature named Capsize generated by interpreting the data already found like heel, trim etc.

5.1.1.1 Classifier comparison

One of the first steps is to select appropriate models for classification. A function that can compare them is written for this purpose. In the graph on the next page, a visualization of the classifiers can be easily made. For each classifier, a graph of the data points with a background is plotted. First, each data point represents a case. Their colour corresponds to the SFAC value of the case, blue for 1 and red for 0. The location is based on two of the features plotted: length and location. These are the most important.

While the points are common for all classifiers, the background basically depicts the predictions of the model. According to the intensity of the blue and red hues, each area will have a higher or lower probability to predict 1 or 0. The relation can be easily seen in several of them. It is also to see the different geometric patterns of each model.

At the bottom right, the accuracy score of the classifier is printed e.g., .82 for 82 Lastly, the last classifier, the stacked classifier, is actually a combination of all the other classifiers. In general, all classifiers have similar performance as does the stacked model.

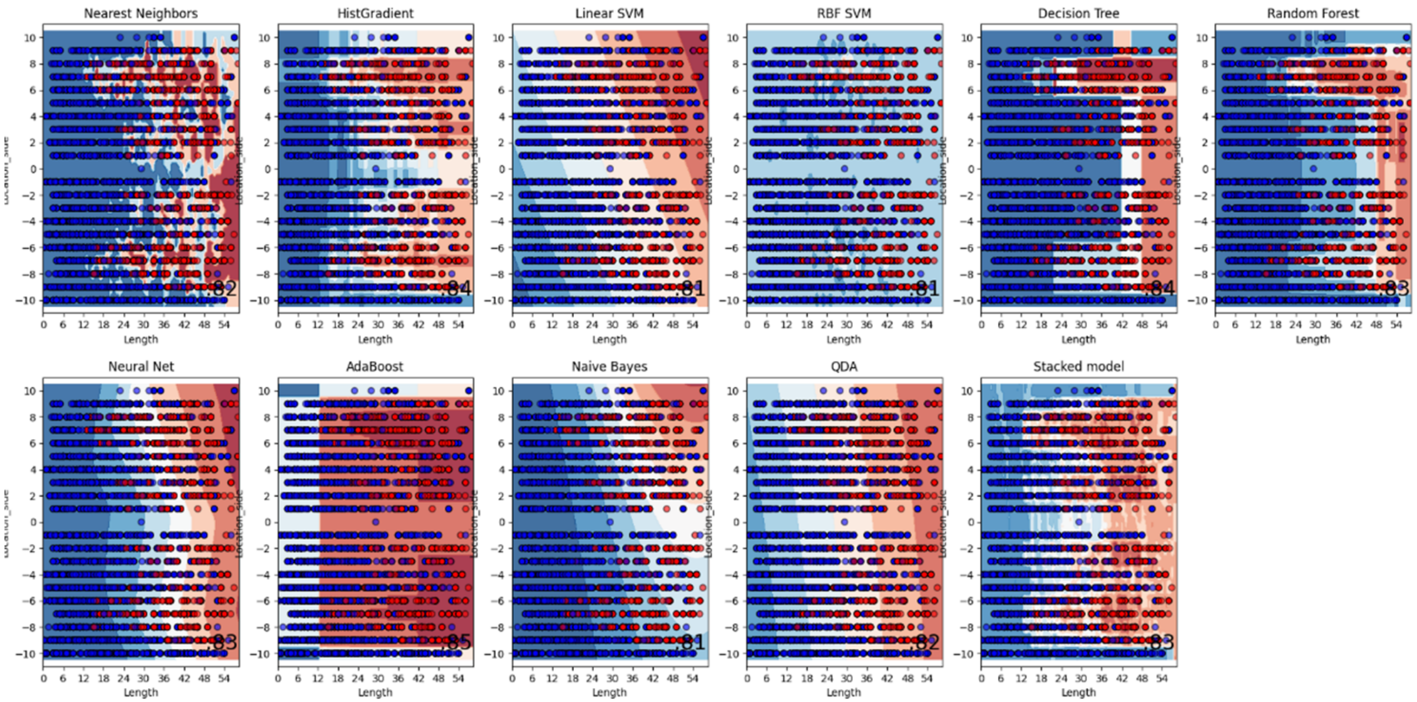

Figure 5.4: Classifier comparison: Bullet points represent known data while painted background displays the model predictions. X axis is length and Y axis is location. Only these two features are shown here while the models use more.

5.1.1.2 Calibration of models

Calibration in the context of machine learning is a process of tweaking or calibrating the probability output of the model to match the ratio of positive classes in the relevant set. To better understand this, it is important to know that while we expect a binary class as a result of a classifier, in reality, the classifier returns the marginal probability that the class is either positive or negative with the sum of them adding up to 1. For example, a possible result could be the numbers 0.04/0.96 meaning a very high probability of the 2nd class being true. The threshold for selecting if the result is positive or negative is usually at 0.5 and will be further expanded later. Therefore, in the calibration process, the aim is for the returned probabilities to depict the real likelihood of the true events. For example, in the problem of a coin toss, a model is perfectly calibrated if the results are always 50%-50% while an uncalibrated model might have half the times 20% and the other half 80% which does not represent the reality of the coin toss mechanics. A popular tool to visualize the calibration is the calibration curve, where the mean predicted probability and the fraction of the positive class are plotted. The models predictions are usually binned in 10 or more bins. Ideally, the data points should lie on the y=x diagonal. In reality, the model may diverge at some points. Below is the plot for the classifiers considered in 8.5.5.3. The best method is also shown separately before and after calibration. The effect is very easy to discern. The calibration is done using the isotonic and the sklearn.calibration.CalibratedClassifierCV method.

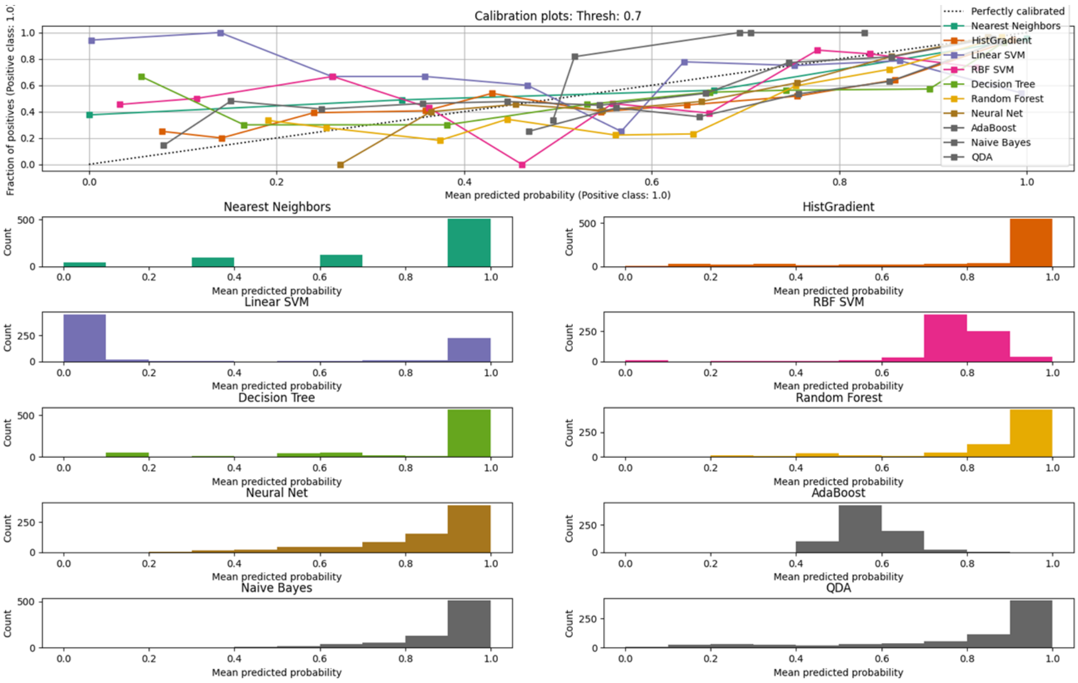

Figure 5.5: Uncalibrated calibration plot for all classifiers

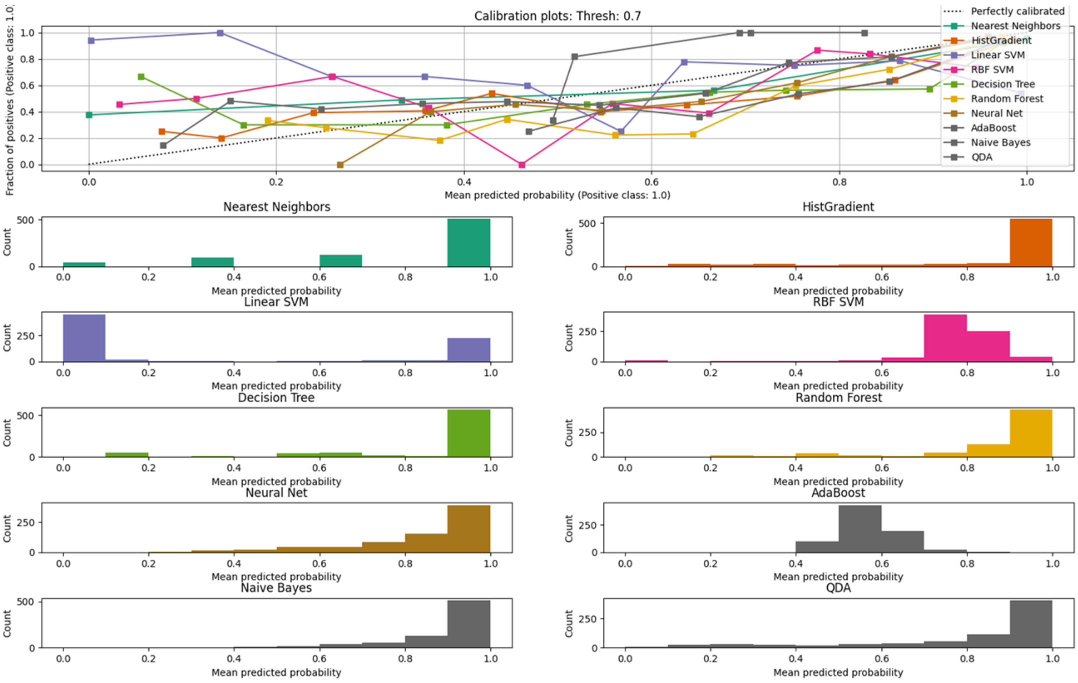

Figure 5.6: Calibrated calibration plot for all classifiers

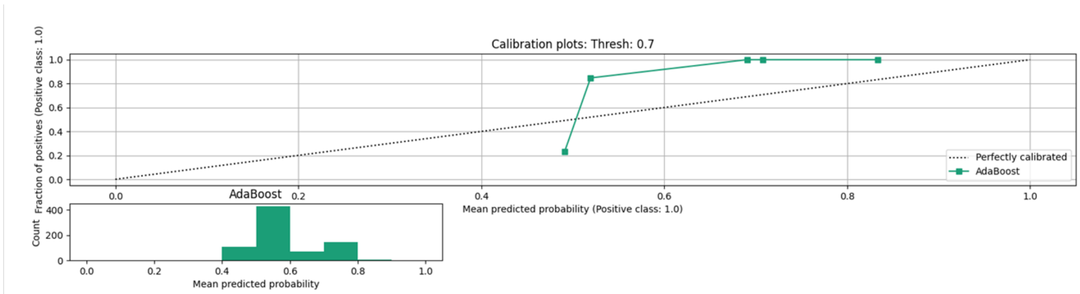

Figure 5.7: Uncalibrated and calibrated calibration plots for the best method chosen from the classifier comparison

5.1.1.3 Receiver Operating Characteristic

Central to understanding how a classifying model produces a result is to understand the discrimination threshold. As discussed before in the previous chapter about calibrating the model, the probability of the class is what is retuned and not the predicted class. It has also been mentioned that this happens after a threshold is introduced that specifies the minimum positive class probability to declare the class positive. This discrimination threshold can be adjusted with great effect on the models performance. A low threshold will be more sensitive, hence having higher recall but will also inadvertently generate more false positives thus lowering the precision metric. The opposite is true for a higher threshold. The problem of optimizing the performance of the model hence becomes 2 dimensional and will also depend on the exact use of the model. A useful way to visualize the effect of differing thresholds can be afforded by a graph called receiver operating characteristic graph or ROC. The shape of the curve can easily demonstrate the bias of the model toward recall or precision, higher true positives and lower false negatives and lower false positives and higher true negatives.

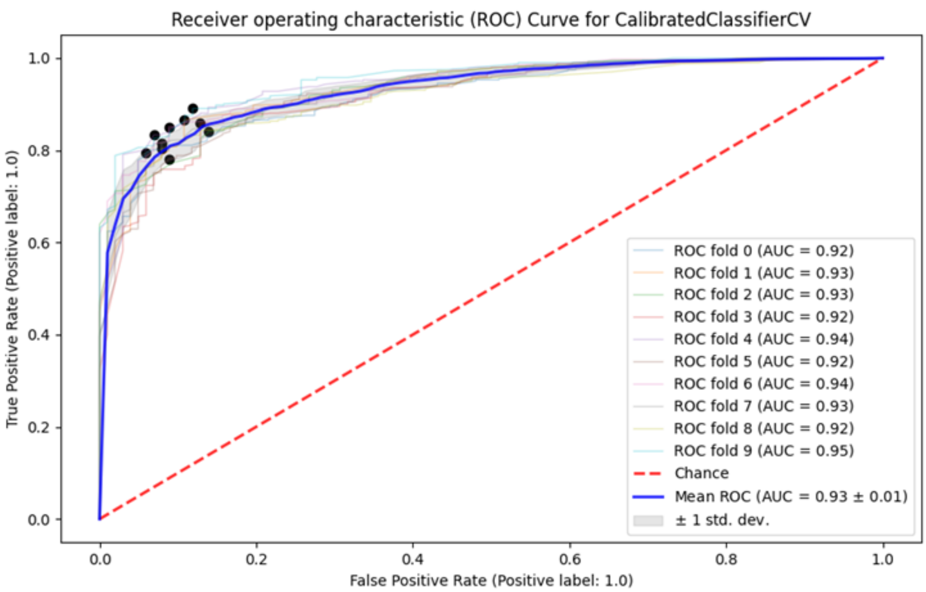

Figure 5.8: ROC curve for the calibrated classifier (best found in comparison)

The ROC curve is plotted above. Note the diagonal which signifies the chance or a random guess instead of a model making the selection. In that case, the no-skill classifier chooses the class with a higher probability always (<0.5). The true positive rate is plotted on the y-axis and the false positive rate is on the x-axis. Ideally, a high TPR is sought but whilst the FPR is kept low. A high TPR with an equally high FPR means that while lots of the positive class (high recall) are captured, there are also lots of misclassifications (low specificity). In some cases, preserving the precision can be more important than capturing all positive classes. The result for the default 0.5 thresholds for the calibrated classifier is printed below (values are the mean of the different folds with the value inside the parentheses being the standard deviation):

Metric	**Value (Std. Dev.)**
Accuracy	0.877 (0.020)
Precision	0.907 (0.015)
Recall	0.943 (0.019)
F1 Score	0.925 (0.012)
Best threshold (avg)	0.76 for 0.74 Youdens and 0.47 for 0.93 F1 score

Table 5.2: Performance metrics for CalibratedClassifierCV predicting SFAC.

The means to find the optimum threshold are to evaluate two values; one is Youdens (J) statistic and the F1 score already introduced. For maximising the Youdens index the threshold should be moved to 0.76 while to maximize the F1 score the threshold should be moved to 0.47, which is actually very close to the 0.5 default value. Youdens index is simply,

$$J = Sensitivity + Specificity1 \tag{5.1}$$

$$J = TruePositiveRateFalsePositiveRate \tag{5.2}$$

It is not clear which index is best used to optimize. As discussed earlier however making as few false positives as possible is most important hence the $F1$ score is perhaps better suited. Also, this means that the default threshold is good enough and as such, it is not changed. Note that the printed result is the result of one of the folds.

5.1.1.4 Predictions using the classifier

The classifier after the calibration is ready to be used. The threshold is also left as is. The model is passed along with the emergent data to a wrapper function that returns the probability for the positive class (if applicable). The prediction is then printed along with the rest of the predictions that stem from the CBR approach.

Prediction Method (SFAC)	**Value**
Simple average	0.76
Similarity weighted prediction	0.75
Posterior probability weighted prediction	0.87
Prediction using the classifier on SFAC	0.41
True SFAC	0.51

Table 5.3: Predictions for case 27. True SFAC is 0.51.

Note that the prediction from the classifier is the probability for the positive class or the probability that SFAC is 1. It is not the survival factor as given in the SOLAS formulation.

5.1.2 Regression

For regression, there are lots of possible target features with different ranges. For example, the heel is quite important as well as the range of positive stability and GZ. SFAC is also a continuous variable that can be predicted. A very similar approach is followed first comparing the models or regressors. Of course, the continuous nature of the data changes several ways of approaching the regression problem compared to the classification.

5.1.2.1 Regressor comparison

Like the classifier comparison, regressors are also compared to decide which model is best. See below the plots used to compare the regression models when fit to the same data to predict heel, GZMAXR and others. For each model, the perfect result would be for the data points to lie in the diagonal where predicted=measured. The predicted value for a data point is on the y-axis and the x-axis is the true value. The distribution of those points is easy to see and how far they are from the perfect diagonal line. On the top right the coefficient of determination, R^2 , is printed along with the Mean Average Error and the 95% confidence interval. All those performance metrics are a distribution since the model is evaluated on different folds of train-test data. Hence the mean along with the standard deviation is printed. We can distinguish that the best regressors are gradient boosting and random forest. The stacked model is also quite good but not better than the individual gradient boosting regressor in terms of capturing the variance through the coefficient of determination or the mean absolute error. The gradient boosting model is selected finally for predicting heel. It has good R^2 and the best MAE of all models. The same model is found to be best when predicting GZMAXR.

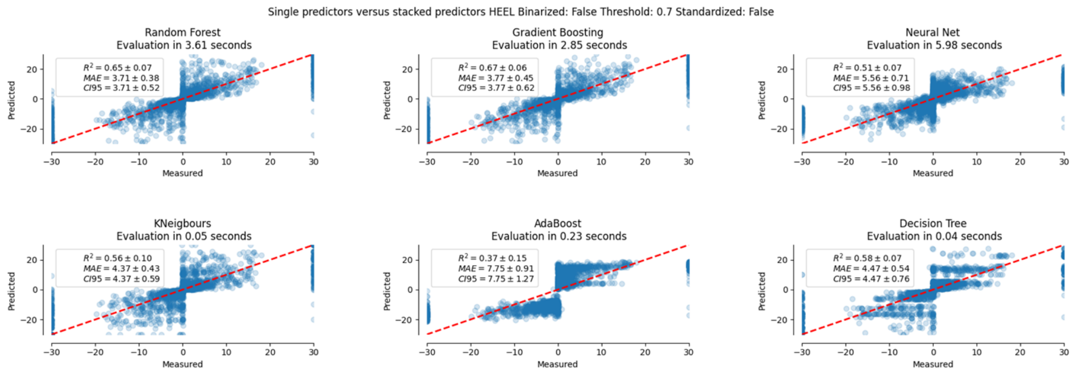

Figure 5.9: Result from regressor comparison on Heel. Data points are plotted on predicted vs measured axis. Notice the coefficient of determination, mean average error and 95% confidence interval evaluation for each model

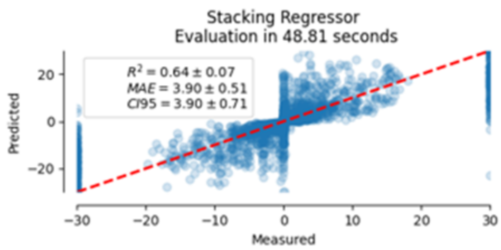

Figure 5.10: Stacking regressor (combination of all previous models for predicting heel).

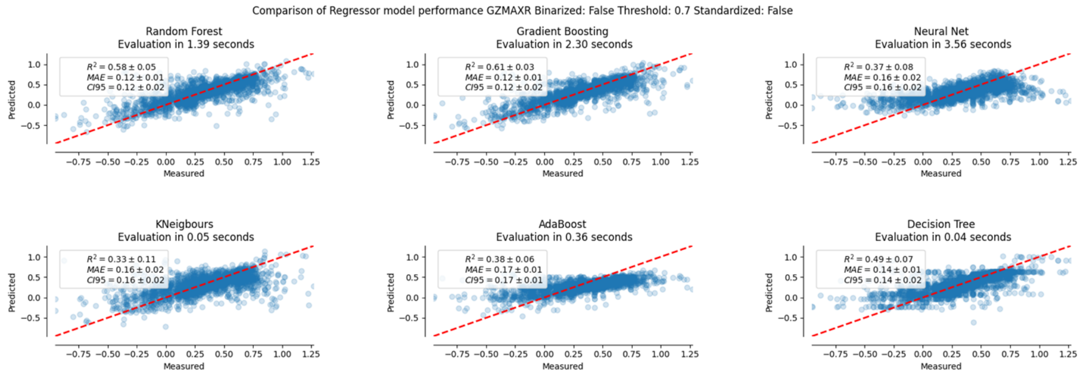

Figure 5.11: Result from regressor comparison for predicting GZMAXR

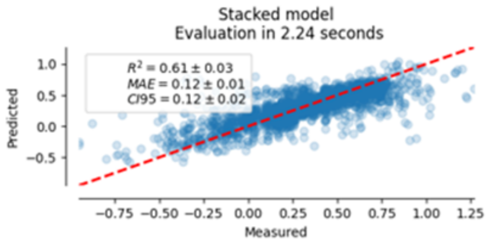

Figure 5.12: Stacking regressor (combination of all previous models for predicting GZ-MAXR).

5.1.2.2 Predictions using the regression model

The predictions are printed along with the rest of the output from other models and the CBR approach.

Table 5.4: Tabulated results for case 9847

	CBR Approach		ML	Truth
	Similarity weighted predicted	Posterior probability weighted	Machine learning prediction	
Heel	0.66	0.12	1.21	-0.02
GZMAXR	0.3	0.33	0.32	0.29
SFAC classifier	0.94	0.98	0.79	0.98
SFAC continuous	0.94	0.98	0.67	0.98

The above process can be followed for all features found in the database.

5.1.2.3 Time domain data & ML

While we have sparse data from the time domain simulations, we can still train a model on the data that is specific to those cases. Unfortunately, the amount of data is not appropriate to make an accurate predictor model but can still be used. For example, here is a regressor attempting to predict the time to capsize (TTC) time trained using the following features: GZMAXR, RANGEF, Max Roll within 3 mins, Length, $Location_{side}$. The max roll at 3 mins is a value only found through a time domain simulation. The rest of the values can be individually predicted by the regressor as shown above. The few points available make it hard for the models to be properly trained hence the very low performance (see Figure 49) In contrast, if we populate the TTC values for all entries in the following manner:

$$ifSFAC = 0, TTC = mean_{TTC}$$

$$ifSFAC > 1, TTC = ((4010 - mean_{TTC}) \cdot SFAC) + mean_{TTC}$$

The $mean_{TTC}$ is the mean of the values for TTC computed by the time domain cases. The result of training on that data is seen in 5.14. Another possible way to derive a database that is large enough, is to actually use a ML model on the existing database to extrapolate the rest of the cases that do not have time domain data. To achieve that the features passed are data from the NAPA cases and the target is the TTC value. So, the model will develop a relationship between the NAPA data and the TTC value only for the cases that have time domain data. Then this relationship will be applied to all cases and get a predicted TTC. The result is seen in Figure 51.

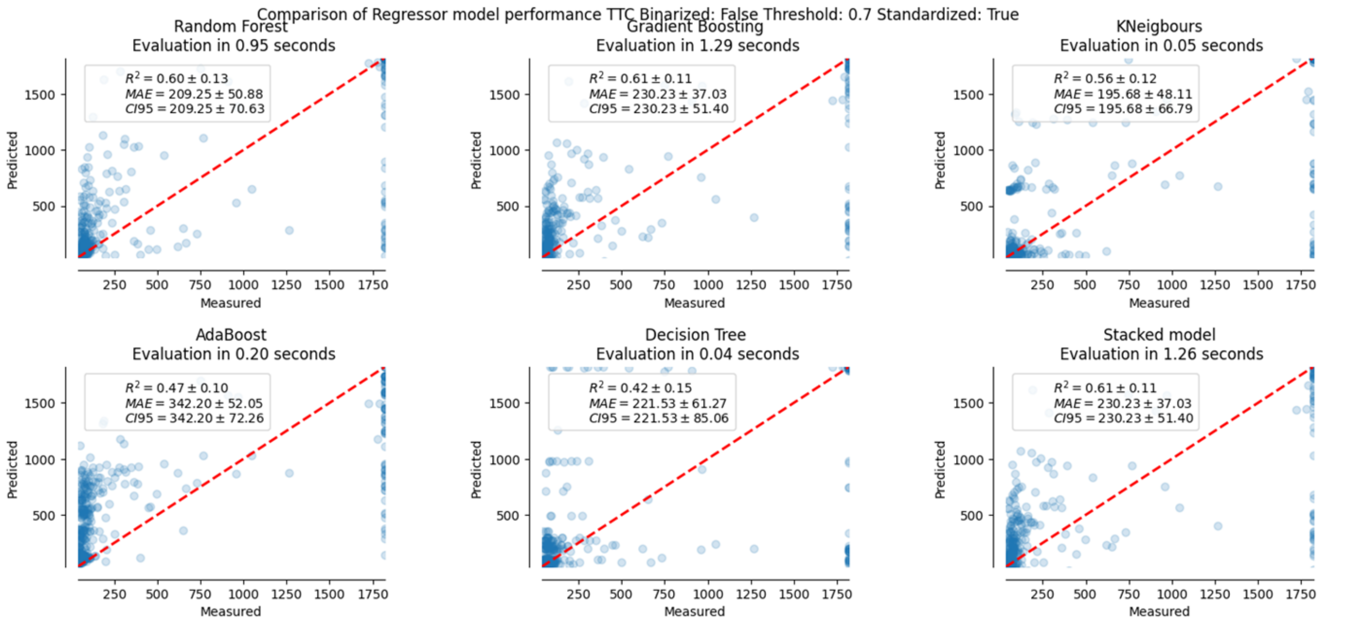

Figure 5.13: Regressor comparison for predicting Time to Capsize using time domain data.

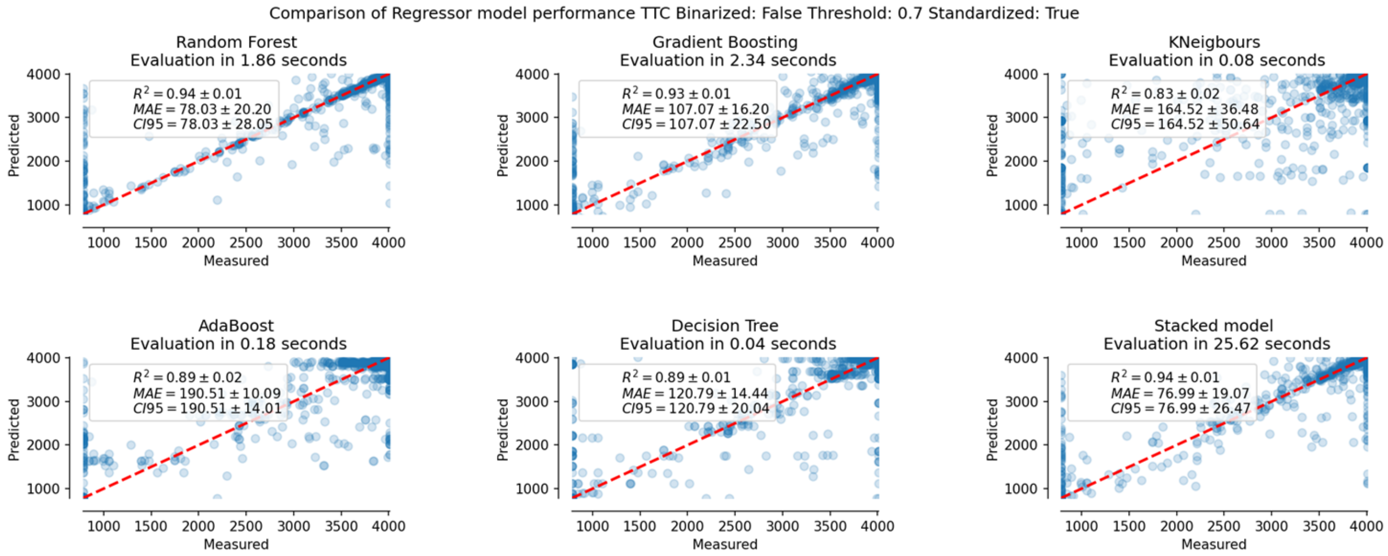

Figure 5.14: Regressor predicting Time to Capsize trained on all the data with transformations explained above (mean approach).

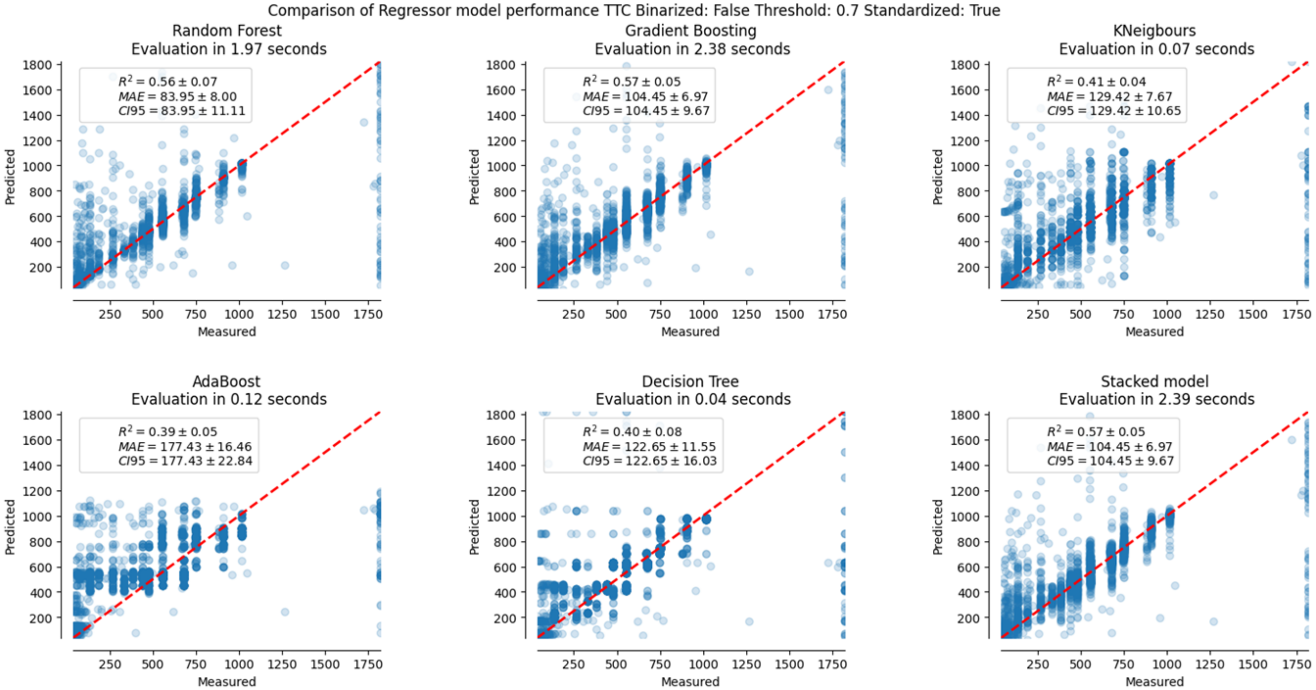

Figure 5.15: Regressor predicting Time to Capsize using data from time domain cases and training ML models.

Chapter 6

CBR implementation

6.1 Introduction

In this chapter the CBR side of the methodology is demonstrated as is implemented also making use of several case studies that help appreaciate and understand its workings as well as gauging its performance.

There are multiple ways to present the predictions from CBR, either as the average, the top case, or weighted averages taking into account also the probability of some cases actually happening coupled with their similarity to the observed one. The combination of those values gives a range of predictions that reinforces consistency and accuracy.

Table 6.1: Output prediction types

Name	Sample	Method
Simple average	Selected cases (by percentile similarity cutoff)	$\frac{\sum_{i=0}^{n}(var_i)}{n}$
Weighted similarity average	Selected cases (by percentile similarity cutoff)	$\frac{\sum_{i=0}^{n} var_i \ Similarity_i}{\sum_{i=0}^{n} Similarity_i}$
Weighted Posterior probability average	Selected cases (by percentile similarity cutoff)	$\frac{\sum_{i=0}^{n} var_i \ P(\mathrm{x} \mid \mathrm{z})_i}{\sum_{i=0}^{n} P(\mathrm{x} \mid \mathrm{z})_i}$
Top case	Top case by similarity	Top case feature value

If the similarity of the selected cases is very high, then the top case might be a good source of information. The weighted similarity is also useful when high similarity cases are present. The posterior probability will theoretically be accurate even when similarity information is scarce as the a priori distribution is included. The simple average lastly represents probably the most conservative estimate as it treats all similar cases equally. Thus, minor to medium biases by the similarity function or errors and noise in the input data is negated. All the above also rely on the sample used which is in turn determined by the percentile used as a cutoff for including a case. In the examples above the cutoff is set at the 92nd percentile. If this is increased, then fewer cases will be included with higher similarities and inversely a lower one will include cases of smaller similarities thus changing considerably the simple average values and less so the weighted averages. The exact values that will need to be set are not something that has been extensively explored and will be the result of fitting this approach into an actual system and testing its response. The algorithm was tested in several scenarios and the results are presented here below.

6.2 Case studies

6.2.1 Case 22

Case 22 is selected to be emulated by the (noisy) case generator. We can also pass a value of 0-1 to the case generator to randomly add noise to the data generated as the inputs. First, the case without any noise requirements. Below is the output of the code.

Table 6.2: Predictions for Case 22

Prediction Type	**Value**
Simple average	0.99
Similarity weighted prediction	0.99
Posterior probability weighted prediction	1.0

Table 6.3: Most Likely Cases for Case 22

Case	**Similarity Score**	**SFAC**
22	1.00	1.00
2662	0.92	1.00
5029	0.89	1.00
9548	0.89	1.00
9537	0.88	1.00
9898	0.86	1.00
662	0.83	1.00
7237	0.83	1.00
1942	0.82	1.00
3565	0.82	1.00
Weighted average SFAC of 10 most likely cases		1.00

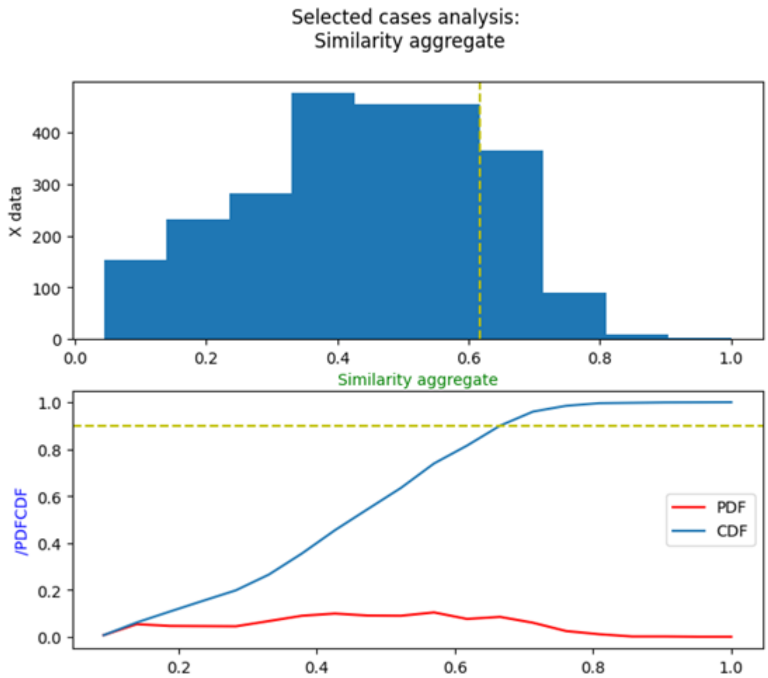

Figure 6.1: Similarity aggregate graphs for Case 22

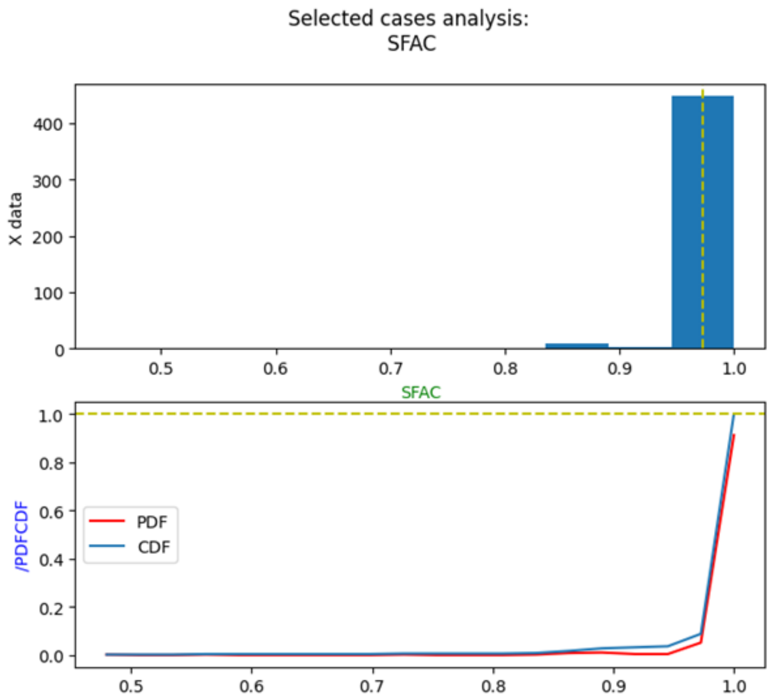

Figure 6.2: SFAC of selected cases

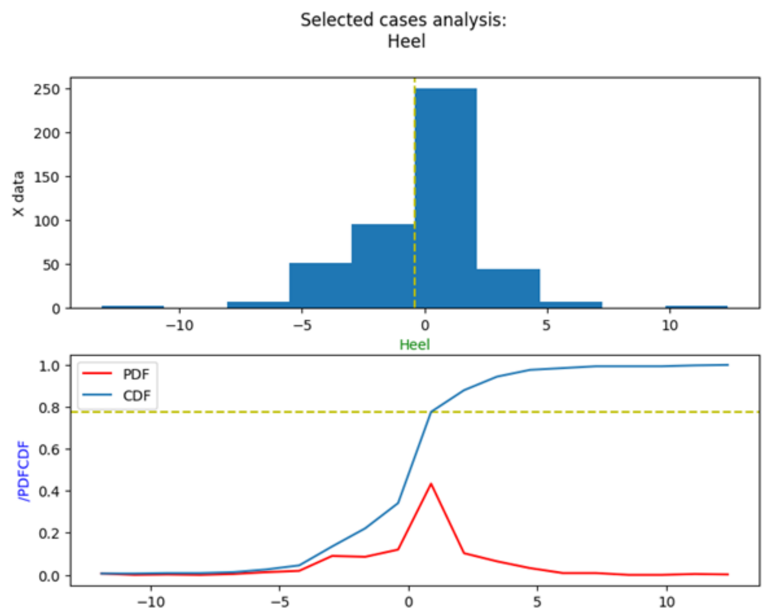

Figure 6.3: Heel of selected cases

In the printout, the basic output predictions are presented as well as a rundown of the 10 top cases along with their SFAC and weighted average based on their similarity score. The graphs also describe the distribution of all selected cases and can give a qualitative overview of the predictions. Namely, the similarity graph shows the subset of the selected cases and how the scores are distributed. There are relatively few high similarity cases, suggesting a quite specific prediction. From the slope of the CDF and especially in the selection area (above the yellow line) the same is deduced. Moving on to the SFAC graph, it is clear that the majority of cases are 1. This is also easily confirmed by the PDF/CDF. Lastly, the heel graph shows the distribution of heel values of the cases with a high density of around 0.5. The mean is plotted with the yellow line as well as its corresponding point on the CDF.

All extracted objects of information point to the correct values with a high degree

Table 6.4: Summary of predictions case 22

Source	SFAC	HEEL
Truth	**1**	**-0.35**
92nd perc. average	0.99	-0.264
Similarity weighted	0.99	-0.28
Probability weighted	1	-0.19

of certainty. Repeating that for case 22 but with added noise, meaning that damage variables will be slightly off their true values and there is the possibility of rooms being missing or replaced with other random ones that may not belong to this case. The exact workings of the noisy case generator are not explained fully but basically revolve around set ranges of noise data as a percentage of the real values. These limits are set so as not to drift too far away from the case to emulate. Note that if another case is pointed as the most likely it is not an inaccuracy as simply the data given may point to another case with greater confidence.

6.2.2 Case 22 with noisy data

The output is seen below. The noise is also printed. The predictions are immediately lower, and we can see the similarity score of case 22 dropped to 0.86. However, the predictions are still very high and by glancing over the graphs we can still maintain high confidence that the case is safe. The fact is that for a safe case, surrounded by similarly safe cases, the algorithm will consistently find even large amounts of noise to not change the outcome much.

Table 6.5: Predictions for Case 22 (Noisy)

Prediction Type	**Value**
Simple average	0.97
Similarity weighted prediction	0.97
Posterior probability weighted prediction	0.99

Table 6.6: Most Likely Cases for Case 22 (Noisy)

Case	**Similarity Score**	**SFAC**
22	0.86	1.00
7619	0.86	1.00
5713	0.83	0.98
7803	0.83	1.00
8558	0.83	1.00
4304	0.83	1.00
2192	0.83	1.00
3690	0.83	1.00
4874	0.82	1.00
2384	0.82	1.00
Weighted average SFAC of 10 most likely cases		1.00

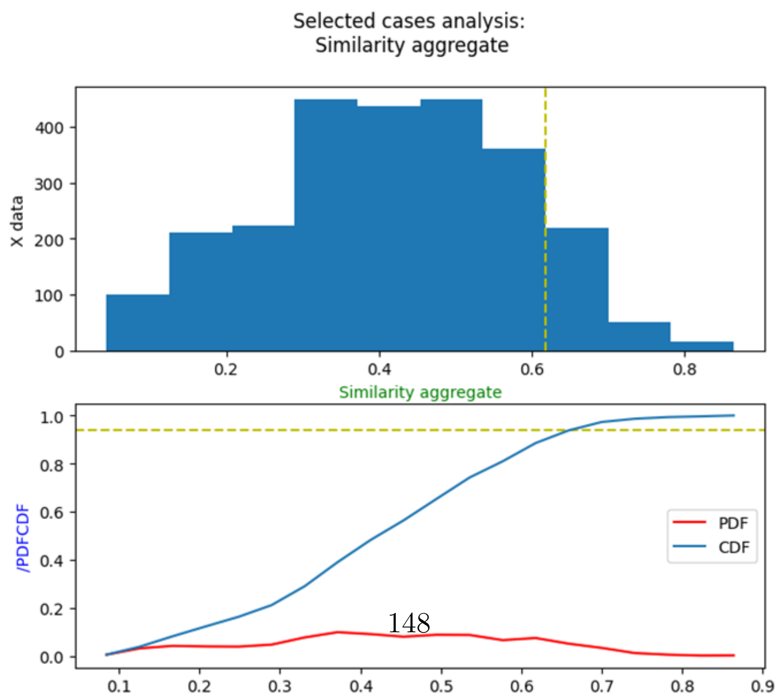

Figure 6.4: Similarity aggregate graphs for case 22(noisy)

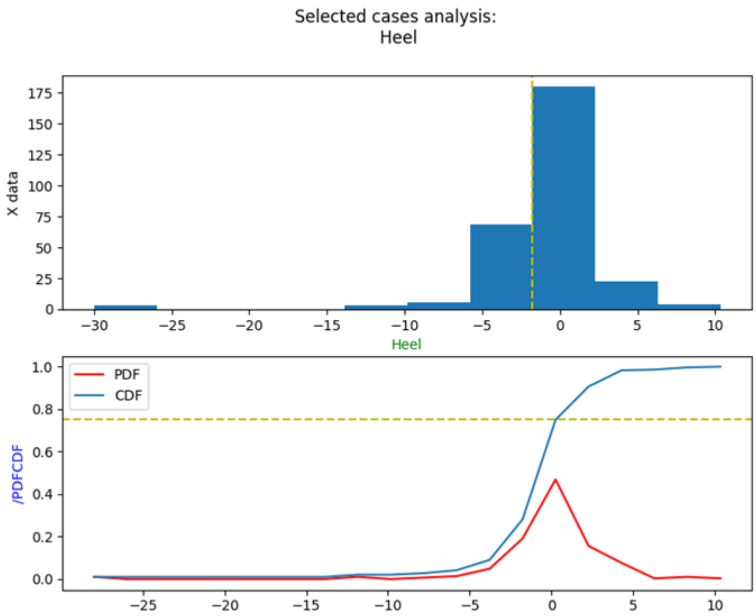

Figure 6.5: Heel of selected cases for case 22(noisy)

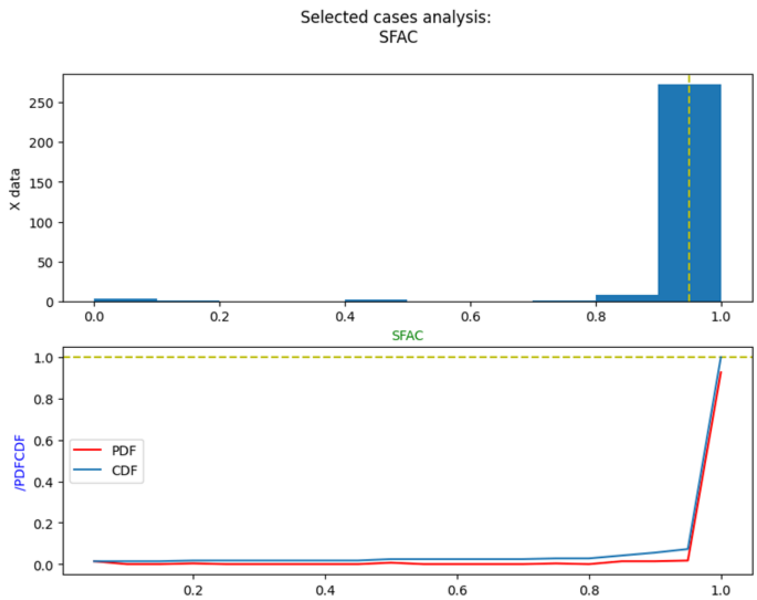

Figure 6.6: SFAC of selected cases for case 22(noisy)

Below is the same approach for another case that is marginal or has lower SFAC and higher heel etc.

6.2.3 Case 2291

Case 2291 is a marginal case for survival where SFAC is 0.7306 and heel is 10.16 deg. This represents a significant damage evolution, and it is expected that the code to be equally able to capture that. Below is the printout when passing the case with no noise.

Table 6.7: Predictions for Case 2291

Prediction Type	**Value**
Simple average	0.64
Similarity weighted prediction	0.65
Posterior probability weighted prediction	0.88

Table 6.8: Most Likely Cases for Case 2291

Case	**Similarity Score**	**SFAC**
2291	1.00	0.73
4842	0.79	1.00
3523	0.78	0.98
7040	0.77	1.00
9840	0.72	0.50
3534	0.70	0.70
6063	0.70	1.00
6133	0.68	0.63
1743	0.66	0.56
9236	0.64	0.50
Weighted average SFAC of 10 most likely cases		0.77

It is immediately obvious that the selected cases have indeed lower SFAC values roughly in the neighbourhood of the expected true value (0.64-0.88). The selected case

2291 is found at the top of the similarity ranking as expected but the rest of the top 10 cases are a mix of survival and marginal (s¡1) cases. The average of the top 10 is quite close to the real value. The graphs are also plotted:

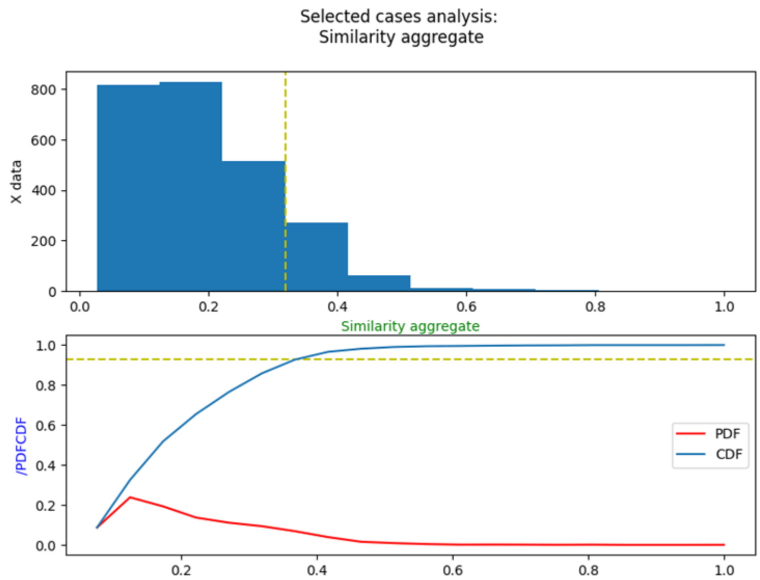

Figure 6.7: Similarity plot for case 2291

The similarity distribution is quite different to case 22. Very few cases have high rankings. This is possibly explained by the database not having a regular distribution of cases. As an effect of that, the similarity cut-off is quite low hence cases are included with lower scores.

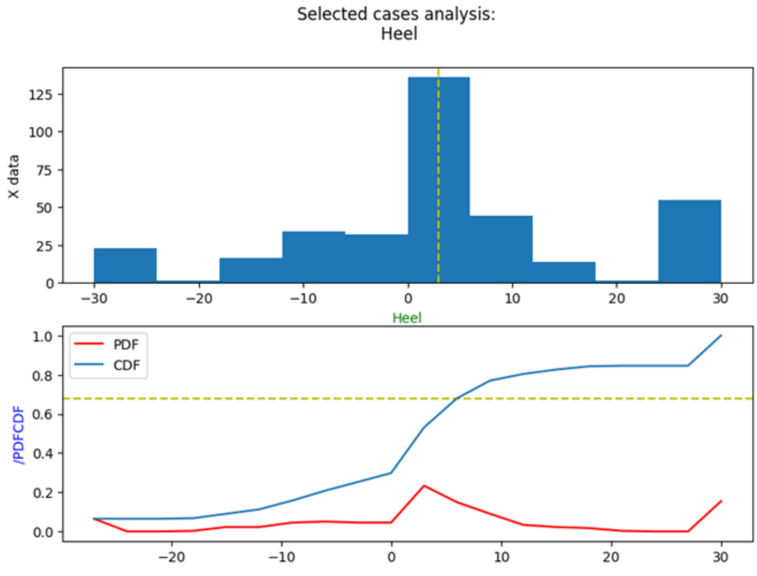

Figure 6.8: Heel distribution of selected cases

The heel distribution has a mean of 3.4, which is quite lower than the true value. However, it is clear that the heel is positive and definitely larger than about 3 from the distribution. A substantial amount of 30(capsize) heel angles are also shown.

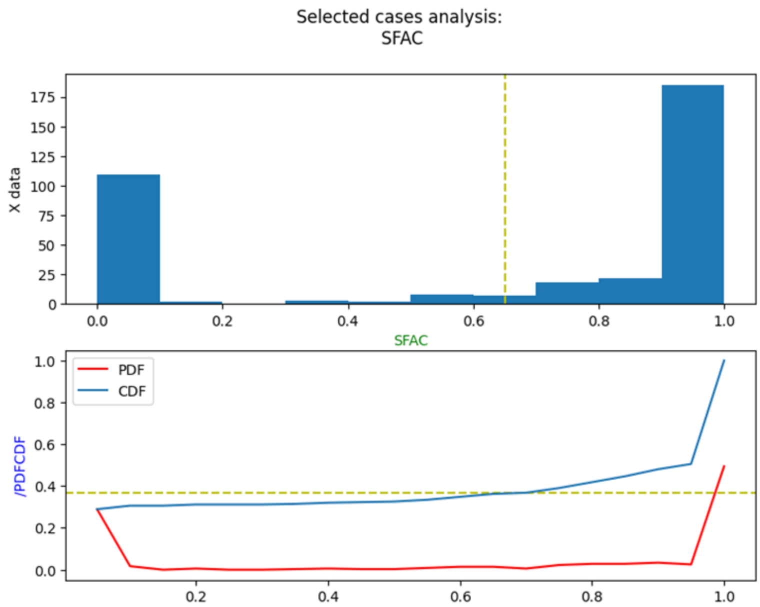

Figure 6.9: SFAC graph of selected cases for case 2291

Finally, for SFAC there are two peaks at 1 and 0 and some cases in between. It is clear that the average is mostly affected by the relationship between zeros and ones. In any case, there are clear indications of a case that can go either way, having equally large probabilities of being a survival or capsize case.

Table 16. Summary of predictions case 2291

Table 6.9: Summary of predictions case 2291

Source	SFAC	HEEL
Truth	0.7306	10.16
92nd perc. average	0.64	3.4
Similarity weighted	0.65	3.9
Probability weighted	0.88	1.61

6.2.4 Case 2291 with noisy data

The output when noise is introduced is shown below:

Table 6.10: Predictions for Case 2291 (Noisy)

Prediction Type	**Value**
Simple average	0.68
Similarity weighted prediction	0.69
Posterior probability weighted prediction	0.9

Table 6.11: Most Likely Cases for Case 2291 (Noisy)

Case	**Similarity Score**	**SFAC**
2291	0.90	0.73
4842	0.80	1.00
7040	0.76	1.00
9840	0.75	0.50
8523	0.70	0.98
6063	0.70	1.00
8534	0.69	0.70
7508	0.65	0.91
4673	0.63	1.00
6133	0.62	0.63
Weighted average SFAC of 10 most likely cases		0.84

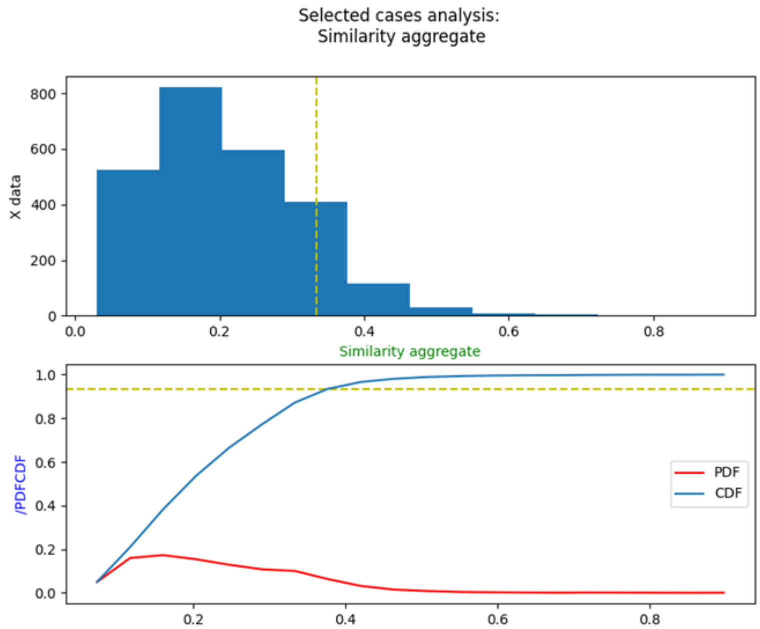

Figure 6.10: Similarity aggregate for case 2291(noisy)

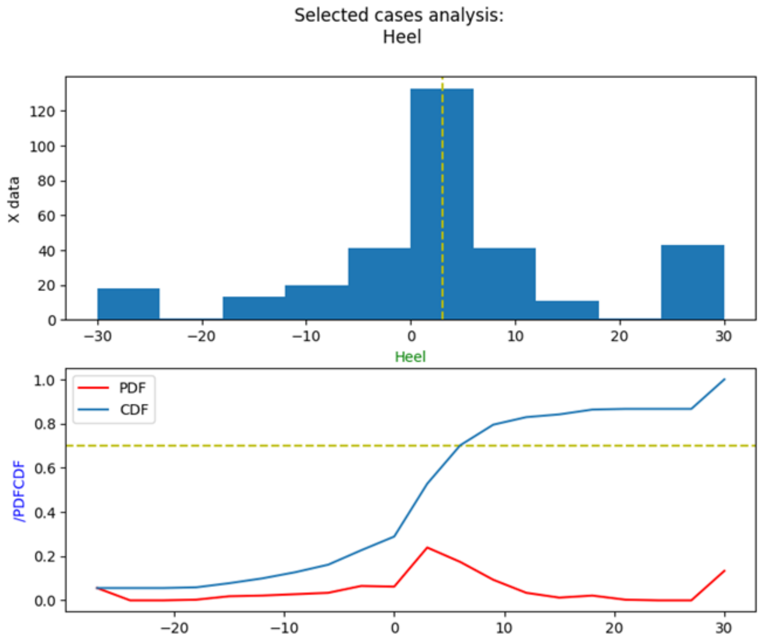

Figure 6.11: Heel distribution of selected cases

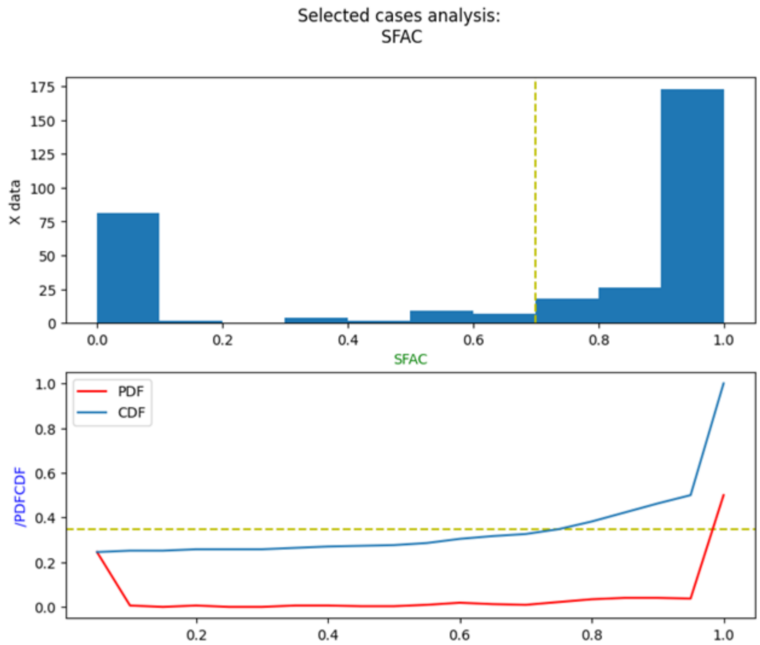

Figure 6.12: SFAC of selected cases for case 2291(noisy)

The noise introduced does not really change the extracted information. A further increase in noise still does not change substantially the results.

Table 6.12: Predictions for Case 2291 (Noisy)

Prediction Type	**Value**
Simple average	0.66
Similarity weighted prediction	0.68
Posterior probability weighted prediction	0.9

Table 6.13: Most Likely Cases for Case 2291 (Noisy)

Case	**Similarity Score**	**SFAC**
2291	0.72	0.73
8523	0.70	0.98
4842	0.66	1.00
4673	0.56	1.00
9840	0.56	0.50
8534	0.55	0.70
7508	0.55	0.91
1743	0.55	0.56
5534	0.54	0.77
7040	0.52	1.00
Weighted average SFAC of 10 most likely cases		0.82

6.2.5 Discussion

All capabilities of the created methodology and Python have been discussed and analysed extensively. The approach requires a larger database to capture all cases- in some cases, due to a limited database the accuracy is reduced;however, the methodology can provide quick, relatively accurate and verifiable data points that can then inform a larger process of decision support. An investigation of the uncertainty sources has also revealed that between the model accuracy, data available, and expected input uncertainty, the method is robust with high accuracy as well as other performance metrics, and a demonstrably small effect from input uncertainty. The multiple sources of information can -with high accuracy give information regarding the severity of the damage. In most cases, this is the most important information missing from the mental picture of the decision maker. Specifically, it has been demonstrated how the database can be traversed and analysed for similar cases to the one described by a sensor or other data. This CBR approach can then make educated guesses about the future of the case by looking at similar cases stored in the database. This is also true in the machine learning approach where models are trained and calibrated to produce predictions given the selected inputs. Here we can also capture trends and relations between variables and extrapolate and interpolate far better than we can in the CBR approach. Model accuracy is of course necessary to trust the result. In the end, both streams of information can be presented and the human operator or some other algorithm can select the most probable range considering the sensibilities and cost of a wrong estimation. For example, a prediction of false survival should be avoided even if that means mislabeling a survival as a non-survival case. Finally, the expected high-level information required from the tool is the expected time to capsize or TTC. This requires ample time-domain simulation results which is a very time-consuming process. Alternatively, and as demonstrated the missing data can be filled in with predictions by utilizing the quasi-static data. Even though this approach introduces a large amount of inaccuracy, the predictions are usable even for this small sample set. As TTC is a very important variable for the rest of the components it is wise that a conservative estimation is passed on. For example, the mean average error can be subtracted to have higher confidence that the prediction is not underestimating the importance.

Chapter 7

Uncertainty & validation/verification

7.1 Introduction

As evident from the procedure of getting the predictions, it is not an exact science. The procedure is created to provide real-time predictions of a very complicated phenomenon that would take hours to properly simulate with current technology. Hence, any output should be regarded as a statistical derivation of what is probably happening. At many points, assumptions, statistical distributions, and uncertainty infiltrate the prediction. Major sources of uncertainty are:

- Sensor accuracy & resolution (level monitoring vs immersion detection)
- Database resolution & Accuracy
- SOLAS p factor
- Similarity scoring

- Model training & accuracy (e.g. overfitting)
- Damage assessment (human or combination)

To begin with, the information fed into the system can be erroneous. In fact, it will most likely be. Also, due to discretizing the variables, any value will be rough rather than exact. Human and technical means can err, either providing false positives or negatives or drifting from the correct value.

The database does not have infinite cases, so any potential case is most probably an interpolation of similar cases. The closer the similar cases are, the better the results. Also, the evaluation of the cases in the database has its own assumptions and inaccuracies that are carried over in the process.

The a priori distribution of damages used here to create a more informed weighted average is a statistical observation. This means that on average, it will point to a more correct prediction. On a case-by-case basis, though, it might lead the methodology to other more probable cases if encountering a rare one. This does not detract from the methodology but should be kept in mind when evaluating the results for low p-factor cases. To combat this, multiple predictions are given as output, see 6.1. More on those predictions will be presented later in the discussion chapter.

The potential of the similarity scoring to compare the emergent case to the case base is also inexact and bases its functionality on the assumption that similar cases will have similar results, which are generally valid. Using Jaccard similarity as a tool to evaluate similarity is an appropriate metric; it is, however, one of many available. The similarity function for the damage variables is quite arbitrarily defined, and the similarity functions used within should be explored further. As a result, the algorithm may promote cases that do not really have similar outcomes.

To combat this uncertainty, the use of statistical distributions is helpful. For example, the fact that the final prediction (weighted average) considers the top cases instead of only the top case allows for better reliability even if the prediction is not that accurate. By adding a range of cases selected based on their similarity, the prediction becomes more conservative. If the simple average and the weighted average predictions coincide, then we can be certain that the prediction will be accurate, as basically all the probable cases share the same outcome. If the average is significantly lower than the weighted average, that means that while the most probable cases have high outcomes, the neighbouring cases of lower similarity have lower outcomes, which attaches a higher uncertainty to the prediction.

The overall certainty of the prediction can also be qualitatively informed by the Probability density distribution of the similarity aggregate. If, for example, few cases have high similarities, and the majority are at the lower end, then that signifies that there are no similar cases found. On the other hand, if there are few cases with very high similarity, then we can be certain that the emergent case is somewhere close to those cases.

Finally, the weighting of the room similarity and damage similarity can be adjusted to optimize the selection of similar cases. The room similarity is a very high-quality source of information as it is quite specific, and the exact permutation of rooms flooded is quite characteristic of a case. However, it is also very hard to obtain in real life, with sensors being placed in a portion of the rooms and the crew being unable to consistently find all flooded rooms. For the examples above, the weighting factor is 60% room and 40% damage similarity for the aggregate.

Uncertainty is also introduced in the largely opaque workings of the several data analysis and machine learning process that are implemented to produce the predictions. In short, predictions can never be 100% accurate as errors and generalisations are added in various steps, from selecting and training a dataset to calibrating and choosing a discrimination threshold.

Lastly, in the context of a form that is directly applicable to contemporary vessels, a large amount of information is either directly or indirectly supplied by human observers. The humans in this case are the members of the crew that might directly observe a damage variable such as the length or location of a breach, infer it from situational awareness (if sailing close to shore and feel the ship colliding with something it is most likely a grounding event in the absence of other ships nearby) or piece together from observing other sensors (cameras, water level sensors or other human observations). Multiple cognitive biases have routinely been associated with human understanding and assessment. For example, confirmation bias may cause a person to discard information that doesnt confirm ones view. Many accidents have been attributed to this effect. But also, in simpler terms, the assessment given by humans will always be rough and inconsistent. The only way to quantify this effect is to assume that all human-derived values (also applicable for sensor data) are in fact a distribution usually centred around the given data point.

In the following chapter, several small studies are undertaken to better understand the "performance" of both methodologies including validation and uncertainty quantification. The results of these studies are necessary to better understand how to use the predictions in terms of a decision support tool.

7.2 Uncertainty quantification for input variables for machine learning methodology

As discussed above, a sensitivity analysis is required to determine the effect of the uncertainty of the input variables in predicting the variables needed for the task at hand. This analysis is focused on the damage variables, such as the length and location of the breach. The uncertainty of the measurement of those variables is quantified as a normal distribution around the measurement. The standard deviation is different for each variable and depends on its inherent characteristics- the damage variables are discretized usually in a range of integer numbers to facilitate human input. For example, for the location of the damage, the ship is divided into tenths. A measurement of 4 would generate a normal distribution with a mean of 4 and a standard deviation of 0.7. The deviation should capture the probable measurements we can expect from a person given the observed value. For length, the 1.5 corresponds to 87% of the measurements of a normal distribution. The deviations are chosen to appear logical, but no specific data is used in their derivation. Below are some random examples. Note that due to the discrete nature of the values(integers), the density bars do not follow a normal distribution as a result.

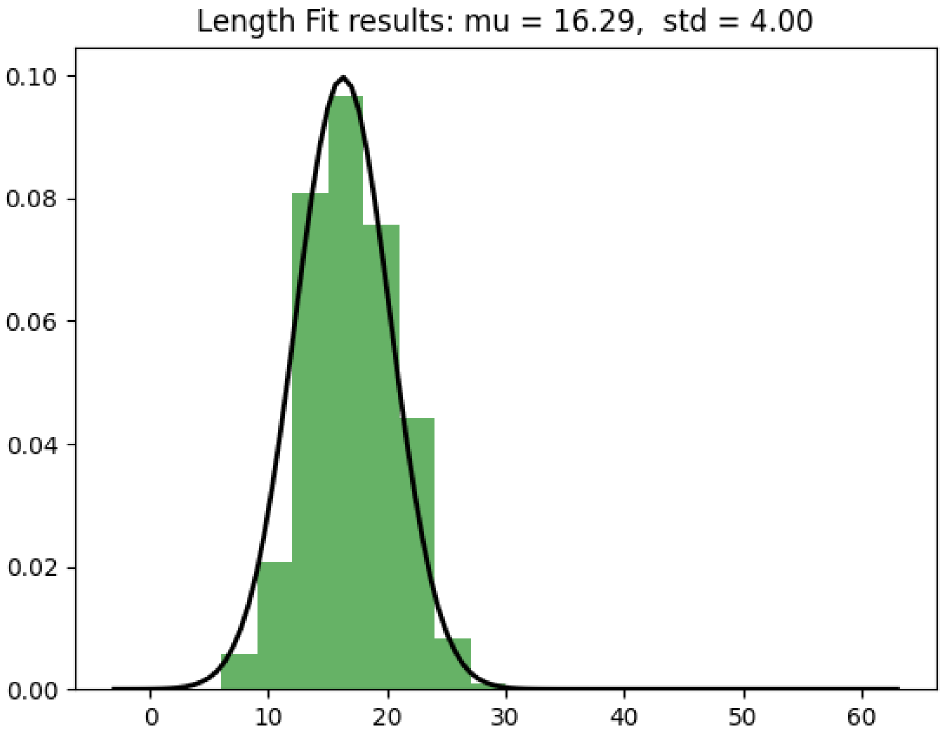

Figure 7.1: Length variable distribution

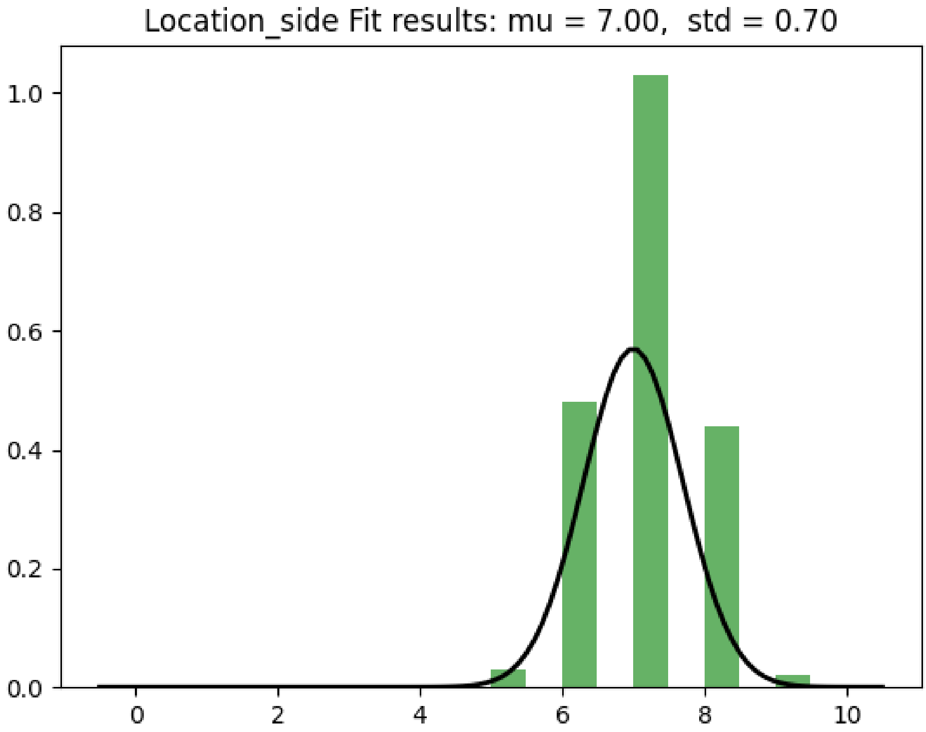

Figure 7.2: Location including side (PS neg) variable distribution

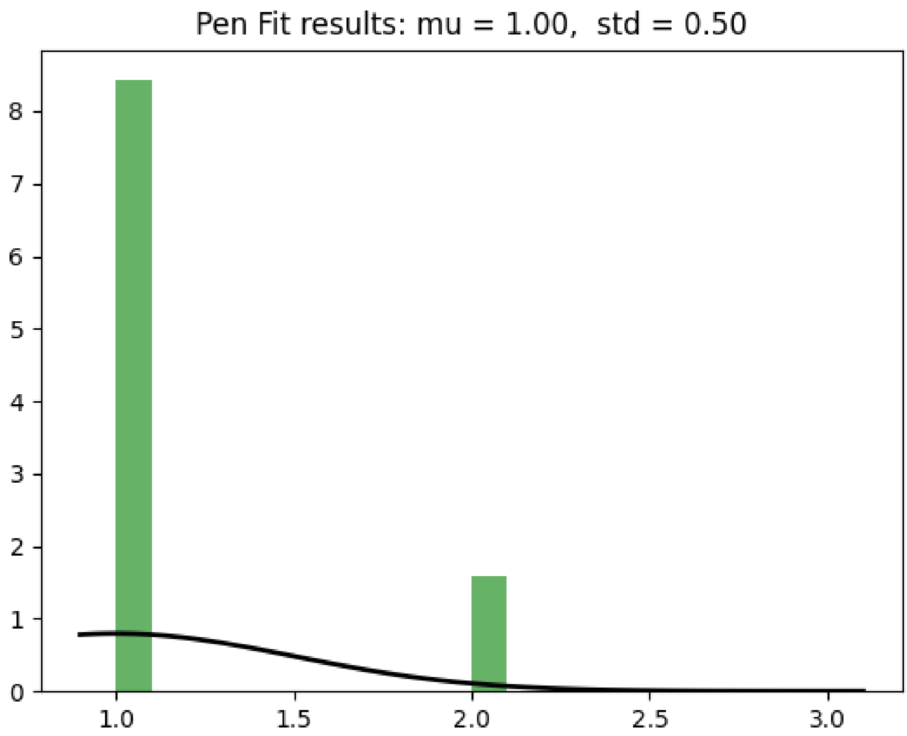

Figure 7.3: Penetration variable distribution

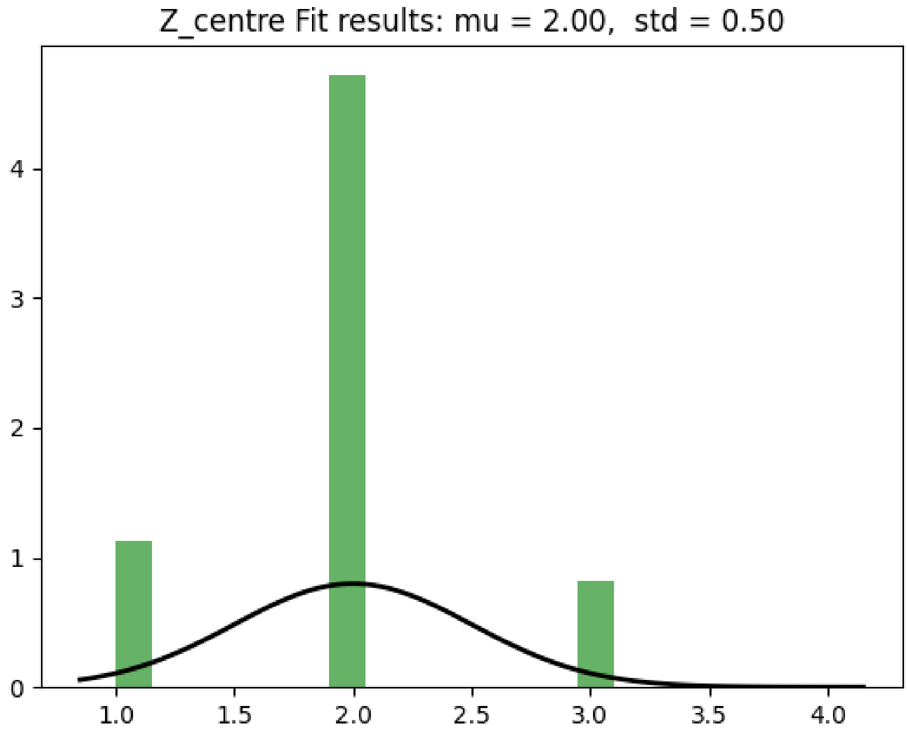

Figure 7.4: Height of centre of damage variable distribution

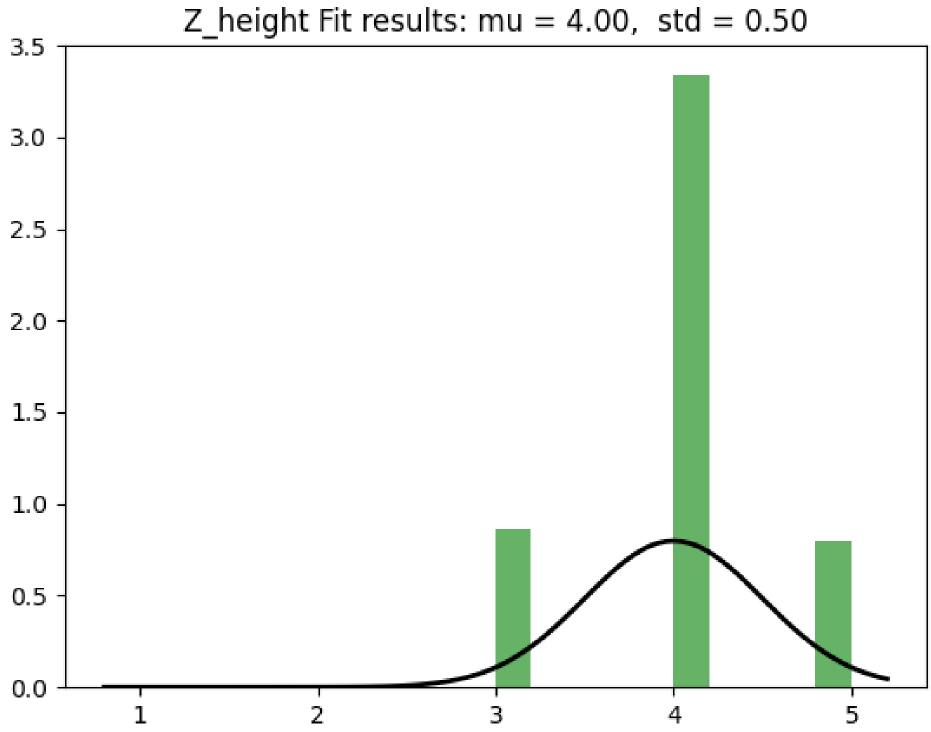

Figure 7.5: Height of damage variable distribution

400 instances for each variable are then chosen randomly and shuffled into a list of cases. Each of those cases is then evaluated with the machine learning models. The original case(without any random instances) is also evaluated and called the exact. The distribution of the 400 predictions and the exact prediction can then be plotted and analyzed. The above process is basically a Monte Carlo simulation.

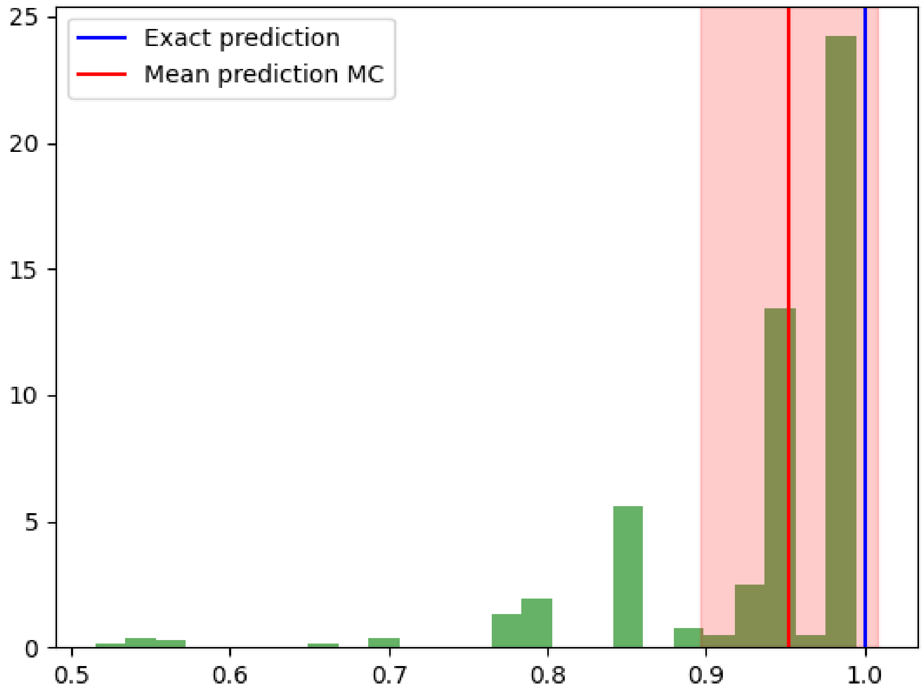

Figure 7.6: Survival factor histogram from 400 different sets of data of random cases. The red shaded area contains the mean plus or minus one standard deviation.

In the above example, it is immediately obvious that most data are found between 0.9-1.0. This space also contains the exact value. The spread of the density of the data suggests the uncertainty of the prediction. The standard deviation is also another indicator of spread. In the above case, it can be thought of as low. The mean and exact predictions are also close, meaning that the exact prediction has high robustness for this case or is affected relatively little by the uncertainty in the input data.

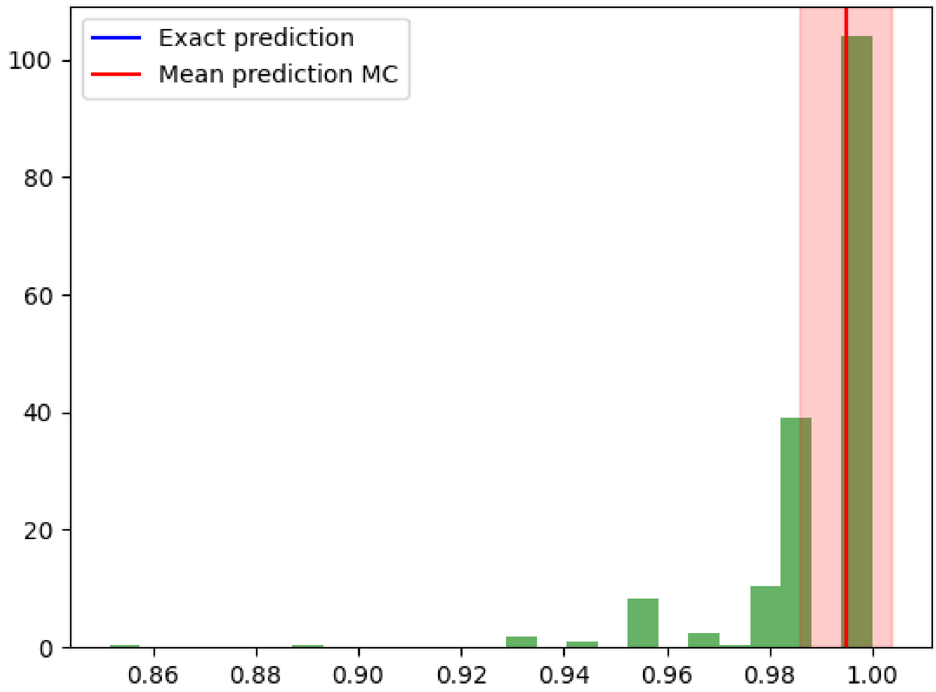

Figure 7.7: Survival factor histogram from 400 different sets of data of random cases. Example 2

In this case, the spread is even lower, and the mean and exact predictions coincide even more. The shaded area is also narrow, meaning that uncertainty is low.

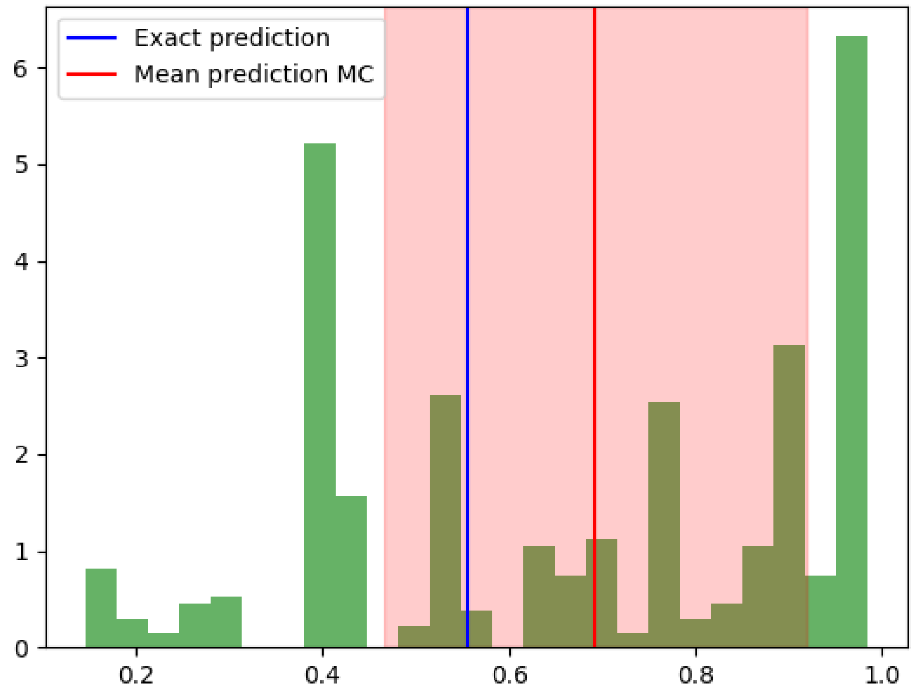

Figure 7.8: Survival factor histogram from 400 different sets of data of random cases. Example 3

The above figure is an example where there is a large spread of values for the prediction. This variance suggests high uncertainty in the prediction as well. The shaded area has a large range as well. In this case, uncertainty is high in the prediction. Note that the above examples have been artificially modified to exaggerate the differences in low/high uncertainty. It is obvious that each case can have a different uncertainty associated with it. In order to evaluate the general effect that input variable variation can have on the end result, a Monte Carlo simulation is carried out. The difference between the exact and median value is computed for 200 random cases, where each case is evaluated 400 times with the input data distributions. The standard deviation is also kept. The purpose of this is to generate a histogram or a PDF showing how the end prediction is affected across the cases of the database. The prediction is done through the ML models generated and calibrated according to the previous chapter. This process is quite computationally intensive, as there are a total of 80200 predictions that need to be made, as well as the generation of random input variable tables. The resulting data is plotted as histograms to better visualize the density of the shift.

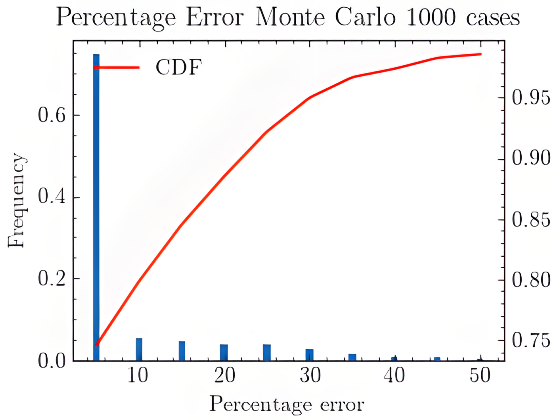

Figure 7.9: Error between mean and exact prediction from the Monte Carlo simulation for SFAC

From plotting the CDF on the above figure, we can deduce what uncertainty we can expect from the prediction of the input variables. For example, from the above 90% of times the prediction will be up to 15% different from the mean, while for 75% of the times the error will be less than 5%. This, in turn, can be translated into the statement that the methodology presented is robust when considering the uncertain nature of the damage variable input. The input variables have been given as distributions that appear logical for an optical assessment, usually by the crew.

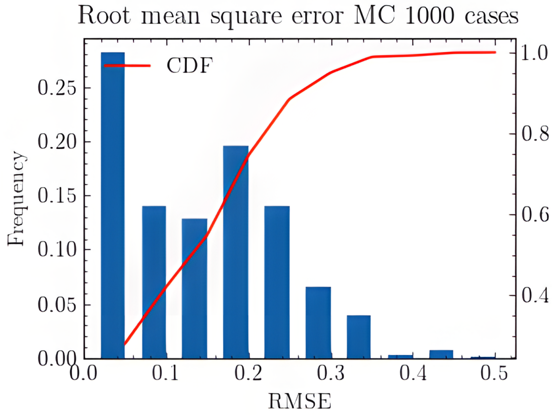

Figure 7.10: Root mean square error of 1000 random cases across 400 permutations each.

It is obvious from the "error" between the mean and exact prediction that the vast majority of cases had a small deviation of the exact value from the mean. In around 65% of cases, there is less than 10% of standard deviation difference between the prediction with the given inputs compared to the mean of the 400 predictions with the distribution of input variables. The error (difference between mean and exact prediction divided by the mean prediction) can also be similarly plotted.

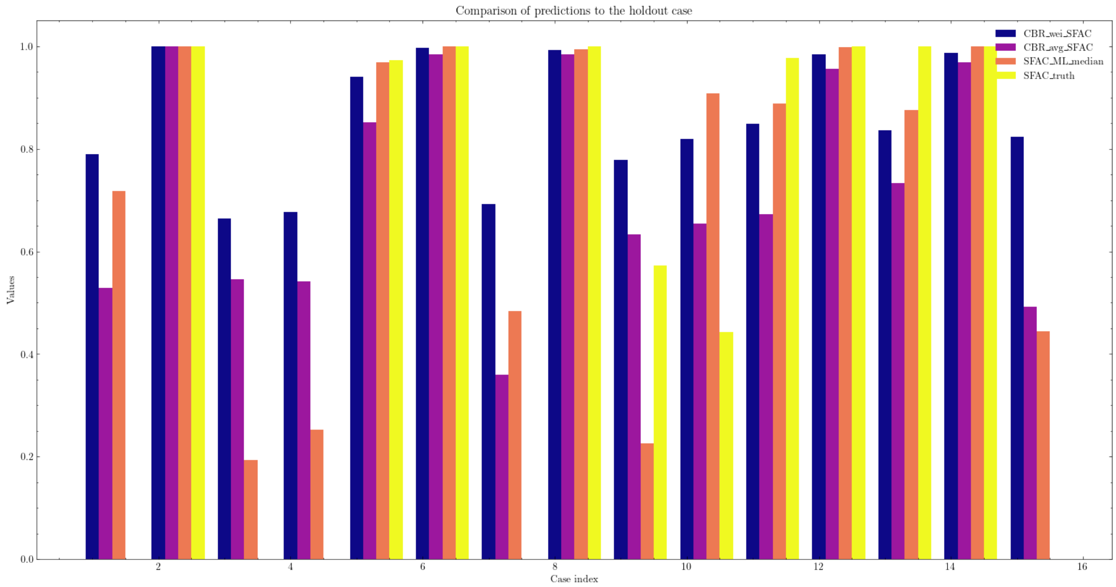

Figure 7.11: Comparison of predictions to the holdout case

7.3 Validation of CBR methodology

The CBR is here tested on its own (disjoint from the ML methodology) to determine whether the individual parts of the methodology work as well as the whole approach itself. The main considerations for this are:

1. **Similarity functionality:** Evaluate if the similarity function accurately selects cases that are similar. This will become evident by the spread of predictions in cases that are not marginal; in marginal cases, there might be a large spread of cases' outcomes, but for cases that are firmly either 1 or 0 (in terms of survival factor), these should be grouped together.

2. **How input noise influences the CBR predictions:** Ideally, and due to the nature of CBR, it should be fairly resilient to input uncertainties since the results

are evaluated in the context of statistics of similar cases.

3. **Deriving the correct outcome from each prediction** might be more involved than simply using the predicted survival factor. There is also an important distinction; the predicted (mean or weighted) survival factor is not the expected one but rather the average value of the predictions. In a situation where half the predictions are 0 and half are 1, the resulting 0.5 is not an accurate prediction for the survival factor but rather an indication that there is little certainty as to the result. In these cases, and considering the nature of a safety system and human nature, it would be prudent to err on the side of caution and declare that case as very dangerous and possibly leading to a sinking. The 0.5 level might be too low already to discriminate, and maybe a higher one is more worthwhile, considering also the decision support implications.

7.3.1 Monte Carlo simulation

To appreciate the "behaviour" of the CBR methodology vis á vis the input uncertainty, a Monte Carlo simulation is set up using the same main code functionalities but with separate functions that choose a random or selected case from the database and remove it from the database for further steps. This simulates the methodology of seeing a case for the first time. Later, the process of generating "dummy" inputs for that case takes over and generates a preselected number of "noisy cases", cases that start with the selected case's data and add random noise to those (ensuring no replication). In the following example, the number of generated "dummy" cases is set to 500. Due to the extent of the database that needs to be evaluated, each time the time to run this simulation is roughly 20 minutes for each case. Each case is then evaluated using the CBR, and the predictions are returned. Using the 500 predictions, the following graphs are plotted.

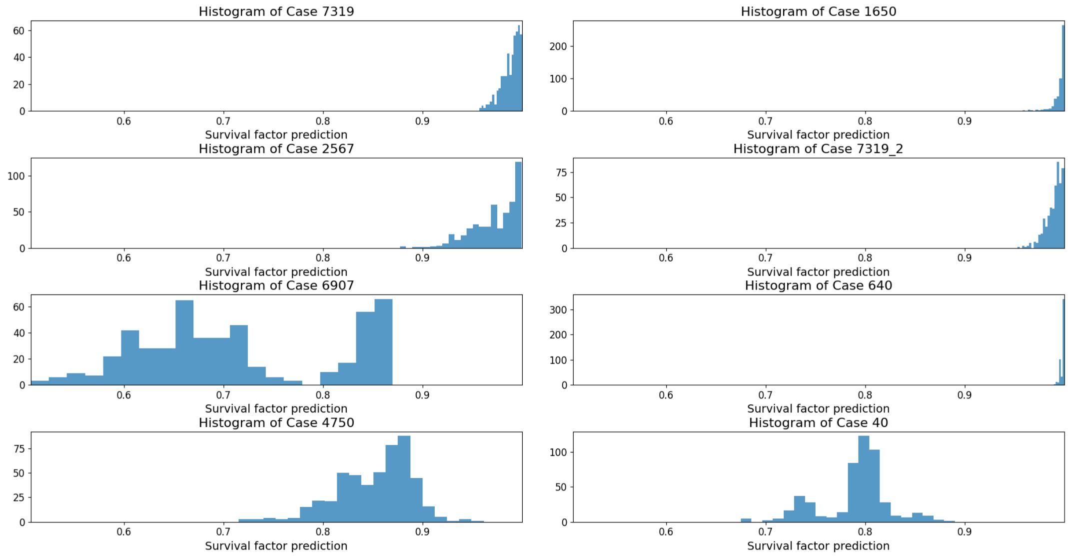

Figure 7.12: Histogram of holdout cases predictions (500 samples)

It becomes immediately obvious that there are several qualitative and quantitative characteristics that can be easily inferred from looking at the values, such as the error bars and median prediction. To begin with, the spread of the Q1 to Q3 range (25th to 75th percentile) is a direct indicator of "uncertainty" or variability in the prediction. A high spread indicates that:

1. With the noise added, the cases have different outcomes, meaning that the case is marginal; with small changes, the prediction can change.

2. Many similar cases are found around this case that have different outcomes, again pointing out that the case is marginal. Also, it can mean that the similarity functionality is not very specific and discriminatory.

Conversely, a low spread indicates that:

1. Even with noise added, the cases around the case share similar results, and thus the prediction is robust.

2. All the noisy cases maintain a high similarity with the holdout case and between themselves so are consistently selected and weighted high, influencing the prediction in having little variability.

3. The case is firmly in survival (1) or sink (0), and any small changes don't produce enough appreciable change to affect the prediction.

The following plot summarizes well-said characteristics.

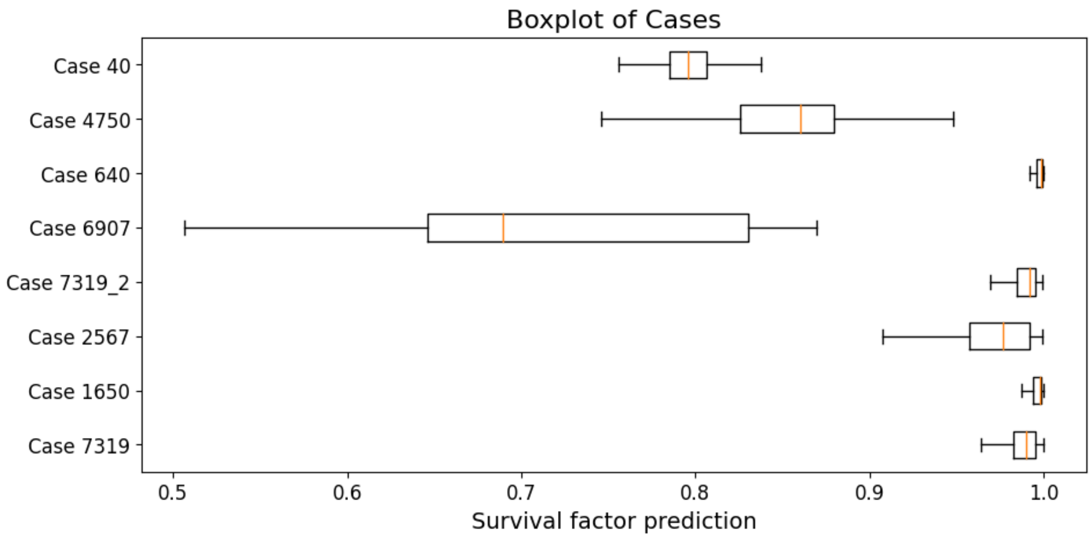

Figure 7.13: Box plot of all holdout cases expressing the distribution characteristics

See below explanation plot below for information on the formulation of the box plot 7.14. Note that 7319_2 is a second iteration of case 7319 to check for appreciable differences due to the relatively small sample size.

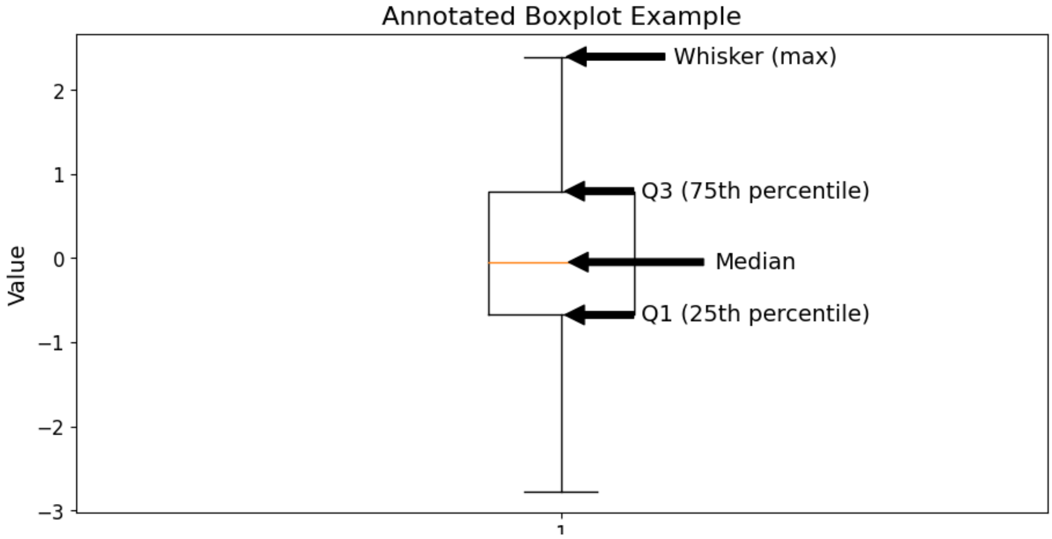

Figure 7.14: Annotated example of box plot components

It is evident here that there are several cases where there are small error bars, cases 40, 640, 7319_2, 1650 and 7319 all exhibit tight predictions close to 1 (survival). This fact reinforces the belief that these predictions are robust, or that CBR provides a robust survival prediction in this case. It is also reassuring to see, since it can only be the effect of the similarity and adaptation functions working properly. In fact, looking at the actual cases below, more conclusions can be made.

BreachIndex	**T**	**TR**	**HEEL**	**GZMAXR**	**SFAC**	**Length**	**Pen**	**Location**	**Location_side**	**Z_centre**	**Z_height**
640	6.23997	0.18032	0	0.51477	1	0.01369	1	180.12805	-9	3	5
1650	6.22798	-0.18835	2.13264	0.35936	1	7.66593	1	18.16383	-2	3	5
7319	6.51690	1.59422	0	0.67290	1	10.87482	3	167.09214	-9	2	4
2567	6.54920	1.08129	0	0.37348	1	12.28058	1	-130.94363	-7	3	5
4750	9	-9	30	-0.00998	0	35.49819	2	17.65726	-2	3	5
40	9	-9	30	-0.09089	0	46.54622	3	10.67755	-1	2	4
6907	7.58380	3.00687	15.1130	0.21930	0	53.46808	3	108.64028	-6	2	3

Table 7.1: Table of breach data with parameters.

The following observations can be made, first considering the non-survival cases:

1. Cases with 0 survival factor (40, 4750, 6907) are clearly far from the 0.9-1 area in terms of 1 standard deviation.

2. Case 6907 is a 0 sfac case but not resulting in a capsize (Final Heel=15 deg), which is reflected in a large standard deviation. This is perhaps expected as that case is close to being a firm survival or sink given the noise and similar cases. Hence, the uncertainty suggested by the error bar has logical roots.

Looking at the rest of the cases:

1. All survivable cases have median predictions firmly in the 0.9-1.0 and further 0.95-1.0 ranges.

2. The standard deviation is very small and lies firmly in the survivable region. This signifies a robust prediction that can be depended upon to draw conclusions for decision support.

7.3.2 Takeaways and Adjustments for Decision Support

The validation of the CBR methodology highlights several critical insights and areas requiring refinement to enhance its utility for decision-making during emergencies. One key takeaway is the robustness of predictions for non-marginal cases, survival cases where survival factors are firmly categorized as 1 (survival). In such cases, the predictions are consistent and reliable, with minimal spread in their distribution. This consistency underscores the effectiveness of the similarity and adaptation functions in identifying relevant cases and appropriately weighting their contributions. Conversely, marginal caseswhere survival factors are near the midpointdemonstrate greater variability in predictions. Usually, anything below 0.9 corresponds to a survival factor of less than 1, but not necessarily 0. This behaviour is expected, reflecting the inherent stochastic nature of the methodology and the challenges posed by input uncertainties and case diversity.

Also, the fact that the CBR methodology is a statistical process and most cases in the database are survival cases, then it is expected to see survival cases being included in sinking scenarios, and thus increasing the average survival factor. The weighted prediction here helps, as does the posterior weighted prediction, as it removes any bias towards cases that are low in probability of occurring.

The spread of predictions in these scenarios serves as a useful indicator of uncertainty. Importantly, the probabilistic nature of the methodology supports a safety-first approach. Cases with survival factors near 0.5, for instance, highlight situations of high uncertainty, reinforcing the need for cautious and conservative decision-making.

To further enhance the methodologys alignment with decision support needs, several adjustments can be made. Refining the similarity function to improve its discriminatory power in marginal cases is one potential improvement. By better differentiating cases with subtle but meaningful differences, prediction variability can be reduced, leading to more precise outcomes. Additionally, implementing dynamic thresholding for decision rules, such as defining survival factors below 0.9 as critical, can improve the systems responsiveness to uncertain situations. These thresholds should reflect the operational and safety priorities of the vessel.

To address the discrepancy found between the CBR prediction and reality, and to adjust the range of predictions to coincide better with the truth and ML values, an adjustment is made to how the results from the CBR predictions are used. Specifically, it is shown that the average of the similarity-weighted cases is usually always higher than the expected value. Above, it is evident that even survivable cases are given high averages. A solution here is to reduce the influence of low-similarity cases and amplify the ones from high-similarity cases. Also, this approach can allow for relative similarities to be considered. This adjustment is presented in the following section.

7.3.3 Softmax Weighting for Case-Based Reasoning

In the implemented Case-Based Reasoning (CBR) system, a Softmax weighting approach was utilized to normalize and emphasize the importance of cases based on their similarity scores. This method transforms raw similarity scores into probabilistic weights, ensuring all weights are positive and sum to one. By doing so, the system effectively prioritizes cases with higher similarity scores while retaining contributions from all cases.

The Softmax function is mathematically defined as:

$$w_i = \frac{\exp(S_i)}{\sum_{j=1}^{N} \exp(S_j)},$$

where w_i is the normalized weight for the i-th case, S_i represents the similarity aggregate score for the i-th case, and N denotes the total number of cases in the dataset. The function applies an exponential transformation to each similarity score, amplifying differences between higher and lower scores. The transformed values are then normalized by dividing each by the sum of all exponentiated scores, guaranteeing that the total weight sums to one. This normalization ensures comparability and interpretability of the weights.

In this CBR system, similarity scores are derived from two key components: numerical similarity and Jaccard similarity. These components are combined into an aggregate similarity score, which serves as input for the Softmax function. The resulting weights are then used to compute weighted averages for predicted outcomes. They also allow for prioritization of cases in terms of relevance to the emergent situation, with cases exhibiting higher similarity receiving greater emphasis.

The Softmax weighting approach offers several advantages. By normalizing similarity scores, it ensures comparability across cases and datasets of varying sizes. Additionally, the exponential transformation enhances differentiation, focusing decision-making on the

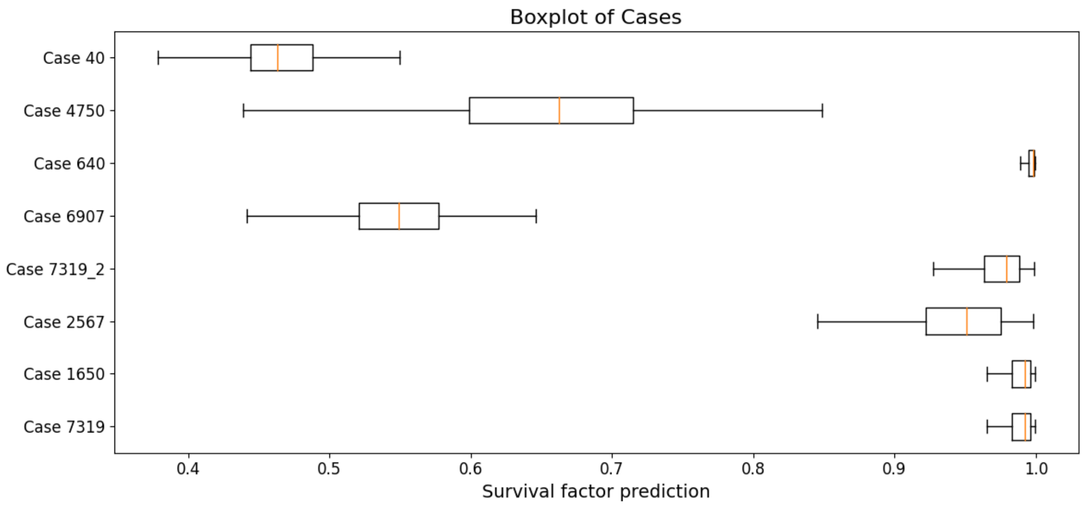

Figure 7.15: Box plot of test cases with 300 samples

most relevant cases while still accounting for contributions from less similar cases. This method also aligns well with probabilistic frameworks, making it ideal for applications requiring robust and interpretable predictions.

The same cases have been used to see the effect of the implementation of the softmax weighting.

Cases 40 and 6907 are decreased in mean value along with error bars. This brings them closer to the true value of 0. Case 4750 is increased in uncertainty and also reduced drastically in terms of its mean, again being closer to the true value of zero.

The cases that were survivable don't change much as they are already very tight around the maximum value of 1. It is expected that the new weighting wouldn't affect them adversely.

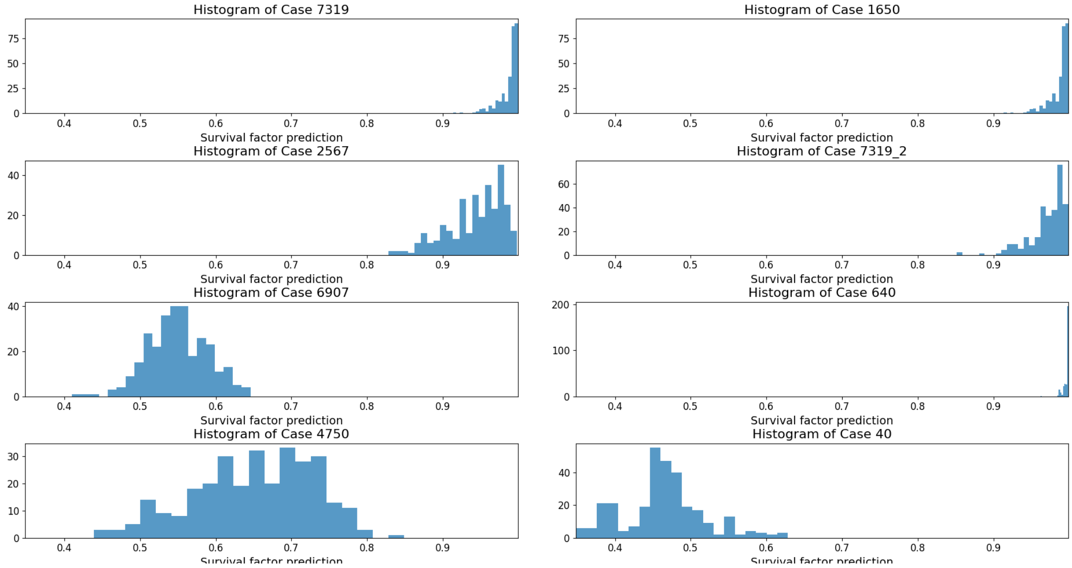

Figure 7.16: Histogram of tested cases prediction distributions

7.3.4 Conclusion

In conclusion, the validation process underscores the CBR methodologys potential as a robust tool for probabilistic damage stability assessment. While its predictions for non-marginal cases are already reliable, targeted refinements in the similarity function, thresholding, visualization, and crew training can further enhance its practicality and alignment with real-world decision-making workflows. These adjustments, coupled with the methodologys inherent safety-conscious design, position it as a valuable resource for improving maritime safety.

Chapter 8

Decision Support System

A decision support system, in the context of flooding emergencies, is a ”tool” or generally framework that aims to support decision-making on board. The major items that this methodology has to include are:

1. An effective and accurate predictor of flooding evolution.
2. Quantifiable uncertainty and risk-averse predictions.
3. Leveraging both methodologies and scalable

8.1 Simple implementation

Initially, it is quite evident what a simple decision support system can look like. In fact, this simple approach has been implemented before for the purposes of the project SafePASS.

In SafePASS, the present methodology(an earlier implementation than the one presented) was used to help tie in multiple information ”streams” and integrate the concept of risk for the passenger onboard. This was achieved in two ways:

1. Information about the casualty evolution, such as roll, GZ, etc. These data integrate with the Bayesian networks developed elsewhere and represent the risk to the passengers in all eventualities, ”simulating” each stage of an evacuation procedure. Data such as roll is used in those Bayesian networks to update the probabilities of some steps. For example, lowering lifeboats with a large roll is more dangerous, as are many other operations such as mustering.

2. The time to capsize itself serves as the ultimate risk, as a capsize event is assumed to bring about the deaths of all passengers remaining aboard. It is also the value that should trigger an evacuation call or not.

Both of these aspects of risk are displayed in the Risk Modelling Tool dashboard (see below) directly (for TTC) and indirectly (with potential fatalities per stage).

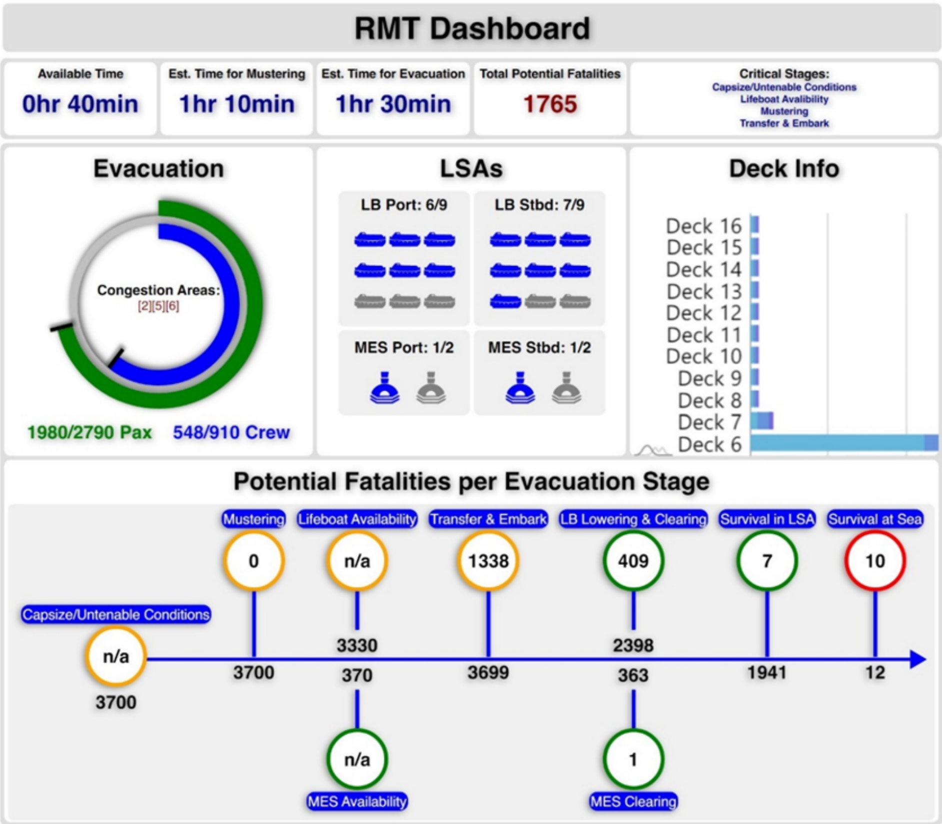

Figure 8.1: SafePASS RMT dashboard with demo data

Note that the potential fatalities at each step are computed based on the probabilities from the Bayesian networks.

Below is the overall structure of the final code as used in SafePASS. Note the individual parts that were presented earlier in this document.

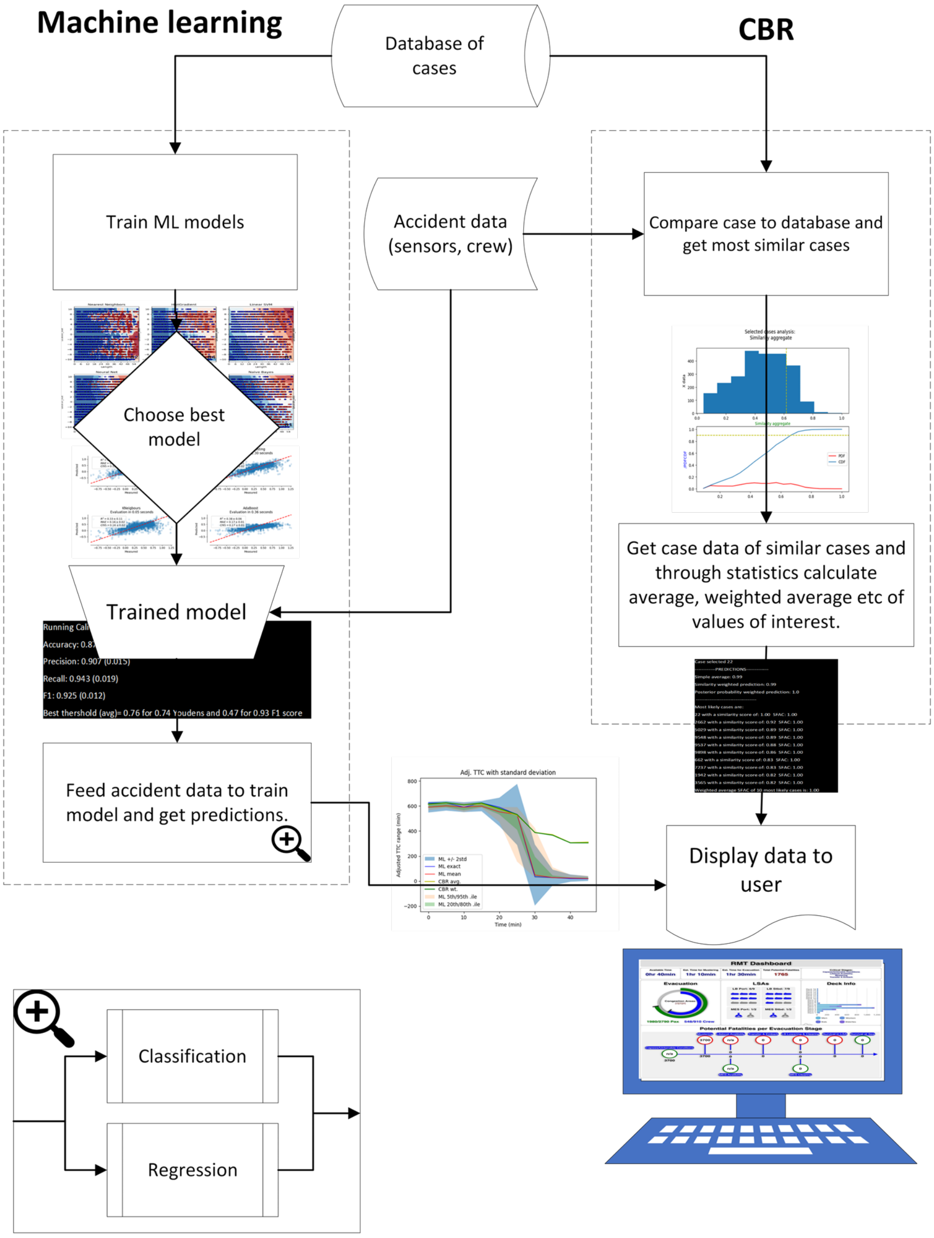

Figure 8.2: Flowchart of methodology as arranged for the implementation in the SafePASS project

To better understand this application, below is a demo scenario that was used to test and understand the performance of the Risk Modelling tool.

8.1.1 Adjusted Time to Capsize (TTC_adj)

Since the Time to Capsize (TTC) is based on data spanning 30 minutes, an adjusted TTC is used to expand the prediction range. This adjustment involves scaling the TTC using exponential factors derived from the predicted survival factor (S_{fac}). The survival factor serves as a modifier to artificially extend or reduce the TTC range:

- A high survival factor (S_{fac}) indicates a safe scenario, where $TTC \to \infty$.
- A low survival factor (S_{fac}) represents a high-risk case, such as sinking or capsizing.

The adjusted TTC (TTC_{adj}) is computed using the following formula:

$$TTC_{adj} = \frac{a \cdot TTC_{ind} \cdot \left(e^{b \cdot S_{fac}} + e^{c \cdot (S_{fac} - 0.7)}\right)}{3600} \text{ hours,} \tag{8.1}$$

where:

- $a = 0.45$ is a scaling coefficient,
- $b = 1.5$ and $c = 12$ are constants governing the exponential adjustments,
- TTC_{ind} is the original individual TTC prediction, and
- S_{fac} is the survival factor.

In this formulation:

- The term $e^{b \cdot S_{fac}}$ increases the TTC exponentially based on the survival factor.
- The term $e^{c \cdot (S_{fac} - 0.7)}$ introduces an additional adjustment, focusing on cases with S_{fac} values below 0.7.

H

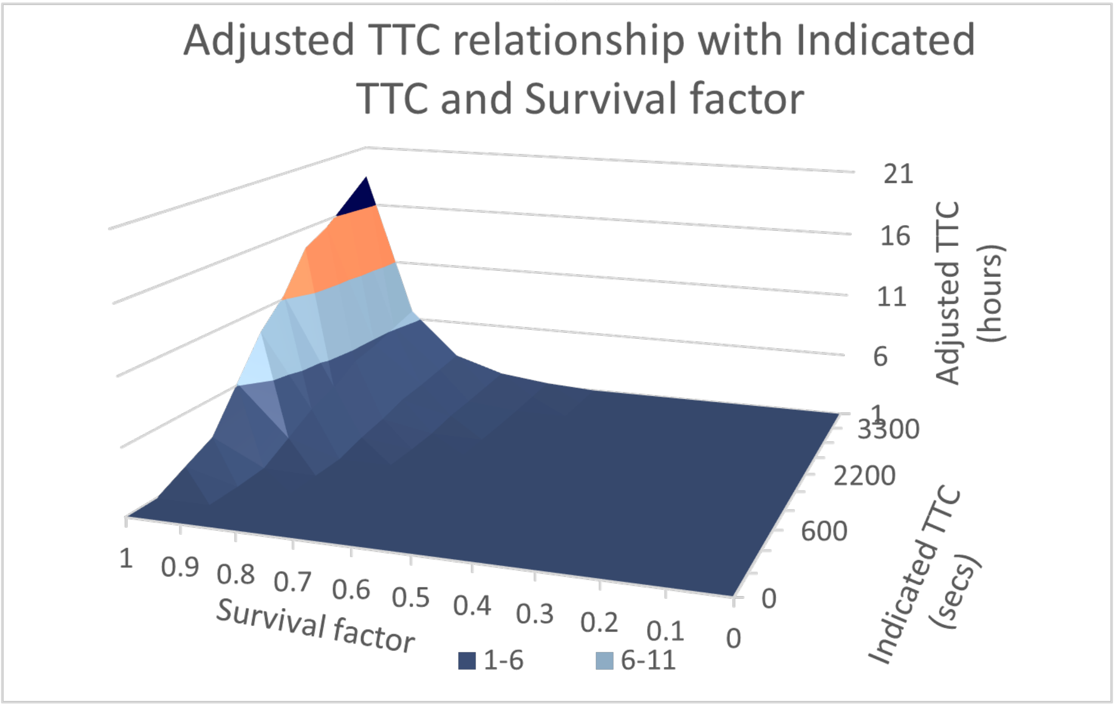

Figure 8.3: Adjusted TTC function 3D map

This approach enables the TTC to reflect a broader range of predictions, accommodating both safe and critical scenarios. The origin of this formulation is not derived in any scientific manner, but rather is an attempt to connect the Time to Capsize to the Survival factor. The use of exponentials allows setting some limits, like the 0.7 that is also used in binarization. In this way, the function exhibits an exponential behaviour in each area (0-0.7, 0.7-1.0). Overall, the underlying assumption is that SFAC of 0.7 and above should be linked with large TTC times (in an exponential manner) until 0.7, where the drop in TTC is very significant instead. In all cases, the indicated TTC is used as the basis of the value and is then augmented or reduced by the exponentials of SFAC. The result can be seen below and conceptually seems valid.

8.1.2 Narrative Summary

The *M/V SafePASS*, a large cruise ship, strikes submerged rocks at high speed during a night voyage, creating a breach in the bilge area that floods the machinery spaces. Initially, the damage is estimated as a 5-meter breach amidships; however, subsequent updates reveal a damage length exceeding 50 meters. The SafePASS system is activated to evaluate flooding risk using crew and sensor inputs.

8.1.3 Sequential Damage Updates

Table 8.1 outlines 10 sequential updates, progressively refining damage variables. By the final update, the damage corresponds to case #6907 in the numerical simulation database.

Table 8.1: 10 Damage Updates Given Sequentially

Index	**Length (m)**	**Pen**	$Z_{\mathbf{centre}}$	$Z_{\mathbf{height}}$	**Location_side**
1	5	1	2	2	-5
2	5	2	2	2	-4
3	5	2	2	2	-6
4	7	1	2	2	-5
5	15	1	2	3	-5
6	21	1	2	3	-5
7	30	2	2	3	-6
8	35	3	2	3	-6
9	40	3	2	3	-6
10	53	3	2	3	-6

8.1.4 Flooding Risk Evaluation

The Case-Based Reasoning (CBR) and Machine Learning (ML) methodologies analyze the damage data to evaluate flooding risk:

- **CBR Methodology**: CBR identifies similar cases and evaluates criticality based on flooded rooms. Predictions converge to lower Time to Capsize (TTC) values as updates refine the damage extent.

- **ML Methodology**: ML uses statistical distributions to evaluate damage variables and predict TTC, survival factors, and heel. Confidence intervals tighten with each update, improving prediction accuracy.

Figures highlight key results:

- Update 7 Predictions: Predictions for TTC at update 7 (Figure 8.16) show increasing likelihood of rapid capsize.
- CBR Case Histograms: Histograms (Figure 8.5) illustrate the shift from high TTC cases to low TTC density as damage updates progress.
- TTC Distribution: ML predictions at update 10 (Figure 8.6) indicate convergence to 20-30 minutes TTC with minimal uncertainty.
- Final Combined Predictions: Figure 8.7 demonstrates agreement between CBR and ML methodologies in the final update.

By update 10, both CBR and ML predictions converge on a TTC of 20-30 minutes. CBR provides conservative estimates leveraging case similarity, while ML evaluates statistical distributions for robust predictions. Together, these methodologies provide reliable situational awareness to minimize the potential loss of life. There is no attempt to connect these two sources of information together, nor their uncertainties, which are tackled in the next chapter.

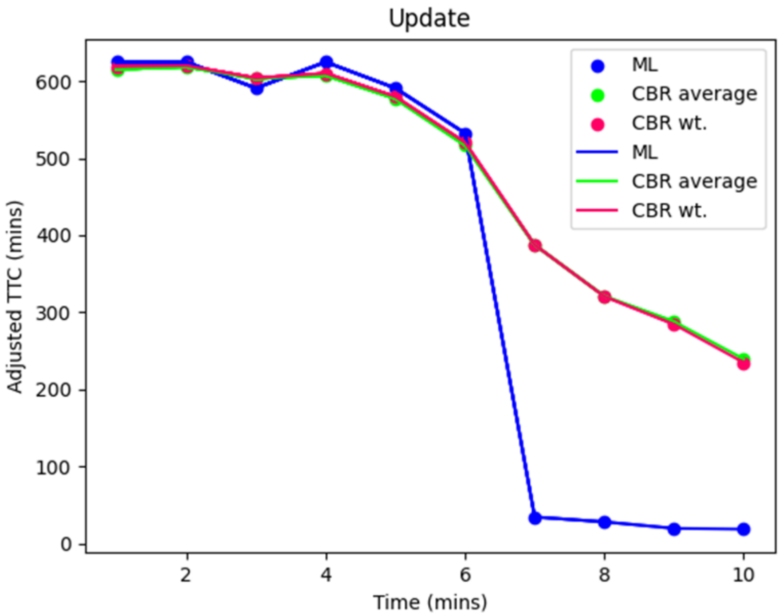

Figure 8.4: CBR and ML Predictions at Update 7

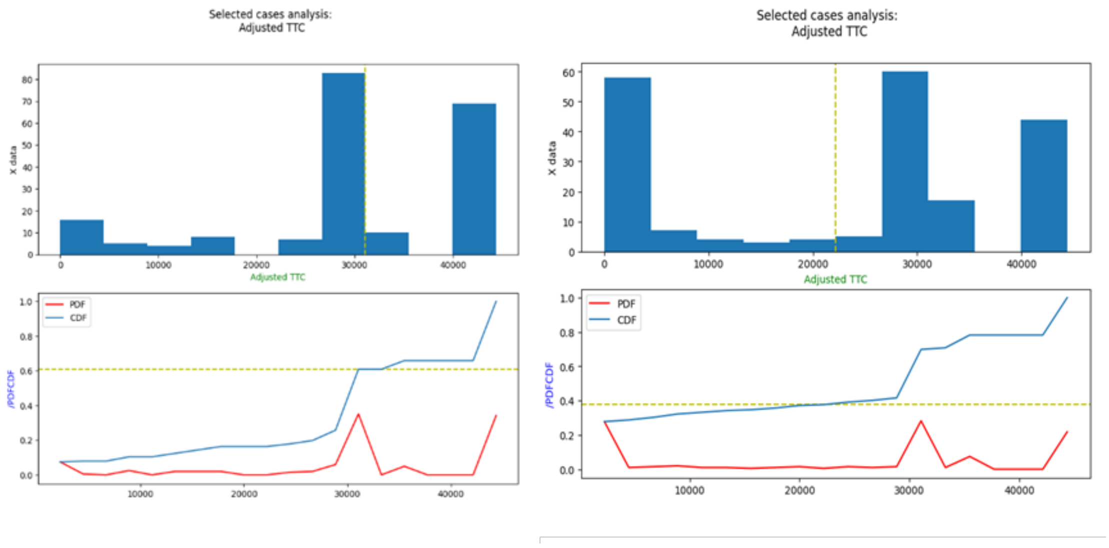

Figure 8.5: CBR Case Histograms

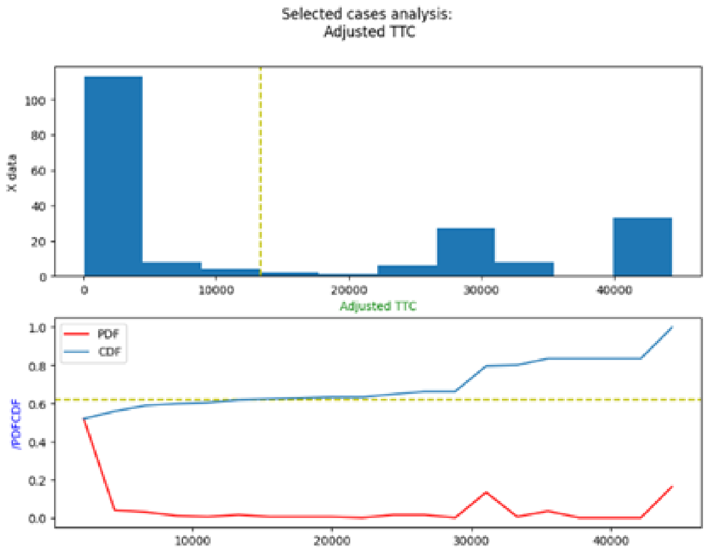

Figure 8.6: ML TTC Predictions at Update 10

8.2 Fusion of Case-Based Reasoning and Machine Learning Predictions

The two methods that were developed attempt to capture the risk of a dangerous flooding event happening, and produce predictions through their implicit mechanisms and a Monte Carlo sampling; CBR and ML methods also produce a distribution of those predictions and consequently quantify uncertainty (to an extent). The mean prediction, as well as the distribution of the predictions, both contain information that can be combined to better tackle the task of providing predictions that are accurate and of quantifiable certainty.

To achieve a ”combination” of all the data sources, a fusion technique is needed that will also take into account the characteristics of the distribution of predictions for each methodology. The simple form of weighted averaging of the predictions is one that can

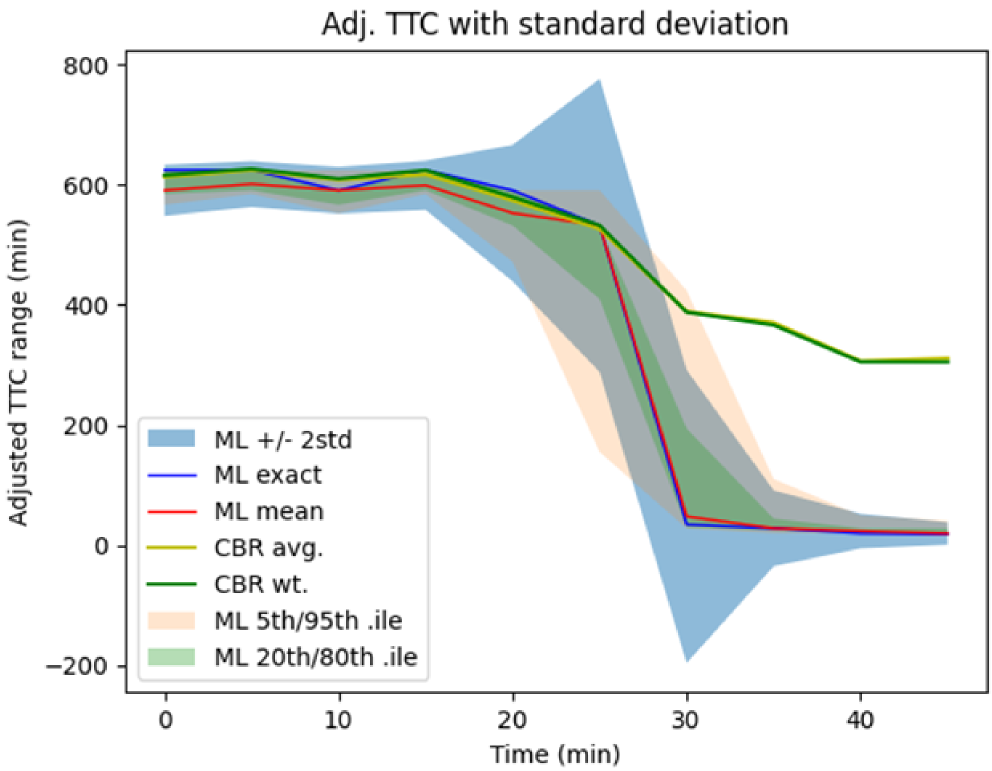

Figure 8.7: Final Predictions from CBR and ML

be used, for example, but this would ignore the attributes of the distribution.

Two methods are used for combining the two methodologies.

1. Bayesian inference

2. Dempster-Shafer

8.2.1 Overview of the Bayesian Framework

Bayesian inference provides a principled method for fusing two predictions by treating one as prior information and the other as evidence. The fusion process integrates these components into a single, updated prediction, accounting for uncertainties in both sources.

8.2.1.1 Components of Bayesian Inference

Bayesian inference relies on the following components:

- **Prior ($P(H)$):** The initial prediction or belief about a hypothesis H, before incorporating additional evidence. For instance, the prediction from a Machine Learning (ML) model can serve as the prior.

- **Likelihood ($P(E|H)$):** The probability of observing evidence E (e.g., the second prediction from a Case-Based Reasoning (CBR) system), given that hypothesis H is true.

- **Posterior ($P(H|E)$):** The updated prediction or belief about H after considering the evidence E, representing the fused result.

- **Normalization ($P(E)$):** Ensures that the probabilities sum to 1 across all possible hypotheses.

8.2.1.2 Fusion of Two Predictions

To fuse predictions from ML and CBR using Bayesian inference:

1. **Represent Predictions as Probabilities:**

    - The ML prediction is taken as the distribution of the random samples passed through the Monte Carlo simulation by adding noise to the case given through the noise generator function using the damage parameter functions (see 7.1, 7.4, 7.5, 7.2)
    - The CBR prediction is also taken as a distribution by using the similar cases weighted by the similarity as other predictions, and thus getting a standard deviation for the assumed normal (Gaussian) distribution.

To integrate the predictions from the Machine Learning (ML) and Case-Based Reasoning (CBR) methods, along with their associated uncertainties, a Bayesian weighted averaging approach is employed. This method ensures that predictions with lower uncertainty (smaller standard deviation) contribute more significantly to the final combined prediction. In Bayesian terms:

- **Prior**: The ML prediction represents the prior knowledge.
- **Likelihood**: The CBR prediction acts as evidence, modifying the prior belief.
- **Posterior**: The combined prediction (mean) and uncertainty (standard deviation) are calculated based on the weighted contributions of the prior and likelihood.

8.2.1.3 Simplification Assumptions

1. Both predictions (ML and CBR) are modelled as Gaussian distributions.
2. Their uncertainties (standard deviations) are known and used as weights.

3. The predictions are independent, so their uncertainties combine reciprocally (as in the variance addition formula).

8.2.1.4 Extracting Predictions and Standard Deviations

The individual predictions and their standard deviations are extracted as follows:

- Predictions: $\mathrm{ML\_TTC}, \mathrm{CBR\_TTC}, \mathrm{ML\_SFAC}, \mathrm{CBR\_SFAC}$
- Standard deviations: $\mathrm{ML\_TTC\_std}, \mathrm{CBR\_TTC\_std}, \mathrm{ML\_SFAC\_std}, \mathrm{CBR\_SFAC\_std}$

8.2.1.5 Combining Predictions Using Bayesian Weighted Averaging

The combined prediction is calculated as a weighted average of the individual predictions, where the weights are inversely proportional to the variances (squared standard deviations) of the predictions. Except for the variance, the two methods are assumed to be equally accurate.

$$CombinedPrediction = \frac{\left(\frac{\text{ML Prediction}}{\text{ML Std}^2} + \frac{\text{CBR Prediction}}{\text{CBR Std}^2}\right)}{\left(\frac{1}{\text{ML Std}^2} + \frac{1}{\text{CBR Std}^2}\right)} \tag{8.2}$$

Here:

- ML Prediction and CBR Prediction are the individual predictions.
- ML Std and CBR Std are the standard deviations (uncertainties) of the predictions.

This formulation ensures that predictions with lower variance (higher certainty) have greater influence on the combined result.

8.2.1.6 Calculating the Combined Standard Deviation

The combined standard deviation is derived using the following formula:

$$\text{Combined Std} = \sqrt{\frac{1}{\left(\frac{1}{\text{ML Std}^2} + \frac{1}{\text{CBR Std}^2}\right)}}$$

This accounts for the uncertainties of both methods and provides the uncertainty for the combined prediction. This is appropriate as the underlying uncertainty is not "covered" by combining the methods but is also present in the combined prediction.

8.2.1.7 Extracting Confidence Intervals and Error Bars

The 95% confidence intervals for the combined predictions are calculated based on the combined mean and standard deviation. Specifically:

$$\text{Confidence Interval} = \text{Combined Prediction} \pm 1.96 \cdot \text{Combined Std}$$

Error bars are directly derived from these confidence intervals, providing a clear visualization of the uncertainty in the combined predictions. These are also the ones that are plotted in the various graphs to follow.

8.2.2 Benefits of the Bayesian Approach

The Bayesian fusion method offers several advantages:

- **Dynamic Adaptability:** The framework adapts to varying levels of confidence in the CBR and ML predictions, ensuring robustness in diverse scenarios.
- **Uncertainty Quantification:** Probabilistic outputs include confidence intervals, offering transparency and aiding decision-makers in assessing prediction reliability.

- **Weighted Contribution:** Predictions are weighted by their relevance, determined by the similarity score and the prior confidence from the ML model.

By leveraging the strengths of both CBR and ML systems, the Bayesian approach provides a unified and probabilistic methodology for decision support in flooding emergencies. This integration enables robust, adaptive, and interpretable predictions.

8.3 Dempster-Shafer Theory

DST is a mathematical framework for modelling and combining uncertain evidence. It assigns belief, disbelief, and uncertainty masses to hypotheses, enabling decisions based on partial and conflicting information.

This method is used to aggregate all predictions into a single data point expressed through the setting of threshold values (explained further below) that correspond to a "dangerous situation". Instead of providing exact Sfac or TTC values, those predictions are used to express the "belief" or "disbelief" that a dangerous flooding incident is expected. This danger is understood here to mean that there is a significant risk to human life, and an evacuation should be ordered. The belief then means that this dangerous scenario is happening, and the disbelief is the inverse. By adjusting the thresholds for SFAC and TTC, the belief-disbelief can be moved accordingly, corresponding to more "aggressive" or "conservative" outcomes. Heres how it works in the context of the code.

8.3.1 Belief Mass Calculation

For each prediction, the DST framework calculates three components:

- **Belief** (b): Confidence that the prediction satisfies a given condition (e.g., being below a threshold).

- **Disbelief** (d): Confidence that the prediction contradicts the condition.

- **Uncertainty (u)**: Remaining confidence, ensuring $b + d + u = 1$.

Given a prediction, its standard deviation (σ), and a threshold (T), the masses are computed as:

$$b = \max(0, \min(1, \frac{T - \text{prediction}}{2 \cdot \sigma})), \tag{8.3}$$

$$d = \max(0, \min(1, \frac{\text{prediction} - T}{2 \cdot \sigma})), \tag{8.4}$$

$$u = 1 - b - d. \tag{8.5}$$

This ensures predictions close to the threshold have higher uncertainty, while predictions far from the threshold lean towards belief or disbelief.

8.3.2 Threshold selection

Selecting any value for threshold is mostly an arbitrary exercise; however, there are several considerations required to make a good selection. Both TTC and SFAC are used as proxies for determining whether a dangerous situation is happening, hence, the threshold for each should lie at a point between "definitely safe" and "potentially dangerous". How close to either region the threshold lies is also a preference for how the system will be used. For this implementation, and having witnessed the working of the predictions generally and specifically in cases, the following thresholds are used.

1. Survival factor: 0.8
2. Time to capsize: 50 min

Generally, any survival factor below 1 indicates quite a substantial role or degradation of the GZ curve, which could be perfectly safe but can easily become dangerous, especially considering a vessel in a seaway. See below the current formulation according to SOLAS.

The survival factor $s_{\text{final},i}$ is calculated as follows:

$$s_{\text{final},i} = K \cdot \max\left(0, \min\left(1, \frac{GZ_{\max} \cdot \text{Range}}{0.12 \cdot 16}\right)\right)$$

Where:

- $GZ_{\max}$: The maximum value of the righting lever (not to exceed 0.12 m).
- Range: The range of positive stability (not to exceed 16°).
- K: A factor dependent on the equilibrium angle θ_e and the limiting angles $\theta_{\min}$ and $\theta_{\max}$, defined as:

$$K = \begin{cases} 1, & \text{if } \theta_{\min} \leq \theta_e, \\ 0, & \text{if } \theta_{\max} \leq \theta_e, \\ \frac{\theta_{\max} - \theta_e}{\theta_{\max} - \theta_{\min}}, & \text{otherwise.} \end{cases}$$

Hence, a value of 0.8 is selected

Explanation of K: The factor K accounts for the position of the equilibrium angle θ_e relative to the ship's critical stability angles $\theta_{\min}$ and $\theta_{\max}$:

- If θ_e is within the range $[\theta_{\min}, \theta_{\max}]$, K interpolates linearly.
- If θ_e is outside this range, K is set to 0 or 1 as appropriate.

A roll angle of 8.6 degrees would result in a survival factor of 0.8, assuming maximum values for $GZ_{\max}$ and Range which seems like a reasonable angle to consider as the demarcation line between safe and potentially dangerous. Of course, the threshold can be easily adjusted, and better values could be determined with further implementation.

For TTC, the threshold selection can be decided more quantitatively. Any TTC close or less than the time required for a complete evacuation is considered very dangerous, as are the consequences of not ordering a timely evacuation. Adding safety margins and uncertainty to that value increases the threshold above the minimum evacuation time required.

In the present implementation, time domain cases (that can provide TTC) are only run for up to 60 mins. Also, not all cases are evaluated in the time domain. A machine learning model is used to populate all cases trained on the subset of cases that have TTC values, along with the rest of the common damage variables. Therefore, the database present is not 1:1 with reality, as a perfectly safe case will be given 60 mins TTC at most. Keeping that in mind, the threshold is set at 50 minutes, which should roughly correspond with the SFAC threshold.

8.3.3 Dempster's Rule of Combination

Dempster's rule combines belief masses from two sources while considering conflicts between them. Let (b_1, d_1, u_1) and (b_2, d_2, u_2) be the belief, disbelief, and uncertainty masses from source 1 and source 2, respectively.

8.3.3.0.1 Conflict Calculation The conflict factor k measures the disagreement between the two sources:

$$k = b_1 \cdot d_2 + d_1 \cdot b_2. \tag{8.6}$$

If $k = 1$, the sources are in total conflict, and the combination fails.

8.3.3.0.2 Combined Belief Masses. The combined belief, disbelief, and uncertainty are given by:

$$b_{\text{combined}} = \frac{b_1 \cdot b_2 + b_1 \cdot u_2 + u_1 \cdot b_2}{1-k}, \tag{8.7}$$

$$d_{\text{combined}} = \frac{d_1 \cdot d_2 + d_1 \cdot u_2 + u_1 \cdot d_2}{1-k}, \tag{8.8}$$

$$u_{\text{combined}} = \frac{u_1 \cdot u_2}{1-k}. \tag{8.9}$$

The normalization factor $(1-k)$ ensures conflicting evidence is penalized, preventing overconfidence in the presence of disagreement.

8.3.4 Application to Predictions

The DST framework is applied separately to predictions of TTC and SFAC:

1. **Belief Mass Computation:** Using the equations for b, d, and u, belief masses are calculated for ML and CBR predictions.

2. **Belief Combination:** The belief masses from ML and CBR are combined using Dempster's rule, resulting in a fused prediction characterized by its combined belief, disbelief, and uncertainty.

8.3.5 Advantages and Limitations of Dempster-Shafer Theory

8.3.5.0.1 Advantages Dempster-Shafer Theory offers several benefits when fusing predictions, particularly in scenarios involving uncertainty and conflicting evidence:

- **Explicit Uncertainty Representation:** Unlike probability-based approaches that require all uncertainty to be assigned to specific outcomes, DST treats uncertainty as a distinct component. This feature allows for better handling of situations where the evidence is incomplete, ambiguous, or imprecise. For example, when ML or CBR predictions are uncertain, DST explicitly captures this uncertainty instead of forcing overcommitment to belief or disbelief.

- **Conflict Resolution:** Dempster's rule of combination effectively handles conflicting evidence between sources. By introducing the conflict factor (k) and normalizing the combined belief masses, the method avoids overconfidence in the presence of contradictory inputs. This is particularly beneficial in systems where ML and CBR might produce diverging predictions due to different underlying models or data biases.

- **Flexibility in Evidence Fusion:** DST does not require probabilities to sum to one, offering more flexibility in how belief is distributed among hypotheses. This is advantageous in dynamic or uncertain environments, such as real-time maritime decision support, where data sources may not always provide complete or consistent evidence.

- **Applicability to Multisource Data:** DST is well-suited for combining evidence from heterogeneous sources. In this context, it seamlessly integrates the data-driven ML predictions with the knowledge-based CBR outputs, leveraging the strengths of both methods while accounting for their uncertainties.

- **Robustness to Incomplete Information:** DST is capable of managing situations where one or more data sources fail to provide meaningful predictions. By assigning higher uncertainty to such inputs, the framework ensures that incomplete evidence does not unduly influence the combined result.

8.3.5.0.2 Limitations Despite its advantages, Dempster-Shafer Theory has some limitations that should be carefully considered:

- **Computational Complexity:** The combination of belief masses involves calculating normalization terms and managing the conflict factor, which can become computationally expensive for a large number of hypotheses. While this may not be an issue for systems with a limited hypothesis space (e.g., TTC and SFAC), more complex systems may require optimization techniques to ensure scalability.

- **Sensitivity to High Conflict:** When the conflict factor (k) approaches one, the normalization process amplifies the uncertainty and reduces the reliability of the combined belief. In extreme cases of total conflict, where $k = 1$, Dempster's rule fails to provide a meaningful combination. This sensitivity necessitates careful preprocessing or validation of evidence sources to minimize extreme conflicts.

- **Subjectivity in Threshold Selection:** The belief mass calculations rely on thresholds (e.g., TTC and SFAC thresholds) and standard deviations, which must be defined a priori. Poorly chosen thresholds or assumptions about uncertainty can lead to suboptimal fusion results. This dependence highlights the importance of domain expertise in configuring the DST framework.

- **Interpretability Challenges:** Although DST provides a rich framework for uncertainty modelling, interpreting the belief, disbelief, and uncertainty masses in a way that is actionable for decision-makers can be nontrivial. This is particularly relevant in safety-critical applications, where stakeholders need intuitive explanations for the results.

- **Handling of Total Uncertainty:** In cases where both sources assign very high uncertainty (low belief and disbelief), the framework may struggle to provide actionable insights. This limitation reflects the underlying assumption that some level of evidence is available from at least one source.

- **Dependence on Accurate Modelling of Sources:** The effectiveness of DST depends on how well the individual sources (ML and CBR) model their respective uncertainties. If these sources underestimate or overestimate their uncertainty, the combined result may be biased.

8.4 Integration of CBR & ML and discussion

The integration of Case-Based Reasoning (CBR) and Machine Learning (ML) in damage stability assessment represents a synergistic approach that combines the interpretability and domain-specific reasoning of CBR with the predictive power and scalability of ML. This integration aims to leverage the strengths of both methodologies while addressing their individual limitations, thereby creating a robust decision support framework for maritime safety applications. This section explores the advantages and limitations of CBR and ML, highlighting their complementary nature and the potential benefits of their integration.

8.4.1 Advantages and Limitations of Case-based reasoning (CBR)

Case-Based Reasoning (CBR) is a problem-solving methodology that relies on retrieving and adapting solutions from past cases to address new problems. It has been widely adopted in decision support systems due to its ability to provide interpretable and context-specific recommendations based on historical data. CBR operates through a cycle of retrieving similar cases, reusing their solutions, revising them to fit the current scenario, and retaining the new case for future use.

One of the primary strengths of CBR lies in its interpretability and transparency. Since CBR bases its predictions on previously recorded cases, the decision-making process is inherently more understandable for human operators. This transparency is particularly valuable in safety-critical domains like maritime operations, where stakeholders need to trust and verify the reasoning behind recommendations. Additionally, CBR facilitates knowledge reuse by directly applying historical cases, making it highly effective in domains where expert knowledge and past experience are critical. This approach allows for incremental learning, as the system can continuously improve by incorporating new cases without requiring complete model retraining. Furthermore, CBR is robust in scenarios with limited data, as it can function effectively with a relatively small number of well-documented cases, unlike ML, which typically requires large datasets for training.

Despite its advantages, CBR has several limitations. Its effectiveness is heavily dependent on the quality and relevance of the case database. If the database lacks cases that are sufficiently similar to the current scenario, the system may struggle to provide reliable recommendations. Additionally, the determination of similarity between cases often relies on heuristic or expert-defined rules, which can introduce subjectivity and bias. Another limitation is CBR's limited ability to capture complex, nonlinear patterns in data. Since CBR relies on explicit comparisons rather than learning intricate interactions among variables, it may be less effective in scenarios with highly complex dependencies. Finally, as the case library grows, the retrieval and adaptation processes can become computationally expensive, necessitating efficient indexing and retrieval strategies to maintain performance.

8.4.2 Advantages and Limitations of Machine Learning (ML)

Machine Learning (ML) has revolutionized predictive modelling by enabling the automated extraction of patterns from large datasets. ML techniques, particularly those based on statistical learning, have demonstrated remarkable success in recognizing complex relationships and making accurate predictions. In the context of damage stability assessment, ML can complement CBR by providing high-speed, data-driven insights that enhance decision-making.

One of the most significant advantages of ML is its ability to handle large-scale datasets and extract meaningful relationships that may not be explicitly defined in rule-based or case-based systems. This capability allows ML models to generalize to new, unseen situations, making them highly adaptable to diverse scenarios. Unlike CBR, which requires explicit matching with past cases, ML models can infer patterns from data, enabling them to make predictions even in novel contexts. Additionally, many ML models can automatically identify significant features in the data, reducing the need for extensive manual feature engineering. Once trained, ML models can make predictions in real time, making them suitable for rapid decision support applications. Furthermore, certain ML methods, such as Bayesian Neural Networks and Monte Carlo Dropout, can provide probabilistic estimates, allowing decision-makers to assess the confidence levels of predictions and make informed choices under uncertainty.

Despite its strengths, ML has several limitations that must be addressed. One of the most notable drawbacks is the lack of interpretability in many ML models, particularly deep learning architectures. These models often function as "black-box" systems, making it difficult for human operators to understand how predictions are derived. This lack of transparency can be a significant barrier in safety-critical applications, where trust and explainability are paramount. Additionally, ML models typically require large, labelled datasets for training, which can be challenging to acquire in maritime safety applications. The quality of ML predictions is also highly sensitive to the quality of the training data; noisy or biased datasets can lead to inaccurate or unreliable predictions. Moreover, ML models may degrade over time as new data becomes available, necessitating periodic retraining to maintain accuracy. Finally, while ML inference is fast, the training process can be computationally expensive, particularly for deep learning models, and may require specialized hardware.

8.4.3 Complementary Nature of CBR and ML

The integration of CBR and ML in damage stability assessment offers a powerful approach that combines the interpretability and domain-specific reasoning of CBR with the predictive accuracy and scalability of ML. By leveraging the strengths of both methodologies, a hybrid decision support system can address the limitations of each approach while enhancing overall performance. For instance, CBR can be used to retrieve and present past cases, ensuring that decision-making remains interpretable and explainable. At the same time, ML can generate fast, probabilistic predictions, providing a quantitative basis for decision support. The two approaches can be fused probabilistically, using techniques such as Bayesian updating or Dempster-Shafer Theory, to balance historical knowledge with data-driven insights. Additionally, the hybrid framework can improve adaptability by allowing CBR to guide ML feature selection and enabling ML to suggest new cases for inclusion in the CBR library. This integration ensures that damage stability predictions are both reliable and actionable, supporting maritime safety decision-making under uncertain conditions.

In summary, the complementary nature of CBR and ML enables the creation of a robust and flexible decision support system that combines the best of both worlds. By integrating these approaches, stakeholders can benefit from the interpretability and domain knowledge of CBR while leveraging the predictive power and scalability of ML, ultimately enhancing the safety and efficiency of maritime operations.

Below is a pictorial representation of the final assembly and integration of the methodology items shown above.

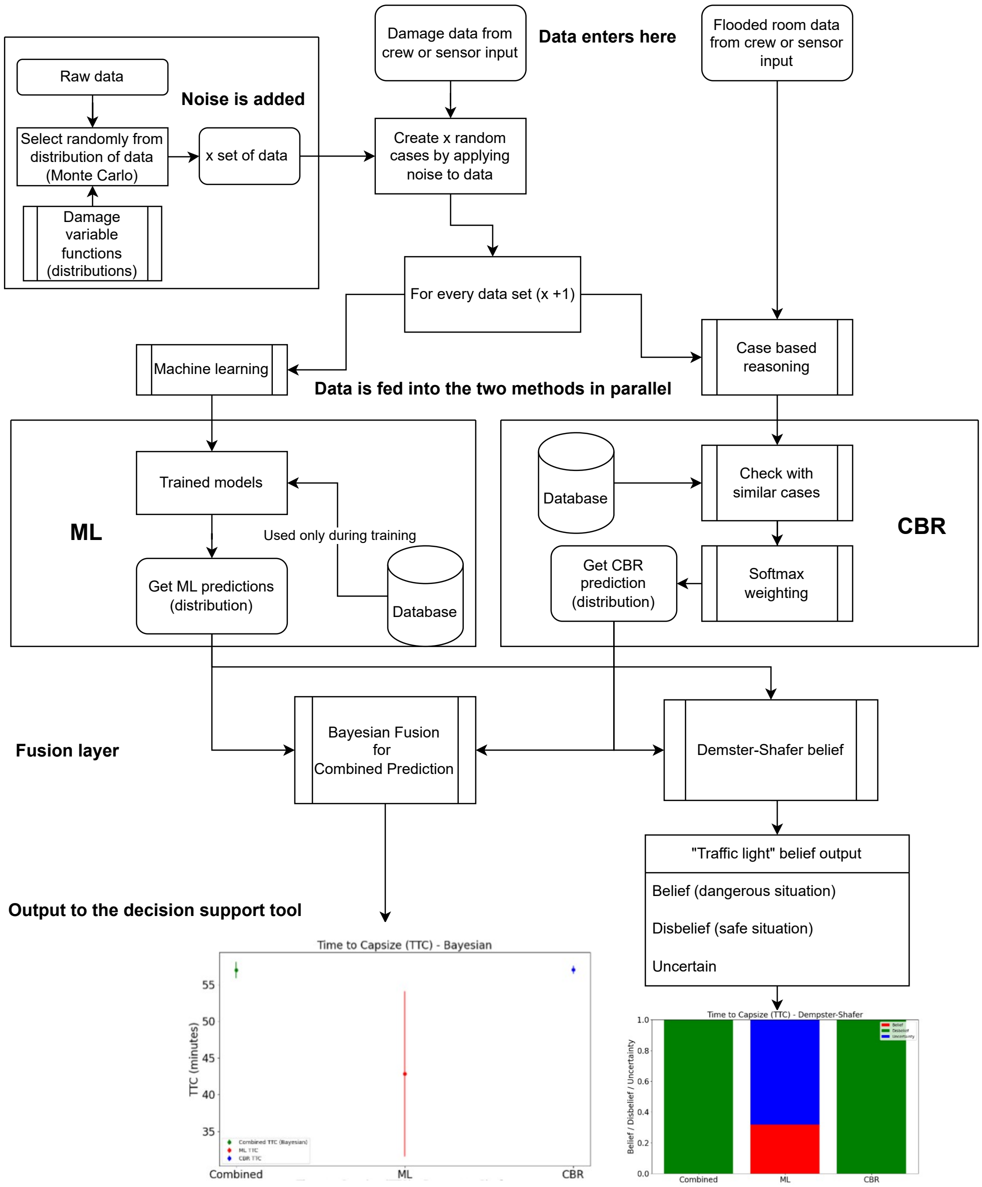

Figure 8.8: Complete methodology presented as a flowchart

8.5 Demonstration and case studies

To evaluate and demonstrate the decision support implementation, all the processes presented are implemented in code along with the other elements shown in previous chapters. Generally, the decision support elements are integrated with predictions and interacted with using a basic graphical user interface. This allows the loading of .csv files containing simulated sensor data and selecting the size of distribution for the ML method, which creates the distribution of predictions. For each time step (row in the .csv file), the predictions are calculated using CBR and ML as well as the decision support elements. The consolidated predictions are then sequentially presented in a graph.

8.5.1 Prediction fusion demonstration

To begin with, here is how the Bayesian fusion and DST approach work on the predictions for each case.

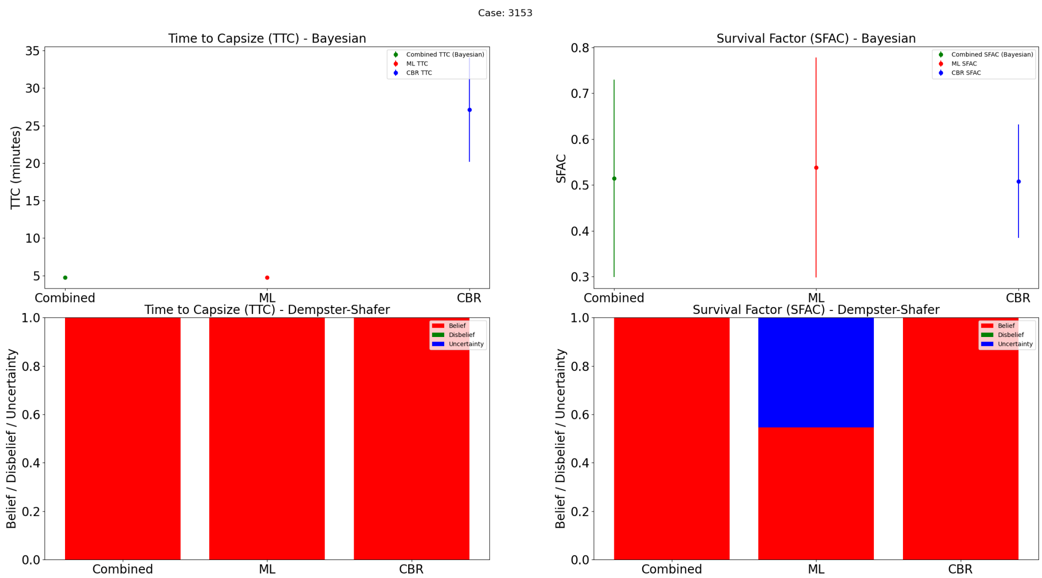

Figure 8.9: Case 3153 predictions: Certain dangerous situation

ML predicts a very low TTC, with CBR predicting a higher one, but with a high range. Both predictions are well below the threshold, hence, DST returns a belief meaning "certain belief that a dangerous situation exists". In terms of SFAC, the combined prediction has a large range that is well below the threshold. The ML prediction has a large range that is also captured as uncertainty in the belief; however, the overall picture is painted as one of clear danger and low survivability.

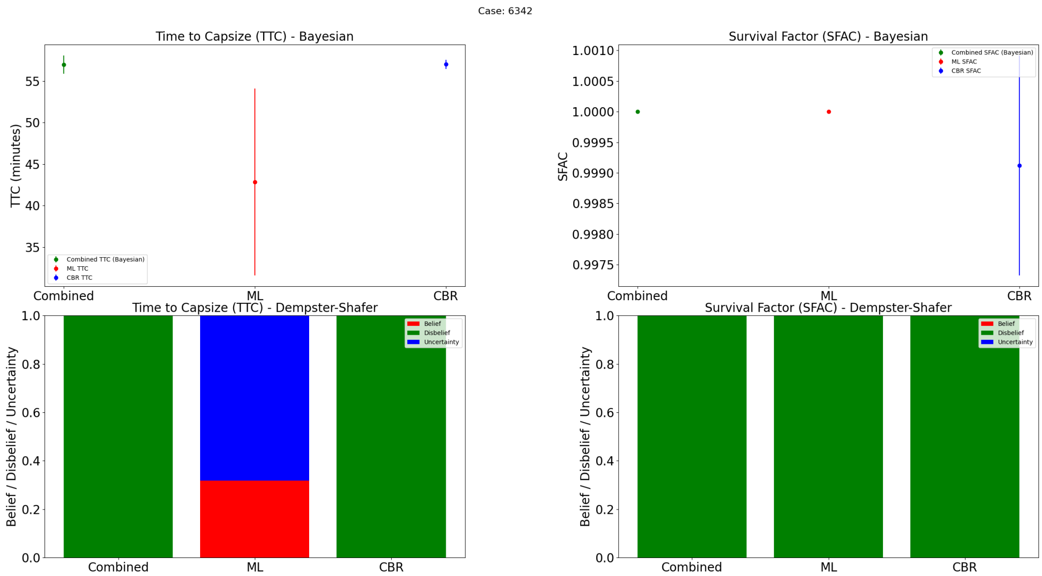

Figure 8.10: Case 6342: Certain survival

In the above case, most predictions provide a robust disbelief and large SFAC and TTC values. Moreover, most values have a low error range except for ML for TTC. This is captured as an uncertainty in the DST as it overlaps the threshold value.

8.5.2 Monte Carlo predictions with fusion and DST

Below are the outputs of further rounds of Monte Carlo simulations, now with the combined prediction, true value (from database) and annotated with the belief from

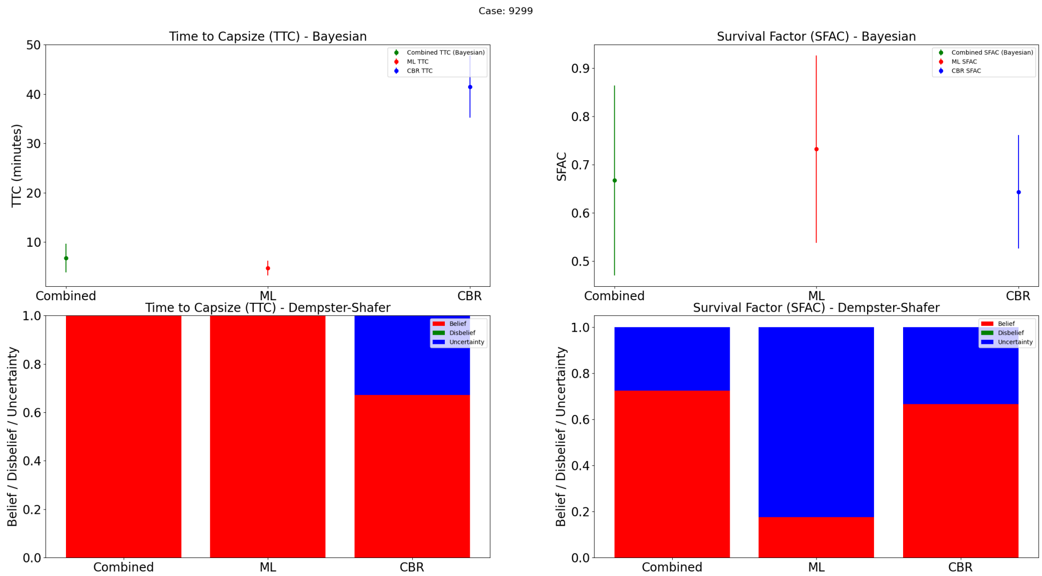

Figure 8.11: Case 9299: A case with uncertainty, especially from the SFAC.

DST.

The scatter points are colored green for the combined prediction plus the error bar, red for the true value of the case used. If a star is also on the point, then that prediction has belief(DST) larger than 0, meaning that it is recognized as a dangerous situation according to the process explained above.

The overall histograms are also plotted to appreciate the overall performance of the predictions.

Overall, as expected, the TTC predictions are usually skewed due to the artificial filling and the limit of 60 mins. The performance of SFAC is usually better and, in most cases, captures the correct outcome (DST) even if the exact value is not correct.

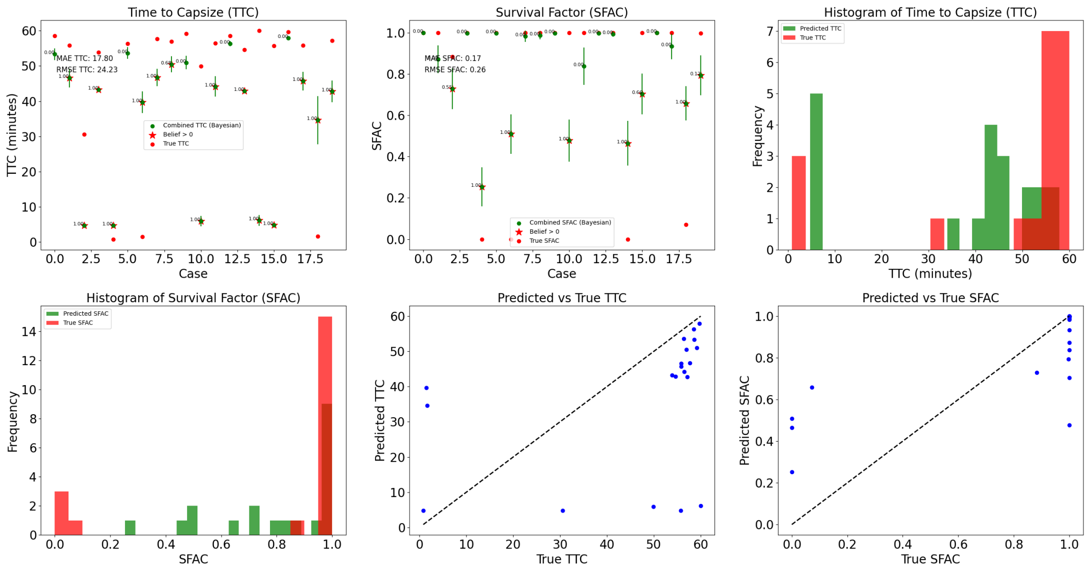

Figure 8.12: 20 random cases predictions analysis

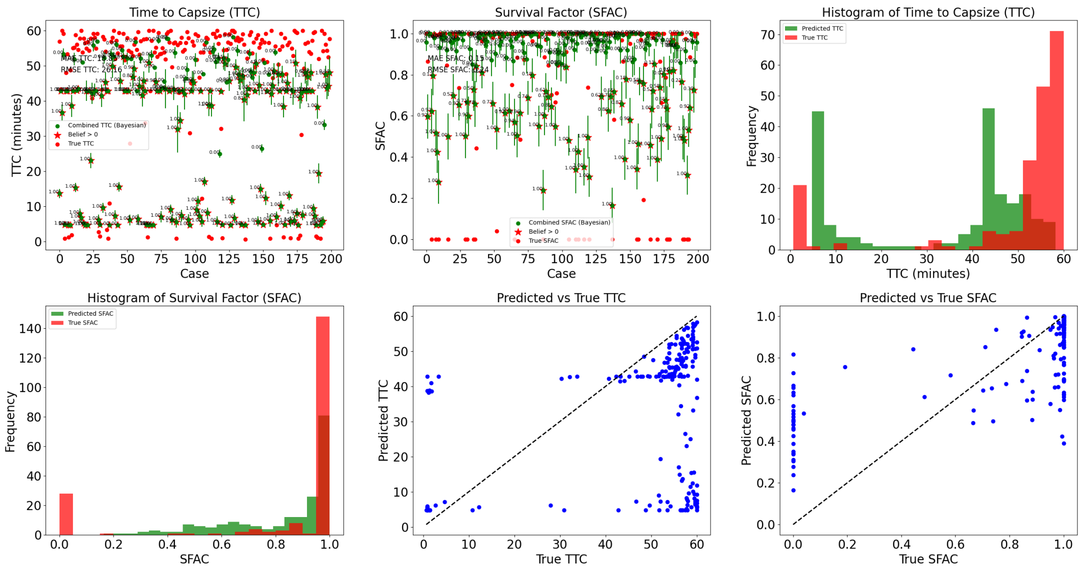

Figure 8.13: 200 random cases with predictions

8.5.3 Emergent case simulator with GUI

A simulator is constructed using the functions in the code and shedding any unnecessary development functions. The resulting function benefits from a graphical user interface that allows starting, running and evaluating the predictions of the simulated incident.

In the following example, the same case is used as in the section 8.1. As before, 10 sequential updates are given, generally increasing in severity.

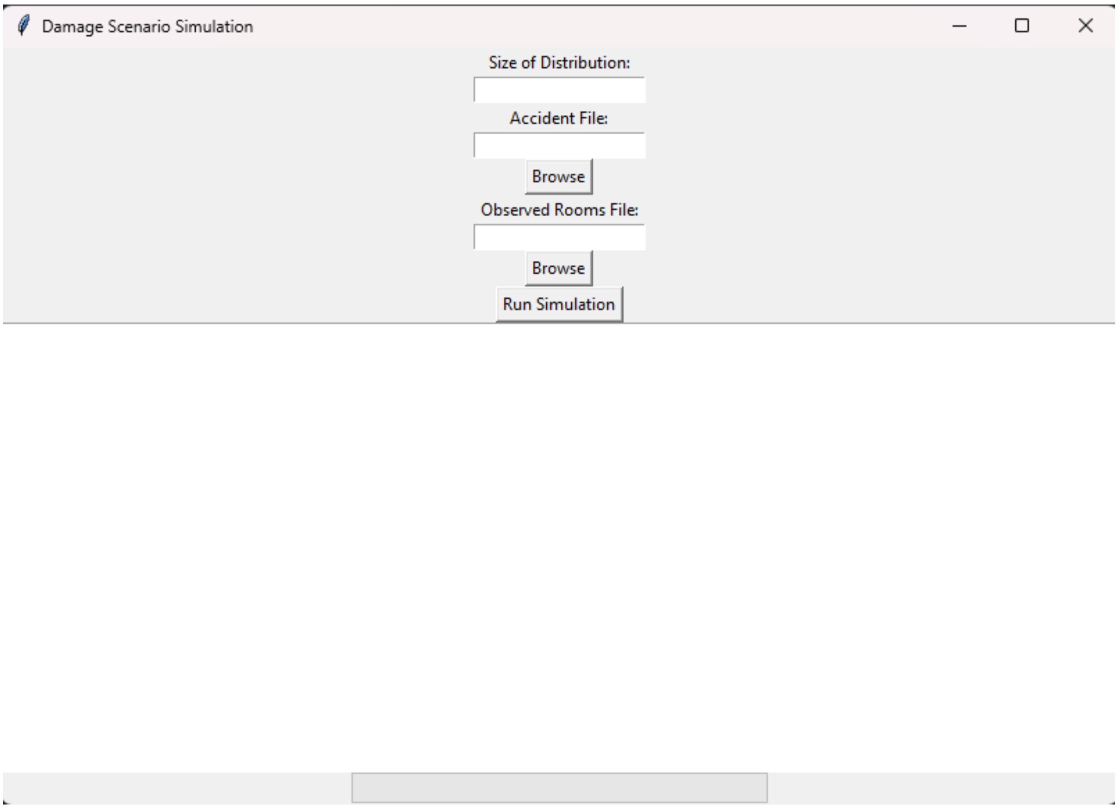

Figure 8.14: Simulator initialization with inputs required.

With each time-step a new prediction is plotted for both TTC and SFAC along with the error bars and colored according to the DST analysis. A green colour means that there is a sufficiently low belief that the case is dangerous, in other words, that it is safe.

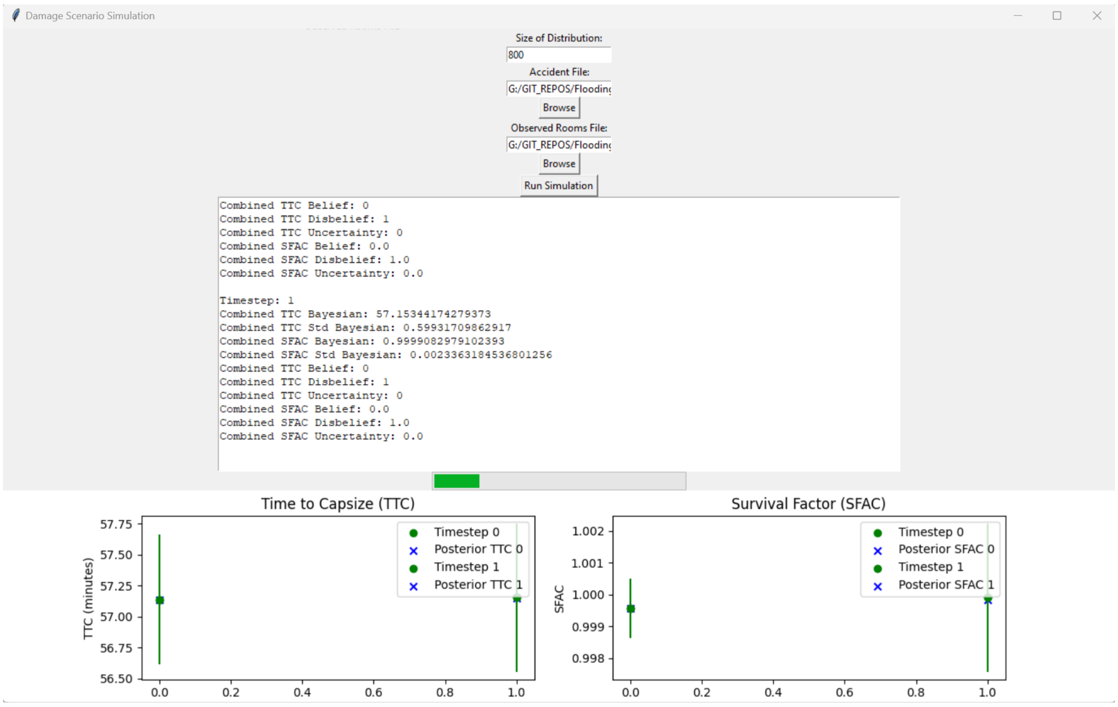

Figure 8.15: GUI during running

Similarly to the previous simple implementation, the increasing severity is evident in the decreasing predicted values for both TTC and SFAC, and after the 6th time step, the DST calculation also returns high belief in the severity of the case, as expected.

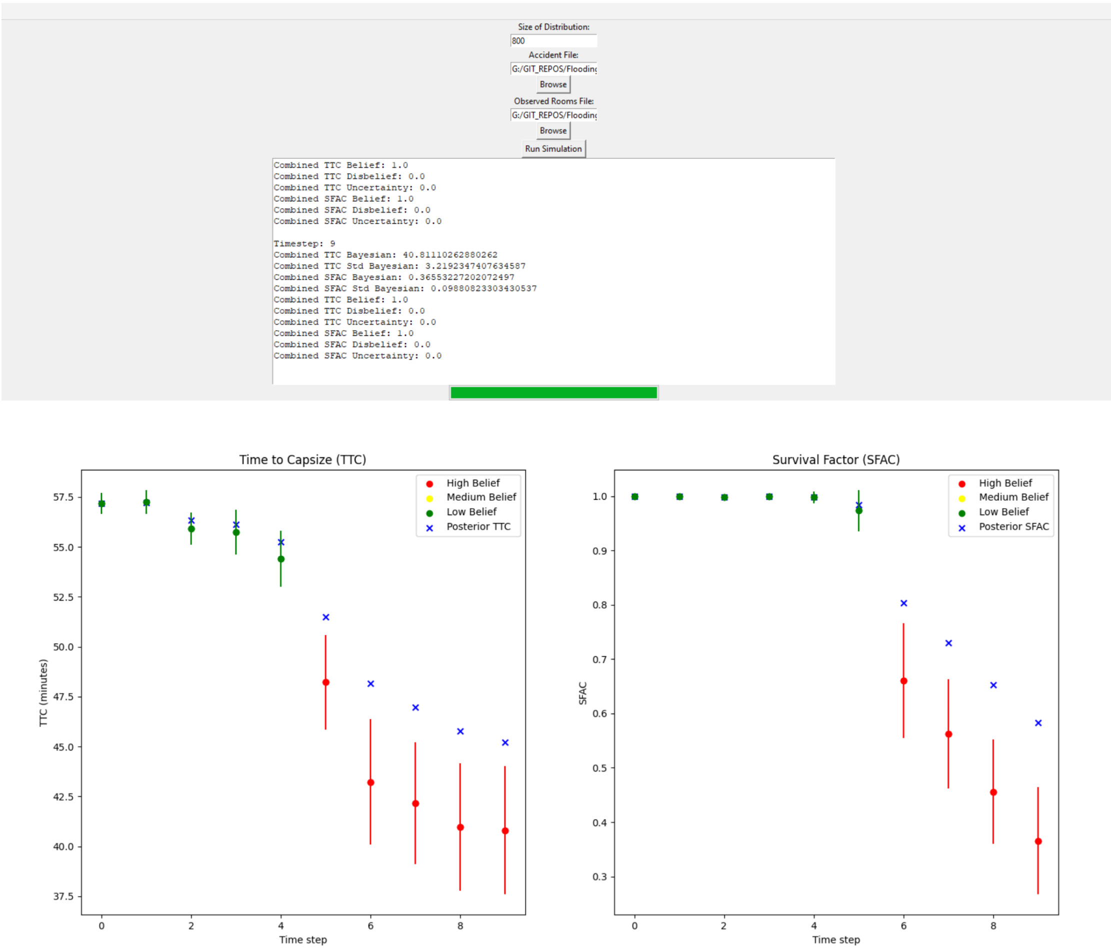

Figure 8.16: GUI state at termination

The posterior values that are plotted are another implementation of Bayes' rule, this time temporally. Each prediction is thought of as a prior, and the next prediction is the posterior. The result is then influenced by the previous cases, also taking into account the uncertainty. This would be useful in case of rapid or noisy inputs to the code and is demonstrated here without having a specific application, as it stands, as the data are artificially created and entered.

Chapter 9

Concluding remarks

9.1 Discussion

The results of this research demonstrate that the integration of Machine Learning (ML) and Case-Based Reasoning (CBR) within a probabilistic decision support framework can significantly improve real-time flooding risk assessment in passenger ships. However, while the methodology offers clear advancements over traditional quasi-static and deterministic models, its practical implementation presents several challenges that must be critically analyzed.

One of the key strengths of the proposed framework lies in its ability to provide real-time probabilistic predictions, leveraging the integration of two data-driven approaches with pre-computed simulations. Unlike conventional damage stability approaches, which often rely on deterministic calculations, this methodology captures the inherent uncertainty in flooding progression and ship stability as well as the uncertainty that can be expected at the inputs of a real-time decision support tool. By integrating Bayesian inference and Dempster-Shafer theory, the framework enhances the robustness of decision-making by quantifying confidence levels in predictions in a straightforward way that lends itself to being used in a decision support environment. The inclusion of uncertainty in the output also allows the methodology to be forthcoming about its limitations and qualify its predictions with confidence intervals. In addition, excluding the initial database creation computational cost that is "paid" in advance and only once, the cost of running the models and CBR methodology is very modest, allowing the scaled application of such a "tool" onboard.

Despite these advantages, the methodology has certain limitations. First, its reliance on historical and simulated data means that its accuracy is contingent on the quality, quantity and representativeness of the dataset. Furthermore, the models and methods could be explored further for optimizing performance, especially in cases of larger datasets.

Another critical consideration is the interpretability of ML-generated predictions. While CBR enhances explainability by referencing historical cases, ML models, particularly deep learning (neural networks) or others, often function as "black boxes," making it difficult for ship operators to fully trust automated recommendations. This research addresses this challenge by incorporating case-based reasoning as a validation layer since its result can be easily traced back to similar (run) cases, but further work is needed to improve transparency and perhaps further connect CBR with ML to allow for an automatic "validation" or check.

From an operational perspective, the adoption of this framework depends on its integration with existing shipboard decision support systems. The methodology does not aim to replace traditional damage control strategies but rather to augment them with real-time predictive insights. In fact, the methodology can be thought of as a "calculator" or "reader" of the typical damage stability probabilistic results adapted for operational use since it can take into account the whole set of damage stability data and present the user with an informed, specific and qualified (in terms of certainty) answer.

It automates the cognitive functions that take place in the minds of the crew during an emergency using their available resources. Therefore, it does not represent a significant paradigm shift for damage stability but rather an effective exploitation of the available data. Of course, implementing such a system requires further alignment with industry standards, regulatory considerations, and crew training to ensure seamless usability in high-pressure situations.

Lastly, it can be easily discerned how the applicability of this methodology extends beyond real-time stability assessment, offering potential benefits for broader safety frameworks, including evacuation planning and damage control optimization. By consolidating multiple sources of information into a unified probabilistic model, this approach facilitates more informed decision-making, reducing the likelihood of unnecessary evacuations and enhancing situational awareness.

While the results presented as a demonstration may appear mediocre, they definitely succeed in validating the efficacy of the proposed approach, especially considering the small dataset and relatively rudimentary ML models. It is noted that the purpose is to establish that these methodologies can improve understanding and allow incorporating of more data sources.

Future work should focus on expanding the dataset to include a wider range of real-world flooding incidents, improving model interpretability, increasing the performance of the machine learning models, as well as exploring the inclusion of physics models so as to incorporate first-principle tools and understanding along with the purely statistical data.

In conclusion, this research advances the state-of-the-art in maritime safety by addressing critical gaps in existing methodologies and providing a robust, transparent, and actionable decision support framework for managing flooding risks in real time. Unlike previous approaches that rely solely on deterministic or quasi-static calculations, this methodology consolidates diverse sources of information, enabling a more flexible and efficient communication of predictive insights to ship crews. By improving real-time situational awareness and integrating probabilistic reasoning, this framework contributes to safer and more informed decision-making in maritime emergencies.

9.1.1 Key Contributions

This research makes several novel contributions to the field of maritime safety, specifically in the area of flooding risk assessment and decision support. The primary contributions are as follows:

1. Development of a unified ML-CBR framework for real-time flooding risk assessment in passenger ships, combining the predictive power of ML with the interpretability of CBR.

2. Application of probabilistic uncertainty quantification methods, including Bayesian inference and Dempster-Shafer theory, to improve confidence estimation in maritime decision support systems.

3. Comprehensive validation of the frameworks efficacy through extensive simulations and case studies, demonstrating its accuracy, robustness, and practical relevance for emergency response.

9.1.2 Novelty of the Research

The novelty of this research lies in the development of a comprehensive decision support framework that integrates machine learning (ML) and case-based reasoning (CBR) methodologies for real-time maritime flooding risk assessment. While significant advancements have been made in damage stability analysis and probabilistic safety tools, this thesis addresses several key gaps in existing approaches.

A primary contribution of this research is the integration of ML and CBR methodologies within a unified framework tailored for maritime safety. This study is the first to leverage the strengths of both techniques, combining MLs pattern recognition and predictive capabilities with CBRs adaptability and interpretability. The resulting approach enhances prediction accuracy by capturing intricate relationships within extensive datasets, while also incorporating historical case knowledge to provide nuanced, context-sensitive assessments. Unlike conventional machine learning models, which may struggle with novel or highly specific scenarios, the inclusion of CBR ensures that predictions remain adaptable to unique and unforeseen circumstances. This dual-method approach not only improves prediction reliability but also enhances transparency, as decision-makers can trace the reasoning behind recommendations by referencing past cases. By offering a structured, explainable decision process, the framework fosters trust and confidence in its outputs, a critical requirement for high-stakes maritime operations.

Another key innovation of this research is its real-time decision support capability under uncertainty. The proposed framework incorporates probabilistic methods, including Bayesian inference and Dempster-Shafer theory, to quantify and manage uncertainty in real-time predictions. This allows for the dynamic calculation of critical safety metrics such as Time to Capsize (TTC) under varying environmental and operational conditions, providing predictive warnings that enable crews to make informed decisions during emergencies. Unlike conventional models that present deterministic outputs, this system explicitly communicates confidence levels by visualising uncertainty through confidence intervals and probabilistic ranges. By clarifying the reliability of predictions, this feature aids decision-makers in prioritising actions when faced with incomplete or noisy data. Additionally, the computational efficiency of the approach ensures that predictions and Monte Carlo simulations are generated within seconds, leveraging pre-calculated data where possible to support rapid evaluations.

A significant aspect of this research is its focus on passenger ships and dynamic emergency scenarios. Existing damage stability tools primarily address general ship types, but this work specifically targets the unique challenges associated with passenger vessels, where the complexity of evacuation and progressive flooding necessitates more sophisticated decision support. Given the high-risk nature of passenger ship accidents where large numbers of people may be exposed to danger, the proposed framework is designed to adapt to any ship type based on the case base and training data, improving its applicability across diverse vessel configurations. Furthermore, unlike static damage stability assessment tools, the developed framework actively simulates and predicts evolving scenarios in real time, ensuring that safety guidance remains responsive to changing conditions. By accounting for complicated geometries, watertight arrangements, and the evolving nature of flooding incidents, this approach achieves significantly higher accuracy than traditional methodologies based on fixed damage lengths.

In addition to addressing immediate safety concerns, this research aligns with broader industry trends and future regulatory needs. The transition from prescriptive to risk-based and goal-based safety regulations, as reflected in recent updates to the SOLAS damage stability criteria, underscores the need for flexible and adaptable assessment tools. This thesis contributes to this shift by integrating probabilistic and scenario-based evaluations that align with the evolving regulatory landscape. Furthermore, as the shipping industry increasingly incorporates real-time monitoring, automation, and environmental sustainability considerations, the demand for predictive decision support systems continues to grow. With the advancement of autonomous shipping and decarbonisation initiatives, proactive risk management solutions are essential. The framework developed in this research anticipates these future needs by providing a robust decision support mechanism that can complement autonomous decision-making processes in next-generation maritime operations.

By addressing these key areas, this work positions itself at the forefront of innovation in maritime safety, offering a novel and practically relevant approach to real-time flooding risk assessment. The integration of machine learning, case-based reasoning, and advanced probabilistic modelling ensures that this research contributes not only to academic advancements but also to the broader goal of improving safety and resilience in maritime operations.

9.1.3 Validity and Trustworthiness of the Generated Datasets

The reliability of the machine learning (ML) models used in this research depends heavily on the quality, validity, and trustworthiness of the generated datasets. Since these datasets form the foundation for model training and evaluation, ensuring their accuracy and representativeness is critical for producing robust and generalizable predictions.

The validity of datasets used in machine learning (ML) is a critical factor in ensuring the accuracy, generalizability, and reliability of predictive models. In ML applications, dataset validity encompasses several aspects, including representativeness, consistency, accuracy, completeness, and freedom from bias. A well-validated dataset enhances the performance of ML models by ensuring they are trained on data that accurately reflects real-world scenarios. Without rigorous dataset validation, ML models may produce misleading results, overfit to specific cases, or fail to generalise effectively to new conditions.

Dataset validity in machine learning (ML) is typically assessed across four key dimensions: construct validity, statistical validity, external validity, and internal validity. These dimensions ensure that the dataset and model not only produce accurate predictions but also maintain reliability, fairness, and applicability across different scenarios.

Construct validity refers to how well the model represents the concept it is designed to predict. This requires a clear definition of the outcome of interest, ensuring that labels and data annotations accurately reflect the intended construct. Additionally, model predictions should align with this definition, meaning that the features used in training must be relevant to the underlying problem. In the context of damage stability assessment, construct validity ensures that the models predictions of ship survivability, time-to-capsize, or progressive flooding correspond to well-established physical principles and real-world observations.

Statistical validity evaluates the reliability of the model's estimated performance, sensitivity, and uncertainty. Performance metrics must be unbiased and appropriately measured across diverse conditions. This includes assessing the models accuracy, precision, recall, and calibration against ground-truth data. The size and representativeness of the dataset used for evaluation also play a critical role in ensuring statistical validity. In this research, cross-validation techniques, uncertainty quantification methods, and sensitivity analysis were applied to verify the stability of model predictions under different conditions.

External validity examines the models generalizability, determining whether it retains predictive accuracy when applied to new or unseen cases. A model must perform consistently across different vessel types, operational conditions, and damage scenarios without overfitting to specific subsets of data. The methodology followed in this work is typically generalizable; however, it has only been applied to one case so far.

Internal validity concerns the causal relationships between the models inputs, predictions, and real-world outcomes. It ensures that the associations learned by the ML model are meaningful rather than the result of confounding variables or spurious correlations. In safety-critical applications such as damage stability assessment, internal validity is maintained by carefully selecting predictor variables based on domain knowledge rather than relying solely on data-driven correlations. This approach minimizes the risk of misleading associations and ensures that the models predictions can be interpreted in a physically meaningful way. The variable pairs selected throughout are based on sound first-principles knowledge of the underlying physical phenomena.

The dataset used in this study is derived from a combination of numerical simulations, historical incident data, and synthetic data augmentation. Time-domain simulations and quasi-static damage stability calculations were performed using industry-standard tools such as NAPA and PROTEUS, ensuring that flooding scenarios and ship responses align with established physical principles. Monte Carlo simulations were employed to introduce controlled randomness, allowing the methodology to account for variations in damage conditions and operational uncertainties. The fundamental assumption is that these simulations are an appropriate representation of the real phenomena. In fact, there are no other choices but to trust the data (as a result of using the tools that created them) as damage stability is demonstrably and notoriously difficult to capture in ”realistic terms”, meaning full-scale scenarios. The methodology tries to ”insulate” or ”protect” the results, especially the decision support aspect, by assessing how variations influence the results, using Monte Carlo simulations and using sound validation strategies for the training of the machine learning models. These more ”statistical” techniques can capture areas of low confidence, no matter the source of the uncertainty and then properly process them. The author believes that these actions can allow greater confidence in using the generated databases and indeed techniques without fearing gross misrepresentation of ground truth.

9.1.4 Limitations

As with all data-driven techniques, the quantity and quality of data are a constant requirement that can influence the effectiveness. Similarly, the depth of the database and particularly the time-domain part limit the application of the methodology. The relative abundance of quasi-static simulations compared to time-domain makes the prediction of characteristics derived from those data more appropriate compared to the time-domain ones. For example, predicting the ultimate survival factor compared to TTC or roll behaviour after x minutes. This gap has been filled with the use of ML models, mostly for demonstration purposes. Similarly, the length of the time domain simulations (30 min max, extended to 60 mins artificially) also constrains the data that can be extracted, albeit in a predictable manner. In any case, the quantity of simulated cases is very important in making sure that the CBR and ML models are "well-fed" and can predict well. Interestingly, the CBR method is able to alert the user in cases where no similar cases are found (in a sparse database) by virtue of the low similarity score that should correspond with a decreased confidence. The ML approach, on the other hand, is not interpretable in that way.

Going into more specific items of the methodology, several components can be further refined and adjusted, but only when the methodology or parts of it are evaluated further. The similarity score element of the CBR methodology is based on arbitrary values so far, based on common sense. Specifically, the damage characteristics curves and the weighting of the two similarity components are not derived epistemically but rather via common sense and limited testing in the former and latter case, respectively. In the case of the damage characteristics curves, perhaps there is no way to accurately say how they should be formulated, as the question of how similar some damages are is mostly a subjective one, as posed. Based on the mostly geometric characteristics of the input, a Bayesian search technique could be used to ascertain what damages are similar or base such a metric on the effects of the damage. The latter approach contains the hazard, however, of linking similar cases by outcome along with damage input, which might influence the performance of the system as it constitutes a distinct bias.

Regarding the ML method, the author has not exhaustively optimized the models or used all ML models. An able selection of commonly used models was instead evaluated, and the best models underwent a round of coarse hyperparameter tuning. This means that it is theoretically possible to get better results trained on the same data. However, due to several factors, this is not sought presently:

1. The data used is not extensive, and further "optimization" of the models could plateau quickly or simply be "over-fitting".

2. The methodology and process are shown to work and can be validated even with sub-optimal results.

3. The concept of making the best predictor in ML is not the focus of this work, as it is mostly a computer science topic. Also, new models, techniques or computing can greatly improve that aspect.

Another shortcoming stemming from the database is the dimensions of the case base. For example, in the time domain, all cases are run for the same significant wave height. Similarly, the vessel loading is the same (GM constant), and there is no allowance for stochastic considerations (numerous cases run for each starting condition). All these items would greatly increase the cases that need to be run, but also more accurately capture the whole "case space" that the vessel might face. For example, by not running more wave heights, the inherent assumption that the effects of waves are not large is introduced. Cases with calm seas may be underestimated in terms of survival (lower chances than expected), while cases with much higher waves overestimated in their safety. The same can be said for a range of GM values, etc. Of course, some of these values might not really affect the outcome much and hence can be safely dropped, especially considering the computational cost increases exponentially with new variables. In practice, a truly exhaustive database is impossible to compute as the number of permutations of a large number of variables will quickly become untenable. Surrogate models and simplifications can help here, averaging the effects of variables so that they can be applied to all cases without creating new instances. For example, the wave action can be modelled as a deduction to the survival factor or an amplifier to motions, and as a mathematical function incorporated into the simulation. This mathematical function will be derived from running a set of simulations with the wave action to derive it, and then generalized to apply to all other cases. This greatly reduces the computational cost and introduces only slight inaccuracies. The inaccuracy is controlled, though, by the fact that both methods are not based on individual cases and that the understanding of uncertainty is integral.

9.2 Impact

The impact of this research extends beyond theoretical advancements in maritime safety by offering a practical, data-driven approach to real-time damage stability assessment and decision support. The integration of Machine Learning (ML) and Case-Based Reasoning (CBR) within a probabilistic framework introduces a novel methodology with wide-reaching implications for ship safety, emergency response, and regulatory compliance.

9.2.1 Impact on Maritime Safety and Emergency Management

One of the most immediate benefits of this research is its contribution to improving maritime emergency response. By providing real-time flooding risk assessments and probabilistic predictions, the developed framework supports ship crews in making informed decisions under high-pressure conditions. In past maritime accidents, uncertainty regarding vessel stability played a significant role in delayed or ineffective evacuation strategies. This research addresses such gaps by equipping decision-makers with real-time insights into progressive flooding, time-to-capsize (TTC), and confidence levels associated with predictive outcomes.

Furthermore, this work enhances shipboard situational awareness by consolidating multiple sources of information, including sensor data, quasi-static cases, and time-domain simulations, into a unified probabilistic model. By reducing uncertainty and increasing the reliability of stability assessments, this framework minimizes the risks associated with unnecessary evacuations, contributing to improved safety outcomes for passengers and crew.

9.2.2 Impact on the Maritime Industry

The proposed decision support methodology has direct implications for the maritime industry, particularly in ship design, operations, and safety management. Ship operators and fleet managers can leverage this framework to enhance onboard decision support systems, leading to more efficient risk assessment and response planning. The ability to quantify uncertainty and provide probabilistic recommendations allows for better resource allocation during emergencies, potentially reducing operational disruptions and financial losses associated with avoidable evacuations or damage mismanagement.

Additionally, the integration of ML and CBR into maritime safety presents an opportunity for enhanced digitalization in the shipping industry. The growing trend toward smart ships and autonomous vessels necessitates advanced decision-support technologies capable of real-time situational analysis. This research contributes to the foundation for autonomous safety monitoring systems, paving the way for future developments in AI-driven ship operations.

9.2.3 Regulatory and Policy Implications

From a regulatory perspective, this research aligns with the International Maritime Organizations (IMO) increasing focus on probabilistic risk assessment and data-driven safety management. Current damage stability regulations, such as SOLAS probabilistic damage stability criteria, rely on predefined assumptions that may not fully capture the complexities of real-world emergencies. The methodology proposed in this research provides a complementary approach by integrating dynamic, real-time predictions with established regulatory frameworks.

By demonstrating the feasibility of ML-based and CBR-enhanced decision support in maritime safety, this research can inform future updates to safety regulations and standards. Classification societies and regulatory bodies may adopt similar probabilistic reasoning methods to refine stability assessment criteria, potentially leading to policy changes that encourage the adoption of AI-powered decision-support tools in the industry.

9.2.4 Academic and Research Contributions

Beyond its practical applications, this research contributes to the academic field by bridging the gap between traditional physics-based modelling and data-driven methodologies in maritime safety. The novel integration of ML and CBR introduces new perspectives on real-time damage stability assessment, inspiring future research in related areas such as autonomous vessel safety, uncertainty quantification, and predictive maintenance in the maritime domain.

Furthermore, the principles outlined in this study could be adapted for other complex, high-risk industries, including offshore engineering, aerospace, and disaster response, where real-time risk assessment is critical.

9.2.5 Long-Term Societal and Environmental Impact

By improving maritime safety, this framework contributes to reducing loss of life and property damage in ship-related incidents. Additionally, better flooding risk assessment and emergency planning can minimize the environmental consequences of maritime disasters, such as oil spills resulting from vessel instability or capsizing.

As the shipping industry moves toward increased automation and digitalization, the ability to integrate advanced predictive models into safety management will become increasingly valuable. The methodology proposed in this research serves as a foundational step toward data-driven maritime safety solutions that are more transparent, reliable, and adaptable to evolving industry needs.

9.3 Recommendations for future research

Stemming from the development of the methodology, validation and limitations outlined above, there are several avenues to improve the presented work, improving areas of accuracy, range of coverage and functionality. In fact, the work by the author can be thought of as more of a framework or demonstration of a case-based approach to real-time damage stability analysis rather than a complete tool ready to be installed on board. As discussed in the *Limitations* section, the exact working of the ML and CBR methods can be improved with better parameters and models. There are infinite possible model combinations that might prove able to capture the problem well, but only when furnished with an appropriately large and diverse database.

Another area where many future improvements should be attempted is the inclusion of other processes or methodologies to help and augment the overall methodology. Some of these had been attempted or developed partially by the author but were not included here.

9.3.1 Further development of CBR & ML

Currently, the integration of CBR and ML involves using separate predictions and fusing them probabilistically. A more ambitious approach could involve deep hybridization, where CBR actively informs ML models, rather than running in parallel. In simple terms, rather than treating CBR and ML to be different representations of the same base truth, a process of combining them can start with the altered assumption that each has specific strengths and weaknesses that can be anticipated and exploited or mitigated.

Below are some potential avenues for exploring a new relationship between the two methods.

CBR/ML-Guided Feature Engineering: Instead of treating CBR and ML independently, CBR-derived insights (e.g., case similarity scores, historical trends) could be used as additional features in ML models to refine predictions dynamically. Using CBR, some meta-features may be identified that can be used to train other ML models. Conversely, the same can be done by using ML-derived insights on the CBR process. In fact, this is done already for deriving the weighting for the damage variables' similarity (see 4.8). The same process can be applied in more aspects of the CBR methodology.

CBR adaptation The adaptation part of the CBR methodology has immense potential for improvement. Right now, it follows the general idea of being quite agnostic to the data and the problem by using simple statistical derivations for most of the adaptation of the case data to the emergent case. Possible improvements include actually introducing physical models to transform parts of the database case to better match the emergent case. These models would have to be informed either by the rest of similar cases or by linking several of the predicted damage case data together using models or mathematical equations. For example, it is known that the heel is linked with survival factor, as is the righting lever. The adaptation process can become even more complicated if the uncertainty and inherent stochasticity are treated as part of the prediction.

Active Learning for Database Expansion: If an ML model identifies a new case as significantly different from previous ones, it could trigger the generation of synthetic cases to expand the CBR database, improving future adaptability. Similarly, algorithms can be added to the methodology that would allow automatic detection of gaps in the case base. For example, random SOLAS cases could be generated and tested for predictions. Cases of high uncertainty returned or low CBR similarity would indicate a "gap" in the database that should trigger a request for a case to be added.

Hierarchy in Decision Support: Rather than treating ML and CBR as two equivalent methods in fusing them, their individual characteristics can be used to influence their fusion. For example, in cases of high case similarity, perhaps the CBR results should be further emphasized. This is somehow already incorporated as the standard deviation influences how much "weight" is attached to each method when carrying out the Bayesian fusion, however, it can be expanded to be based on more metrics other than the standard deviation etc.

9.3.2 Internal connectivity model & hydraulic model

One way that sensors are most likely to be used is to inform which spaces have flooding water present in them, and perhaps the level of it (immersion or water level sensors)- these sensors are actually already present in some ships. This information is then fed into the methodology, here in the CBR part only. An internal connectivity model can be created using graph theory with spaces represented as nodes and connections as edges. This model can then be used to examine and predict which rooms are damaged given a specific damage. Also, in cases where not all spaces are furnished with a sensor (which is always the case, actually), the connectivity graph can predict all the rooms that will eventually be flooded. For example, if a flooded room has unprotected openings to two other rooms that have no sensors, then these two rooms should also be considered as part of the flooded volume. The author had developed this process while working on another project, SEAMAN.

Moreover, the data from the rooms can be used as a secondary training set. For example, the flooded volume, centre of gravity (of floodwater) and moments of inertia of the flooded areas all affect the damage stability, so they could be added to the training data set for the machine learning model, for example. Specific connections can also be made between values to act as other prediction models, such as linking the inflow volume of water and, moment of inertia of the flooded area to predict instantaneous roll angles.

The connections between the rooms can be further used in a probabilistic framework. Currently, the methodology assesses similar cases to create a distribution based on their similarity; however, in reality, a slightly different case may be the result of a watertight door being open or the degradation of some other boundary. A probability can be associated with such events happening. Calculating the outcomes of these neighbouring cases, weighted by the probability of them occurring and not the similarity, is a very useful way to better populate the distribution of cases around the exact prediction, with all the added benefits to uncertainty quantification that it entails.

Lastly, a robust connectivity model and associated functions can link cases together based on first principles analysis, like flooded mass and location, thus reducing the need to run simulations for all cases. If cases with small differences can be associated because of the connected flooded volumes, then there is no need to include more of these cases in between in an attempt to populate a database. Also, this inherently improves the adaptability and flexibility of the whole technique when facing cases far from similar ones.

9.3.3 Surrogate models and first principles' tools

Several aspects of the damage stability phenomenon are well understood from first principles, or other more complicated phenomena can be simplified using surrogate models. The end result is that several aspects of the evolution of an incident can be calculated instead of being predicted using the case-based approaches. For example, if the exact spaces that are flooded are known, then the last floating position can be calculated quite easily instead of being predicted. Integrating such models and simplifications can reduce the uncertainty of predictions, as they are no longer based solely on the working of the models (ML and CBR). Similarly to above, their inclusion can also drastically reduce the need for further variables to be introduced to cases. For example, wave height, loading and metacentric height, trim, etc, could all be simplified as adjustments made to existing cases. Another less drastic technique would be to categorize values together to avoid a continuous value. This improvement is linked very closely to the one just above.

Publications

Major parts of this research has been published with open access (see below), as well as contributed to the writing of 2 deliverables for project SafePASS (EU grant No. 815146).

Machine learning and Case-based reasoning for real-time Onboard prediction on the survivability of ships, Panagiotis Louvros et al.,2023 in *Journal of Marine Science and Engineering*

Other publications by the author

Enhancing the Safety of Large Passenger Vessels in Emergencies with Technology Infusion. Evangelos Stefanou, Panagiotis Louvros, Evangelos Boulougouris, Alexandros Komianos, Nikolaos Souglakos, Dracos Vassalos in *Proceedings of the 2nd International Conference on the Stability and Safety of Ships and Ocean Vehicles*, 14-18 October 2024, Wuxi, China

Deterministic Fire Risk Assessment via Pyrolysis Modelling on a Large Public Space of a Very Large Passenger Vessel. Alexandros Komianos, Nikolaos Souglakos, Fotios Stefanidis, Panagiotis Louvros, Evangelos Stefanou, Evangelos Boulougouris, Gerasimos Theotokatos, Dracos Vassalos in *Proceedings of the 2nd International Conference on the Stability and Safety of Ships and Ocean Vehicles*, 14-18 October 2024, Wuxi, China

Alternative Evacuation Procedures and Smart Devices Impact Assessment for Large Passenger Vessels under Severe Weather Conditions. Stefanou, E.; Louvros, P.; Stefanidis, F.; Boulougouris, E. in *Sci 2024, 6, 12.* `https://doi.org/10.`

3390/sci6010012

Fuel cell, ammonia powered container ship: A case study. Panagiotis Louvros, Nikoletta L Trivyza, Alexandros Komianos, Evangelos Boulougouris, in *Transportation Research ProcediaVolume 72, 2023,* https://doi.org/10.1016/j.trpro.2023.11.713.

Multi-objective optimisation as an early design tool for smart ship internal arrangement. Louvros, P., Boulougouris, E., Coraddu, A., Vassalos, D., Theotokatos, G. (2021) in *Ships and Offshore Structures, 17(6), 13921402.* https://doi.org/10.1080/17445302.2021.1926142

Appendix A

Appendix A

A.1 Full Dataset

Below are the first 5 pages of the full dataset, as found at code completion. This dataset includes generated data from the CBR methodology, such as similarity scores. The table below explains the variables found in the table further below:

Variable	Description
STAGE	Name of stage
PHASE	Name of phase
SIDE	Heeling side PS or SB Equilibrium floating position
T	Draught in the equilibrium floating position
TR	Trim in the equilibrium floating position
HEEL	Equilibrium heeling angle
GZMAXR	Maximum height of GZ between equilibrium and angle of unprotected flooding
GZMAXSOL	Maximum righting lever GZmax as defined for calculation of 's' in SOLAS II-1, regulation 7 (SLF 42/5)

Variable	**Description**
RANGEF	Range of the greatest hump between equilibrium and angle of unprotected flooding
RANGESOL	Range of the GZ curve as defined for calculation of 's' in SOLAS II-1, regulation 7 (SLF 42/5)
SFACMOM	S factor (moment) by SOLAS II-1
SFACSOL	Factor 's' representing the probability of survival of the damage as defined in Revised SOLAS chapter II-1
SFACTYPE	Type of s factor as defined in Revised SOLAS chapter II-1: 'FINAL', 'INTERMEDIATE', 'CRITERIA' or 'NONE'
PFAC	Factor 'p' representing the probability that the damage occurs within certain zone(s) and having the penetration b
ASI	Attained subdivision index
PFAC*(1-SFAC)	-
N.DAM	Damage number (constituent database indexing)
Breach n.	Breach number (indexing)
Pfac	Duplicate of PFAC
Breach Side	0,1 derived from SIDE
Length	Damage length (longitudinal)
Pen	Penetration as binarized
Zmin	Minimum height of the damage
Zmax	Maximum height of damage
IntT	Initial draught (loadcase)
Rooms	Flooded rooms
Location	Location of damage (x-direction)

Variable	Description
Location_side	Transformation of Location by binarizing in 10 parts and adding negative for port side
Z_centre	Height of the centre (centroid) of the damage
Z_height	Height of damage opening (how big the damage is in z direction)
Roll at End Sim Time	Instantaneous roll angle at simulation end time (PROTEUS)
Sim End Time	Simulation end time. If capsize, time of capsize; otherwise, returns max sim time (30 min)
Time at Max Roll Angle	Time step when max roll is achieved
Max Roll Angle	Maximum roll angle during simulation
Max Steady Heel within 30 mins	Maximum heel (sustained) angle within 30 mins
Time at Max Steady Heel	Time step when max heel is achieved
Max Roll within 3 mins	Maximum roll achieved within the first 3 mins
Time at Max Roll within 3 mins	Time step of maximum roll within 3 mins
Time at Steady Heel at 30.00	Time spent in a steady heel at 30 mins
Steady Heave at end	Steady heave at end (unused)
Capsize	Boolean process for capsize (True/False, 1/0)
HEEL(abs)	Final heel
Location(abs)	Location of damage as binarized but in absolute numbers (without the negative/port effect)
Max Roll within 3 mins(no capsize)	Maximum roll within 3 mins but only for cases of no capsize
TTC	Time to Capsize (PROTEUS, returned as sim end time if capsize or max sim end time, otherwise filled by ML model)
TTC True	Time to Capsize only for cases with real-time domain capsizes, otherwise blank

Appendix A. Appendix A

Variable	**Description**
Adjusted TTC	Adjusted Time to Capsize for all cases according to function explained in text
Jaccard Similarity on rooms	Similarity score as a result of comparing emergent case flooded rooms to case base data, CBR methodology
Numerical Similarity on damage	Similarity score as a result of comparing damage variable data between emergent case and case base data, CBR methodology
Similarity aggregate	Similarity aggregate score by adding weighted Jaccard and numerical similarity scores, CBR methodology
Conditional probability	Result of Bayes Theorem on similarity aggregate and PFAC
Softmax weight	Weighting from Softmax procedure as explained in text

	0	1	2	3
SIDE	PS	PS	PS	PS
T	6.772227305	7.130388203	6.862850851	6.749749418
TR	-3.858875348	-4.662239844	-4.19281727	-3.77695841
HEEL	12.11534949	11.48244351	11.34057213	12.09427309
GZMAXR	0.29153136	0.243758903	0.315136392	0.287605569
GZMAXSOL	0.12	0.12	0.12	0.12
RANGEF	7.456207721	5.697742733	7.610298674	7.696527174
RANGESOL	7.456207721	5.697742733	7.610298674	7.696527174
SFACMOM	1	1	1	1
SFACSOL	0.496136425	0.512237372	0.56167104	0.501910302
SFACTYPE	FINAL	FINAL	FINAL	FINAL
PFAC	0.00010006	0.00010006	0.00010006	0.00010006
ASI	4.96434E-05	5.12545E-05	5.62008E-05	5.02212E-05
PFAC*(1-SFAC)	5.04166E-05	4.88055E-05	4.38592E-05	4.98389E-05
N.DAM	3549	3254	2735	3608
Breach n.	9236	6394	1743	9840
Pfac	0.00010006	0.00010006	0.00010006	0.00010006
Breach Side	1	1	1	1
Length	47.91380259	52.97779667	46.63398039	38.07512633
Pen	2	2	2	2
Zmin	0.270334475	0.036855646	5.757344351	2.058761487
Zmax	6.892780797	7.283277214	8.231416594	7.914766291
IntT	6.209	6.209	6.209	6.209
Rooms	0009_C', 'R030009_	)09_P', 'R040009', '	009_P', 'R040009', '	', 'R030009_C', 'R03(
Location	20.01557325	30.01120621	26.23108497	24.05736584
Location_side	-2	-2	-2	-2
Z_centre	1	1	2	2
Z_height	4	4	2	3
Roll at End Sim Time				
Sim End Time				
Time at Max Roll Angle				
Max Roll Angle				
Max Steady Heel within 30 mins				
Time at Max Steady Heel				
Max Roll within 3 mins				
Time at Max Roll within 3 mins				
Time at Steady Heel at 30.00				
Steady Heave at end				
Capsize	1	1	1	1
HEEL(abs)	12.11534949	11.48244351	11.34057213	12.09427309
Location(abs)	2	2	2	2
Max Roll within 3 mins(no capsize)				
TTC	2733.80411	1807.43012	3035.31352	3030.78135
TTC True				
Adjusted TTC	2695.855161	1839.202568	3431.568233	3022.153918
Jaccard similarity on rooms	1	0.666666667	0.733333333	0.916666667
Numerical similarity on damage	1	0.903872058	0.84149536	0.710772507
Similarity aggregate	1	0.808989902	0.79823055	0.793130171
Conditional probability	0	0	0	0
Softmax weight	0.000896773	0.000740846	0.000732917	0.000729189

	4	5	6	7
SIDE	PS	PS	PS	PS
T	6.434285046	6.583891471	7.090002813	6.84039684
TR	-2.410952679	-2.487859385	-3.941499163	-3.134801476
HEEL	10.54613933	10.16488705	10.40672627	10.9403114
GZMAXR	0.363461434	0.339993088	0.357685888	0.315335762
GZMAXSOL	0.12	0.12	0.12	0.12
RANGEF	12.48865858	12.47972979	7.774070831	9.51948509
RANGESOL	12.48865858	12.47972979	7.774070831	9.51948509
SFACMOM	1	1	1	1
SFACSOL	0.701329768	0.730599917	0.632627837	0.62564033
SFACTYPE	FINAL	FINAL	FINAL	FINAL
PFAC	0.00020012	0.00020012	0.00010006	0.00010006
ASI	0.00014035	0.000146208	6.33008E-05	6.26016E-05
PFAC*(1-SFAC)	5.97699E-05	5.39124E-05	3.67593E-05	3.74584E-05
N.DAM	358	301	3569	3222
Breach n.	8534	2291	9462	6133
Pfac	0.00020012	0.00020012	0.00010006	0.00010006
Breach Side	1	1	1	1
Length	40.65243053	43.00839702	48.79824676	52.91088561
Pen	2	1	2	1
Zmin	2.267187929	1.165664626	3.506221486	0.857254219
Zmax	8.992294877	7.032293418	7.357620861	7.342874807
IntT	6.209	6.209	6.209	6.209
Rooms	01', 'R030001', 'R03	020201', 'R030009_	09', 'R040101', 'R04	R030009_P', 'R040(
Location	20.62290671	22.20229063	34.37288372	30.62070114
Location_side	-2	-2	-3	-2
Z_centre	2	2	2	2
Z_height	4	3	2	4
Roll at End Sim Time				
Sim End Time				
Time at Max Roll Angle				
Max Roll Angle				
Max Steady Heel within 30 mins				
Time at Max Steady Heel				
Max Roll within 3 mins				
Time at Max Roll within 3 mins				
Time at Steady Heel at 30.00				
Steady Heave at end				
Capsize	1	1	1	1
HEEL(abs)	10.54613933	10.16488705	10.40672627	10.9403114
Location(abs)	2	2	3	2
Max Roll within 3 mins(no capsize)				
TTC	3275.49622	3303.3392	2925.78404	3110.283
TTC True				
Adjusted TTC	5718.19397	6593.470125	3987.352453	4150.943563
Jaccard similarity on rooms	0.75	0.5	0.473684211	0.352941176
Numerical similarity on damage	0.809369626	0.735813983	0.721421713	0.78864159
Similarity aggregate	0.785621776	0.64148839	0.622326712	0.614361425
Conditional probability	0	0	0	0
Softmax weight	0.000723734	0.000626589	0.000614697	0.00060982

	8	9	10	11
SIDE	PS	PS	PS	PS
T	6.7384953	9	6.407686723	6.626943776
TR	-2.980273261	-9	-2.265969501	-1.825139974
HEEL	9.773638818	30	10.66323721	4.645976893
GZMAXR	0.380536183	-0.361128362	0.356189874	0.423639713
GZMAXSOL	0.12	0	0.12	0.05
RANGEF	11.36403741	0	12.59391979	19.08708201
RANGESOL	11.36403741	0	12.59391979	7
SFACMOM	1	0	1	1
SFACSOL	0.742006798	0	0.693502571	1
SFACTYPE	FINAL	INTERMEDIATE	FINAL	INTERMEDIATE
PFAC	0.00010006	0.00010006	0.00010006	0.00070042
ASI	7.42452E-05	0	6.93919E-05	0.00070042
PFAC*(1-SFAC)	2.58148E-05	0.00010006	3.06681E-05	0
N.DAM	3320	3183	2934	2198
Breach n.	7009	5741	3505	4842
Pfac	0.00010006	0.00010006	0.00010006	0.00070042
Breach Side	1	1	1	1
Length	26.13158143	47.16169961	29.0261386	38.92902025
Pen	2	3	2	1
Zmin	4.638586072	2.043493663	6.094673782	1.426268097
Zmax	7.757434889	9.893713426	9.360278115	8.40166642
IntT	6.209	6.209	6.209	6.209
Rooms	', 'R030009_P', 'R04	'R030009_P', 'R04(	R020201', 'R03000	30009_P', 'R040009
Location	28.64283045	18.92714638	20.55000151	29.11050386
Location_side	-2	-2	-2	-2
Z_centre	2	2	3	2
Z_height	2	4	2	4
Roll at End Sim Time		175.056		
Sim End Time		73.599		
Time at Max Roll Angle		73.599		
Max Roll Angle		60		
Max Steady Heel within 30 mins		59.155		
Time at Max Steady Heel		73.599		
Max Roll within 3 mins		60		
Time at Max Roll within 3 mins		73.599		
Time at Steady Heel at 30.00		71.512		
Steady Heave at end				
Capsize	1	0	1	1
HEEL(abs)	9.773638818	30	10.66323721	4.645976893
Location(abs)	2	2	2	2
Max Roll within 3 mins(no capsize)		[]		
TTC	3296.1283	73.599	3092.06499	3530.42577
TTC True		73.599		
Adjusted TTC	6969.784465	33.1269975	5224.717029	65263.32927
Jaccard similarity on rooms	0.75	0.150684932	0.666666667	0.333333333
Numerical similarity on damage	0.519015821	0.872215291	0.525617028	0.717639978
Similarity aggregate	0.611409492	0.583603147	0.582036883	0.56391732
Conditional probability	0	0	0	0
Softmax weight	0.000608023	0.000591349	0.000590423	0.000579821

	12	13	14	15
SIDE	PS	PS	PS	PS
T	6.377279572	6.495280623	7.740013545	6.282221777
TR	-1.536188814	-1.792712266	-4.112838548	-0.724407575
HEEL	7.551233307	5.678276993	8.178046971	4.192626668
GZMAXR	0.063433422	0.469316065	0.372129939	0.487285645
GZMAXSOL	0.063433422	0.05	0.12	0.12
RANGEF	10.54231433	18.19244432	7.322528531	21.55350006
RANGESOL	10.54231433	7	7.322528531	16
SFACMOM	0.251390591	1	1	1
SFACSOL	0.155111765	1	0.759529491	1
SFACTYPE	FINAL	INTERMEDIATE	FINAL	FINAL
PFAC	0.00020012	0.00030018	0.00010006	0.00040024
ASI	3.1041E-05	0.00030018	7.59985E-05	0.00040024
PFAC*(1-SFAC)	0.000169079	0	2.40615E-05	0
N.DAM	2765	2654	3513	872
Breach n.	7451	8483	8953	3578
Pfac	0.00020012	0.00030018	0.00010006	0.00040024
Breach Side	1	1	1	1
Length	48.90962248	23.91778632	49.09479139	23.38851661
Pen	1	2	2	2
Zmin	2.38498376	0.868300388	0.311845454	3.039557013
Zmax	9.057624451	6.505302309	8.564340613	8.586538997
IntT	6.209	6.209	6.209	6.209
Rooms	'R020503', 'R03000C	', 'R030009_P',	'R001_N', 'R060201_N	)_P', 'R020201', 'R0
Location	16.9926365	19.3476578	48.15819805	14.31149901
Location_side	-2	-2	-3	-2
Z_centre	2	1	2	2
Z_height	4	3	5	3
Roll at End Sim Time	7.329			
Sim End Time	1820			
Time at Max Roll Angle	33.013			
Max Roll Angle	14.084			
Max Steady Heel within 30 mins	8.531			
Time at Max Steady Heel	805.732			
Max Roll within 3 mins	14.084			
Time at Max Roll within 3 mins	33.013			
Time at Steady Heel at 30.00				
Steady Heave at end	0.112			
Capsize	1	1	1	1
HEEL(abs)	7.551233307	5.678276993	8.178046971	4.192626668
Location(abs)	2	2	3	2
Max Roll within 3 mins(no capsize)	14.084			
TTC	3600	3548.62788	2394.78188	3548.62788
TTC True	3600			
Adjusted TTC	2046.721641	65599.81285	5568.689112	65599.81285
Jaccard similarity on rooms	0.078125	0.5	0.25	0.583333333
Numerical similarity on damage	0.867194091	0.577512674	0.742606382	0.517723578
Similarity aggregate	0.551566455	0.546507604	0.545563829	0.54396748
Conditional probability	0	0	0	0
Softmax weight	0.000572704	0.000569814	0.000569276	0.000568368

	16	17	18	19
SIDE	PS	PS	PS	PS
T	6.378470354	9	6.437830682	6.39333774
TR	-1.515513409	-9	-1.500363087	-1.290970391
HEEL	7.305872241	30	5.391670439	5.195417793
GZMAXR	0.375441612	-0.024754362	0.440075346	0.404720805
GZMAXSOL	0.12	0	0.05	0.12
RANGEF	17.38547877	0	19.12448907	19.81025597
RANGESOL	16	0	7	16
SFACMOM	1	0	1	1
SFACSOL	0.980696676	0	1	1
SFACTYPE	FINAL	INTERMEDIATE	INTERMEDIATE	FINAL
PFAC	0.00020012	0.00020012	0.0005003	0.0005003
ASI	0.000196257	0	0.0005003	0.0005003
PFAC*(1-SFAC)	3.86298E-06	0.00020012	0	0
N.DAM	3136	114	15	188
Breach n.	8523	6815	5076	7040
Pfac	0.00020012	0.00020012	0.0005003	0.0005003
Breach Side	1	1	1	1
Length	44.80558541	47.67063354	21.40188016	30.77738329
Pen	1	1	2	1
Zmin	2.182105203	2.330120811	0.414988483	0.778213687
Zmax	7.682973556	12.23896761	8.430153858	6.353907093
IntT	6.209	6.209	6.209	6.209
Rooms	10009_A', 'R02020	'R030009_P', 'R040	', 'R030009_C', 'R0	_A', 'R020201', 'R0
Location	9.797623578	19.91441696	25.73063135	19.38832194
Location_side	-1	-2	-2	-2
Z_centre	2	2	2	1
Z_height	3	5	5	3
Roll at End Sim Time		175.13		
Sim End Time		53.549		
Time at Max Roll Angle		53.549		
Max Roll Angle		60		
Max Steady Heel within 30 mins				
Time at Max Steady Heel				
Max Roll within 3 mins		60		
Time at Max Roll within 3 mins		53.549		
Time at Steady Heel at 30.00				
Steady Heave at end				
Capsize	1	0	1	1
HEEL(abs)	7.305872241	30	5.391670439	5.195417793
Location(abs)	1	2	2	2
Max Roll within 3 mins(no capsize)		[]		
TTC	3512.62578	53.549	3548.62788	3548.62788
TTC True		53.549		
Adjusted TTC	52770.52263	24.10246864	65599.81285	65599.81285
Jaccard similarity on rooms	0.384615385	0.095238095	0.583333333	0.416666667
Numerical similarity on damage	0.642084024	0.828242782	0.499184243	0.59512612
Similarity aggregate	0.539096568	0.535040907	0.532843879	0.523742339
Conditional probability	0	0	0	0
Softmax weight	0.000565607	0.000563317	0.000562081	0.000556989

Bibliography

[Acanfora et al., 2019] Acanfora, M., Begovic, E., and De Luca, F. (2019). A fast simulation method for damaged ship dynamics. *Journal of Marine Science and Engineering*, 7(4).

[Acanfora and Cirillo, 2017] Acanfora, M. and Cirillo, A. (2017). A simulation model for ship response in flooding scenario. *Proceedings of the Institution of Mechanical Engineers Part M: Journal of Engineering for the Maritime Environment*, 231(1):153–164.

[Almeida Santos Neves et al., 2011] Almeida Santos Neves, M., Belenky, V. L., de Kat, J. O., Spyrou, K., and Umeda, N. (2011). Contemporary Ideas on Ship Stability and Capsizing in Waves. In Moreau, R. M., editor, *Fluid Mechanics and Its Applications*. Springer Science, 1st edition.

[Arnott and Pervan, 2016] Arnott, D. and Pervan, G. (2016). A Critical Analysis of Decision Support Systems Research Revisited: The rise of Design. In Willcocks, L. P., Sauer, C., and Lacity, C. M., editors, *Enacting Research Methods In Information Systems*, volume 3, pages 43–103. Springer International Publishing AG Switzerland, St Louis, 1st edition.

[Atzampos, 2019] Atzampos, G. (2019). *A Holistic Approach to Damage Survivability Assessment of Large Passenger Ships*. PhD thesis.

[Atzampos et al., 2019] Atzampos, G., Vassalos, D., Cichowicz, J., Paterson, D., and Boulougouris, E. (2019). eSAFE-cruise ship survivability in waves. In *17th International Ship Stability Workshop*, pages 10–12.

[Begovic et al., 2013] Begovic, E., Mortola, G., Incecik, A., and Day, A. H. (2013). Experimental assessment of intact and damaged ship motions in head, beam and quartering seas. *Ocean Engineering*, 72:209–226.

[Benreguig and Murphy, 2019] Benreguig, P. and Murphy, J. (2019). Modelling air compressibility in OWC devices with deformable air chambers. *Journal of Marine Science and Engineering*, 7(8).

[Bock et al., 2019] Bock, F. E., Aydin, R. C., Cyron, C. J., Huber, N., Kalidindi, S. R., and Klusemann, B. (2019). A review of the application of machine learning and data mining approaches in continuum materials mechanics.

[Boisson, 1998] Boisson, P. (1998). *Politiques_et_droit_de_la_sécurité_mar*. Bureau Veritas.

[Bose and Mahapatra, 2001] Bose, I. and Mahapatra, R. K. (2001). Business data mining - a machine learning perspective. *Information & Management*.

[Braidotti et al., 2019] Braidotti, L., Marino, A., and Bucci, V. (2019). On the Effect of Uncertainties on Onboard Progressive Flooding Simulation.

[Braidotti and Mauro, 2020] Braidotti, L. and Mauro, F. (2020). A fast algorithm for onboard progressive flooding simulation. *Journal of Marine Science and Engineering*, 8(5).

[British Government, 1990] British Government (1990). *Report on the Loss of the SS Titanic*. Sutton Publishing Ltd.

[Bulian et al., 2020] Bulian, G., Cardinale, M., Dafermos, G., Lindroth, D., Ruponen, P., and Zaraphonitis, G. (2020). Probabilistic assessment of damaged survivability of passenger ships in case of grounding or contact. *Ocean Engineering*, 218:107396.

[Cao et al., 2019] Cao, X. Y., Tao, L., Zhang, A. M., and Ming, F. R. (2019). Smoothed particle hydrodynamics (SPH) model for coupled analysis of a damaged ship with internal sloshing in beam seas. *Physics of Fluids*, 31(3).

[Carter et al., 2019] Carter, T., Williams, J. G., and Roberts, S. E. (2019). Crew and passenger deaths from vessel accidents in United Kingdom passenger ships since 1900. *International Maritime Health*, 70(1):1–10.

[Chen, 2012] Chen, Q. (2012). Integrated Flooding Control and Standard for Stability and Crises Management FLOODSTAND-deliverable: Uncertainty bounds on time to capsize models Organisation SSRC Revision 2.0. Technical report.

[Chen, 2013] Chen, Q. (2013). *Uncertainty Quantification in Assessment of Damage Ship Survivability*. PhD thesis, University of Strathclyde, Glasgow.

[Cheng et al., 2017] Cheng, H., Zhang, A. M., and Ming, F. R. (2017). Study on coupled dynamics of ship and flooding water based on experimental and SPH methods. *Physics of Fluids*, 29(10).

[Cherkassky and Mulier, 2007] Cherkassky, V. and Mulier, F. (2007). LEARNING FROM DATA Concepts, Theory, and Methods Second Edition. Technical report.

[Committee on Commerce United States Senate,] Committee on Commerce United States Senate. TitanicReport.

[Concordia, 2012] Concordia, C. (2012). Marine Casualties Investigative Body Cruise Ship Report on the safety technical investigation. Technical report.

[Department of Transport, 1987] Department of Transport (1987). Formal Investigation MV Herald of Free Enterprise. Technical report, The Merchant Shipping Act 189.

[Dogan and Birant, 2021] Dogan, A. and Birant, D. (2021). Machine learning and data mining in manufacturing.

[Dominguez-Péry et al., 2021] Dominguez-Péry, C., Vuddaraju, L. N. R., Corbett-Etchevers, I., and Tassabehji, R. (2021). Reducing maritime accidents in ships by tackling human error: a bibliometric review and research agenda. *Journal of Shipping and Trade*, 6(1).

[Elgendi, 2019] Elgendi, M. (2019). Characteristics of a Highly Cited Article: A Machine Learning Perspective. *IEEE Access*, 7:87977–87986.

[EMSA, 2015] EMSA (2015). Risk Acceptance Criteria and Risk Based Damage Stability. Final Report, part 2: Formal Safety Assessment. EMSA/OP/10/2013 project. Technical report.

[EMSA, 2023] EMSA (2023). Annual Overview Of Marine Casualties And Incidents 2023 Annual Overview Of Marine Casualties And Incidents. Technical report.

[Gao et al., 2004] Gao, Q., Kara, F., Shigunov, V., and Vassalos, D. (2004). Numerical Simulation of Damage Ship Flooding. In *7th Numerical Towing Tank Symp. NuTTS'2004*, Hamburg.

[Gao et al., 2010] Gao, Z., Vassalos, D., and Gao, Q. (2010). Numerical simulation of water flooding into a damaged vessel's compartment by the volume of fluid method. *Ocean Engineering*, 37(16):1428–1442.

[Gao and Vassalos, 2015] Gao, Z. L. and Vassalos, D. (2015). The dynamics of the floodwater and the damaged ship in waves. *Journal of Hydrodynamics*, 27(5):689–695.

[Gunasekaran and Ngai, 2012] Gunasekaran, A. and Ngai, E. W. (2012). Decision support systems for logistics and supply chain management. *Decision Support Systems*, 52(4):777–778.

[Halford et al., 2005] Halford, G. S., Baker, R., Mccredden, J. E., and Bain, J. D. (2005). How Many Variables Can Humans Process? *PSYCHOLOGICAL SCIENCE*.

[Han et al., 2011] Han, J., Kamber, M., and Pei, J. (2011). *Data Mining. Concepts and Techniques*. Morgan Kaufmann, third edition.

[Heckscher, 2018] Heckscher, C. M. (2018). A Nearctic-Neotropical Migratory Songbird's Nesting Phenology and Clutch Size are Predictors of Accumulated Cyclone Energy. *Scientific Reports*, 8(1).

[IMO, 2013] IMO (2013). Revised Guidelines For Formal Safety Assessment (FSA) For Use In The Imo Rule-Making Process. Technical report, MSC-MEPC.2/Circ.12.

[Inter-governamental Maritime Consultative Organization, 1973] Inter-governamental Maritime Consultative Organization (1973). Resolution A.265(VIII) Regulations On Subdivision and Stability Of Passenger Ships As Equivalent To Part B Of Chapter II Of The International Convention For The Safety Of Life At Sea.

[International Maritime Organization., 2009] International Maritime Organization. (2009). *SOLAS, consolidated edition, 2009 : consolidated text of the International Convention for the Safety of Life at Sea, 1974, and its Protocol of 1988 : articles, annexes and certificates.* International Maritime Organization.

[International Maritime Organization (IMO), 2009] International Maritime Organization (IMO) (2009). International Convention for the Safety of Life at Sea (SOLAS), 2009 Amendment. MSC.216(82). Technical report.

[International Maritime Organization (IMO), 2019] International Maritime Organization (IMO) (2019). Revised explanatory notes to the SOLAS chapter II-1 subdivision and damage stability regulations. MSC.429(98). Technical report.

[ITTC, 2002] ITTC (2002). The Specialist Committee on Prediction of Extreme Ship Motions and Capsizing Final Report and Recommendations to the 23rd ITTC.

[ITTC, 2021] ITTC (2021). Recommended Procedures and Guidelines Estimation of Roll Damping. Technical report.

[Jasionowski, 2001] Jasionowski, A. (2001). *An Integrated Approach to Damage Ship Survivability Assessment*. PhD thesis, University of Strathclyde.

[Jasionowski, 2011] Jasionowski, A. (2011). Decision support for ship flooding crisis management. *Ocean Engineering*, 38(14-15):1568–1581.

[Journee et al., 1997] Journee, J. M., Vermeer, H., and Vredeveldt, A. (1997). Systematic model experiments on flooding of two Ro-Ro vessels. In *6th International Conference on Stability of Ships and Ocean Structures, STAB'97*, Varna, Bulgaria. TUDelft, Faculty of Marine Technology, Ship Hydromechanics Laboratory, Report No. 1106-P,.

[Karolius et al., 2018] Karolius, K. B., Cichowicz, J., and Vassalos, D. (2018). Risk-based positioning of Flooding Sensors to reduce Prediction Uncertainty of Damage Survivability. In *13th International Conference on the Stability of Ships and Ocean Vehicles*, pages 16–21.

[Kat and Otto, 1988] Kat, D. and Otto, J. (1988). *Large Amplitude Ship Motions and Capsizing in Severe Sea Conditions*. PhD thesis, UC San Diego.

[Khalfallah et al., 2023] Khalfallah, H. B., Jelassi, M., Demongeot, J., and Saoud, N. B. B. (2023). Decision support systems in healthcare: systematic review, meta-analysis and prediction, with example of COVID-19. *AIMS Bioengineering*, 10(1):27–52.

[Kolodner, 1996] Kolodner, J. (1996). Making the implicit explicit: Clarifying the principles of case-based reasoning. In Leake, D., editor, *Case-Based Reasoning: Experiences, Lessons & Future Directions,*, pages 349–370, Menlo Park, USA. AAAI Press.

[Kolodner, 1992] Kolodner, J. L. (1992). *An Introduction to Case-Based Reasoning***, volume 6.

[Krueger and Nafouti, 2015] Krueger, S. and Nafouti, O. (2015). A New Approach for the Water-On-Deck-Problem ofRoRo-Passenger Ships. In *12th International Conference on the Stability of Ships and Ocean Vehicles*.

[Lee et al., 2023a] Lee, G. J., Reed, A. M., van Walree, F., Peters, A., Gualeni, P., Katayama, T., and Duan, W. (2023a). Air Pressure Scale Effects During Damage Model Tests. In Spyrou, K. J., Belenky, V. L., Katayama, T., Bačkalov, I., and Francescutto, A., editors, *Contemporary Ideas on Ship Stability: From Dynamics to Criteria*, pages 555–564. Springer International Publishing, Cham.

[Lee et al., 2023b] Lee, G. J., Reed, A. M., van Walree, F., Peters, A., Gualeni, P., Katayama, T., and Duan, W. (2023b). The Inertia Contributions Due to Floodwater Mass. In Spyrou, K. J., Belenky, V. L., Katayama, T., Bačkalov, I., and Francescutto, A., editors, *Contemporary Ideas on Ship Stability: From Dynamics to Criteria*, pages 545–554. Springer International Publishing, Cham.

[Lee et al., 2012] Lee, S., You, J.-M., Lee, H.-H., Lim, T., Rhee, S. H., and Rhee, K.-P. (2012). Preliminary tests of a damaged ship for CFD validation. *International Journal of Naval Architecture and Ocean Engineering*, 4(2):172–181.

[Letizia, 1996] Letizia, L. (1996). *Damage Survivability Of Passenger Ships In A Seaway.* PhD thesis, University of Strathclyde.

[Luhmann et al., 2018] Luhmann, H., Bulian, G., Vassalos, D., Olufsen, O., Seglem, I., and Pöttgen, J. (2018). eSAFE-D4. 3.2Executive summary. Joint Industry Project eSAFE-enhanced Stability After a Flooding EventA joint industry project on Damage Stability for Cruise Ships,. Technical report.

[Manderbacka et al., 2018] Manderbacka, T., Themelis, N., Bačkalov, I., Boulougouris, E., Eliopoulou, E., Hashimoto, H., Konovessis, D., Leguen, J.-F., Míguez González, M., Rodríguez, C. A., Rosén, A., Ruponen, P., Shigunov, V., Schreuder, M., and Terada, D. (2018). An overview of the current research on stability of ships and ocean vehicles: the STAB2018 perspective. In *e 13 th International Conference on the Stability of Ships and Ocean Vehicles.*

[Manderbacka et al., 2011] Manderbacka, T. L., Matusiak, J. E., Ruponen, P. T., Manderbacka, T. L., Matusiak, J. E., and Ruponen, P. T. (2011). Ship Motions Caused by Time-Varying Extra Mass on Board. In *Proceedings of 12th International Ship Stability Workshop*, Annapolis.

[Manos et al., 2006] Manos, B., Bournaris, T., Kamruzzaman, M., Begum, M., Anjuman, A., and Papathanasiou, J. (2006). Regional Impact of Irrigation Water Pricing in Greece under Alternative Scenarios of European Policy: A Multicriteria Analysis. *Regional Studies*, 40(9):1055–1068.

[Matthies et al., 2007] Matthies, M., Giupponi, C., and Ostendorf, B. (2007). Environmental decision support systems: Current issues, methods and tools. *Environmental Modelling & Software*, 22(2):123–127.

[National Transportation Safety Board, 2015] National Transportation Safety Board (2015). Sinking of US Cargo Vessel SS El Faro. Technical report.

[Niotis et al., 2019] Niotis, A., Vassalos, D., Boulougouris, E., Cichowicz, J., and Paterson, D. (2019). Verification of damage ship survivability with computational fluid dynamics. In *17th International Ship Stability Workshop*, pages 10–12.

[Ölçer and Majumder, 2006] Ölçer, A. I. and Majumder, J. (2006). A Case-based decision support system for flooding crises onboard ships. *Quality and Reliability Engineering International*, 22(1):59–78.

[Ölçer et al., 2006] Ölçer, A. I., Tuzcu, C., and Turan, O. (2006). An integrated multi-objective optimisation and fuzzy multi-attributive group decision-making technique for subdivision arrangement of Ro-Ro vessels. *Applied Soft Computing Journal*, 6(3):221–243.

[Pantic,] Pantic, M. Introduction to Machine Learning &
}Case-Based Reasoning.

[Papanikolaou, 2001a] Papanikolaou, A. (2001a). Benchmark Study on the Capsizing of a Damaged Ro-Ro passenger Ship in Waves. *Final Report to the ITTC Specialist Committee on the Prediction of Extreme Motions and Capsize.*

[Papanikolaou, 2001b] Papanikolaou, A. (2001b). On the Stability of Fishing Vessels with Trapped Water on Deck. *Ship Technology Research.*

[Papanikolaou, 2007] Papanikolaou, A. (2007). Review of Damage Stability of Ships-Recent Developments and Trends SHOPERA. In *10th Int. Symposium on Practical Design of Ships and Other Floating Structures (PRADS).*

[Papanikolaou et al., 2003] Papanikolaou, A., Boulougouris, E., Eliopoulou, E., Alisafaki, A., Papanikolaou, A., Spanos, D., Boulougouris, E., Eliopoulou, E., and Alissafaki, A. (2003). Investigation into the sinking of the Ro-Ro passenger ferry express Samina.

[Papanikolaou et al., 2014] Papanikolaou, A., Boulougouris, E., and Sklavenitis, A. (2014). The sinking of the Ro-Ro passenger ferry SS Heraklion. *International Shipbuilding Progress*, 61(1-2):81–102.

[Papanikolaou et al., 2013] Papanikolaou, A., Hamann, R., Lee, B. S., Mains, C., Olufsen, O., Vassalos, D., and Zaraphonitis, G. (2013). GOALDS - Goal Based Damage Ship Stability and safety standards. *Accident Analysis and Prevention*, 60:353–365.

[Papanikolaou and Spanos, 2005] Papanikolaou, A. and Spanos, D. (2005). The 24 th ITTC Benchmark Study on Numerical Prediction of Damage Ship Stability in Waves Prepared for: 24 th ITTC Specialists Committee on Stability in Waves. Technical report.

[Papanikolaou and Spanos, 2007] Papanikolaou, A. and Spanos, D. (2007). International Benchmark Study On Validation of Numerical Codes for the Prediction of the Motions and Flooding of Damaged Ships in Waves. Technical report, SAFEDOR.

[Papanikolaou and Spanos, 2008] Papanikolaou, A. and Spanos, D. (2008). Benchmark study on numerical codes for the prediction of damage ship stability in waves. In *10th International Ship Stability Workshop.*

[Papanikolaou et al., 2000] Papanikolaou, A., Zaraphonitis, G., Boulougouris, E., and Eliopoulou, E. (2000). Investigation into the capsizing of damaged Ro-Ro passenger ships in waves. In *7th International Conference on Stability of Ships and Ocean Vehicles*.

[Pennanen et al., 2015] Pennanen, P., Ruponen, P., and Ramm-Schmidt, H. (2015). Integrated decision support system for increased passenger ship safety. In *Damaged Ship III*, London.

[Penttilä and Ruponen, 2010] Penttilä, P. and Ruponen, P. (2010). Use of Level Sensors in Breach Estimation for a Damaged Ship. In *5th International Conference on Collision and Grounding of Ships*.

[Petacco et al., 2024] Petacco, N., Pollicardo, A., and Gualeni, P. (2024). An insight about roll damping within the second generation intact stability criteria. *Journal of Ocean Engineering and Marine Energy*.

[Rebala et al., 2019] Rebala, G., Ravi, A., and Churiwala, S. (2019). *An Introduction to Machine Learning*. Springer International Publishing, Cham.

[Roberts et al., 2021] Roberts, S. E., Carter, T., Smith, H. D., John, A., and Williams, J. G. (2021). Forgotten fatalities: British military, mining and maritime accidents since 1900. *Occupational Medicine*, 71(6-7):277–283.

[Rodrigues, 2024] Rodrigues, J. M. (2024). A Review of Methods for Modelling Flooding, Its Progression and Outcome in Damaged Ships. *Journal of Marine Science and Engineering*, 12(2).

[Ruponen, 2006] Ruponen, P. (2006). Pressure-Correction Method for Simulation of Progressive Flooding and Internal Air flows. *Ship Technology Research*, 53(2):63–73.

[Ruponen, 2007] Ruponen, P. (2007). *Progressive Flooding Of A Damaged Passenger Ship*. PhD thesis, Helsinki University of Technology.

[Ruponen et al., 2013] Ruponen, P., Kurvinen, P., Saisto, I., and Harras, J. (2013). Air compression in a flooded tank of a damaged ship. *Ocean Engineering*, 57:64–71.

[Ruponen et al., 2010] Ruponen, P., Ltd, N., and Kurvinen, F. P. (2010). Experimental And Numerical Study On Progressive Flooding In Full-Scale. *Intl J Maritime Eng*, 152.

[Ruponen et al., 2019] Ruponen, P., Pennanen, P., and Manderbacka, T. (2019). On the alternative approaches to stability analysis in decision support for damaged passenger ships. *WMU Journal of Maritime Affairs*, 18(3):477–494.

[Ruponen et al., 2017] Ruponen, P., Pulkkinen, A., and Laaksonen, J. (2017). A method for breach assessment onboard a damaged passenger ship. *Applied Ocean Research*, 64:236–248.

[Ruponen et al., 2007] Ruponen, P., Sundell, T., and Larmela, M. (2007). Validation of a simulation method for progressive flooding. *International Shipbuilding Progress*, 54:305–321.

[Ruponen et al., 2022] Ruponen, P., van Basten Batenburg, R., van't Veer, R., Braidotti, L., Bu, S., Dankowski, H., Lee, G. J., Mauro, F., Ruth, E., and Tompuri, M. (2022). International benchmark study on numerical simulation of flooding and motions of a damaged cruise ship. *Applied Ocean Research*, 129.

[Ruth and Rognebakke, 2019] Ruth, E. and Rognebakke, O. (2019). CFD in damage stability. In *17th International Ship Stability Workshop*, pages 10–12.

[Samuel, 1959] Samuel, A. L. (1959). Some Studies in Machine Learning Using the Game of Checkers.

[Santos and Soares, 2010] Santos, T. and Soares, C. G. (2010). Probabilistic approach to damage stability. *Advanced ship design for pollution prevention*, (Taylor \& Francis Group London, UK):227–243.

[Santos et al., 2002] Santos, T. A., Winkle, I. E., and Guedes Soares, C. (2002). Time domain modelling of the transient asymmetric flooding of Ro-Ro ships. *Ocean Engineering*, 29:667–688.

[Sarker, 2021] Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions.

[Schröder-Hinrichs et al., 2012] Schröder-Hinrichs, J. U., Hollnagel, E., and Baldauf, M. (2012). From Titanic to Costa Concordia-a century of lessons not learned. *WMU Journal of Maritime Affairs*, 11(2):151–167.

[Sonne Ravn, 2003] Sonne Ravn, E. (2003). *Probabilistic Damage Stability of Ro-Ro Ships*. PhD thesis, Technical University of Denmark.

[Spanos, 2008] Spanos, D. (2008). S A F E D O R Benchmark Study on Numerical Codes for the Prediction of the Motions and Flooding of Damaged Ships in Waves Instrument: Integrated Project Thematic Priority: Sustainable Surface Transport. Technical report.

[Spanos and Papanikolaou, 2001] Spanos, D. and Papanikolaou, A. (2001). Numerical study of the damage stability of ships in intermediate stages of flooding.

[Spanos and Papanikolaou, 2009] Spanos, D. and Papanikolaou, A. (2009). SAFEDOR International Benchmark Study on Numerical Simulation Methods for the Prediction of Parametric Rolling of Ships in Wave. Technical report.

[Stening et al., 2011] Stening, M., Járvelá, J., Ruponen, P., and Jalonen, R. (2011). Determination of discharge coefficients for a cross-flooding duct. *Ocean Engineering*, 38(4):570–578.

[Taimuri et al., 2023] Taimuri, G., Ruponen, P., and Hirdaris, S. (2023). A novel method for the probabilistic assessment of ship grounding damages and their impact on damage stability. *Structural Safety*, 100:102281.

[Taveira Pinto, 2020] Taveira Pinto, F. (2020). Scaling Issues in Hydraulic Modelling.

[Tian et al., 2024] Tian, S., Peng, F., Wang, Z., and Li, J. (2024). Numerical Study of Discharge Coefficients for Side-Damaged Compartments. *Journal of Marine Science and Engineering*, 12(9):1502.

[Tsakalakis, 2012] Tsakalakis, N. (2012). *Performance-based damage survivability of passenger ships and design implications*. PhD thesis, University of Strathclyde, Glasgow.

[Turban, 1995] Turban, E. (1995). *Decision support and expert systems (4th ed.): management support systems*. Prentice-Hall, Inc., USA, 4th edition.

[van Walree and Papanikolaou, 2007] van Walree, F. and Papanikolaou, A. (2007). Benchmark Study Of Numerical Codes For The Prediction Of Time To Flood Of Ships: Phase I. In *Proceedings of the 9th International Ship Stability Workshop, Maritime Research Institute Netherlands (MARIN), National Technical University of Athens (NTUA).*

[Varela et al., 2014] Varela, J. M., Rodrigues, J. M., and Guedes Soares, C. (2014). On-board decision support system for ship flooding emergency response. In *Procedia Computer Science*, volume 29, pages 1688–1700. Elsevier.

[Vassalos, 2014] Vassalos, D. (2014). Damage stability and survivability - 'nailing' passenger ship safety problems. *Ships and Offshore Structures*, 9(3):237–256.

[Vassalos, 2015] Vassalos, D. (2015). Damage Survivability of Cruise Ships-Evidence and Conjecture. In *12th International Conference on the Stability of Ships and Ocean Vehicles*, Glasgow.

[Vassalos et al., 2016] Vassalos, D., Boulougouris, E., and Paterson, D. (2016). An alternative system for damage stability enhancement. Technical report.

[Vassalos and Guarín, 2009] Vassalos, D. and Guarín, L. (2009). Designing for Damage Stability and Survivability-Contemporary Developments and Implementation. Technical Report 5.

[Vassalos et al., 2000] Vassalos, D., Hamamoto, D., Molyneux, D., and Papanikolaou, A. (2000). The water on deck problem of damage Ro-Ro ferries. In *In Contemporary Ideas on Ship Stability*. Elsevier, 1st edition.

[Vassalos and Letizia, 1995] Vassalos, D. and Letizia, L. (1995). Formulation of a non-linear mathematical model for a damaged ship subject to flooding. In *Sevastianov Symposium: Kaliningrad, Russia.*

[Vassalos and Mujeebahmed, 2021] Vassalos, D. and Mujeebahmed, M. P. (2021). Conception and evolution of the probabilistic methods for ship damage stability and flooding risk assessment. *Journal of Marine Science and Engineering*, 9(6).

[Vassalos and Paterson, 2021] Vassalos, D. and Paterson, D. (2021). Towards unsinkable ships. *Ocean Engineering*, 232:109096.

[Vassalos and Turan, 1994] Vassalos, D. and Turan, O. (1994). *A Realistic Approach to Assessing the Damage Survivability of Passenger Ships*. Society of Naval Architects and Marine Engineers (U.S.).

[Vassalos Dracos et al., 1997] Vassalos Dracos, Turan, O., and Pawlowski, M. (1997). Dynamic Stability Assesment of Damaged Passenger/Ro-RO Ships and Proposal of Rational Survival Criteria. *Marine Technology*, 34:241–266.

[Vermeer et al., 1994] Vermeer, H., Vredeveldt, A. W., Journée, J. M. J., and Delft, T. U. (1994). Mathematical modelling of motions and damaged stability of Ro-Ro ships in the intermediate stages of flooding. Technical report.

[Vredeveldt and Journee, 1991] Vredeveldt, A. W. and Journee, J. (1991). Roll motions of ships due to sudden water ingress, calculations and experiments. In *RINA International Conference on Ro-Ro Safety and Vulnerability*, London.

[Watson and Marir, 1994] Watson, I. and Marir, F. (1994). Case-Based Reasoning: A Review. *The Knowledge Engineering Review*, 9(4):327–354.

[Zaraphonitis and Papanikolaou, 1997] Zaraphonitis, G. and Papanikolaou, A. (1997). On a 3-D mathematical model of the damage stability of ships in waves GOUTCAT View project PRELIMINARY DESIGN OF A SWATH ROPAX FERRY View project. Technical report.

[Zhang et al., 2019] Zhang, X. L., Lin, Z., Mancini, S., Li, P., Li, Z., and Liu, F. (2019). A numerical investigation on the flooding process of multiple compartments based on the volume of fluid method. *Journal of Marine Science and Engineering*, 7(7).