AALBORG UNIVERSITET
ESBJERG

CLASSIFICATION OF INFORMATION TYPES

IN DECISION ANALYSIS

Author: Peter Mikkel Frederiksen, 20160754, pfre21@student.aau.dk
Supervisor: Prof. Michael Havbro Faber, Aalborg Universitet

May 2023

Title page

Project title:	Classification of Information Types in decision analysis
Educational institution:	Aalborg University Esbjerg
Education:	Master of Science in Risk and Safety Management
Semester:	Fourth Semester
Project period:	Spring 2023
Supervisor:	Michael Havbro Faber
Author:	Peter Mikkel Frederiksen
ECTS Points:	30 ECTS Points
Number of pages:	64 pages
Wordcount:	20870 words
Appendix:	Appendix 1 – Search words Appendix 2 – Table for literature review Appendix 3 – Runge-Kutta Methods Appendix 4 – Runge-Kutta single iteration Appendix 5 – Data Appendix 6 – R-Code

Preface

This project is the master's thesis of the Risk and Safety Management program at Aalborg University. The project covers 30 ECTS points and is the concluding project for the 2-year master's programme at Aalborg University Esbjerg.

The main purpose is to explore the influence of information states in a decision context. This is facilitated by Bayesian decision analysis in combination with different information taxonomies. The thesis presents a theoretical decision problem as an exemplification of using a machine learning classification to classify information states in a decision context.

All figures are made by the author, some of which are heavily inspired by JCSS (2008) and Glavind et al. (2020).

The citations follow APA guidelines, apart from few, where the original journal article or book have been unavailable. In such cases, they are cited through the source of context of which they have been found. I.e., It has not been possible to obtain a copy of Feltham's "The Value of Information (1968)". However, main points have been introduced in Zhang et al. (2021). The content is then kept to what is available in Zhang et al. (2021), but cited as "Feltham (1968, as cited in Zhang et al. 2021)".

Acknowledgements

The original idea of exploring system representations through big data techniques is introduced by Sebastian Glavind and Michael Faber. Sebastian and Michael inspired me to pursue some of the "further work" described in Sebastian's Ph.D. thesis. This project is the final product of that pursuit. During the project, I've had many valuable discussions with Wei-Heng Zhang, who has devoted much of his time to discussing different classification techniques as well as thoroughly introducing me to the different aspects of Bayesian decision analysis and the Value of Information concepts. Thank you Wei-Heng, Michael, and Sebastian.

I am thankful to my loving girlfriend, who has been supporting me throughout the entire process. She has been extraordinary when times have been tough and her belief in my capabilities provided the boost I needed to carry on. Even though her professional and scientific background is completely different, she provided much-appreciated feedback and discussions throughout the project – thank you.

Abstract

This project focuses on how a supervised classification method can be implemented in a decision-making framework and how this can provide knowledge about the expected utility of the classification, with this also misclassification of Type I and Type II errors.

The problem is exemplified through an experimental platform of a simple pendulum. The pendulum is a physical system that exhibits a dynamic behavior over time that depends on several characteristics. A pendulum might appear to be a somewhat artificial system, but depending on the chosen system parameters may represent many phenomena of socio-ecological systems, such as climate variations, variations in traffic, or responses of structural and mechanical systems.

A probabilistic model is developed for the representation of the different information states. The already developed database of realizations of system parameters and system behaviors is augmented to reflect observations of the system behaviors, including the effect of information states based on the probabilistic model.

The experimental platform establishes a foundation for a machine-learning classification of the observations into different information states. The information states will be predicted with adequate precision to reflect the decision-makers preferences regarding Type I and Type II errors. The choices involved in the above-outlined approach can be identified by utilization of the concept of Value of Information. The Value of Information may now be applied to obtain the Value of Classification, based on the expected value gain of utilizing a classification scheme.

Table of Content

List of Figures

List of Tables

1 Introduction

In the domain of reliability engineering and system safety, there is a common problem in determining the state of a system. This difficulty arises from having only a limited amount of information about the system's performance, thus making it challenging to define its exact state. Moreover, there are usually many potential states that the system could be in, including ones where the system is damaged. These different states can all offer plausible explanations for the observed data, meaning that different competing damage states can be observed within the same data.

Glavind et al. (2022) proposed a method for managing systems by utilizing big data techniques to address this challenge. The method involves creating a probabilistic representation of the system along with the data that can be gathered from it. The aim is to establish a representation that is consistent with the available knowledge. In their study, Glavind et al. (2022) applied big data classification techniques to simulated scenarios to assess the probabilities of different system states based on specific observations. Specifically, they utilized a clustering algorithm to analyze the scenarios with the objective "to identify and understand possible regularities in the realizations of random variables related to particular system performances of interest. "

System management broadly relies on the available knowledge about the system. Models representing their behavioral characteristics are created based on the best available information. These models enable analyses of system behavior over time and provide support for decision-making. Uncertainties often accompany available knowledge, and this uncertainty may effectively be addressed in a probabilistic model.

Understanding the origin of observations is vital for making informed decisions. For instance, observed accelerations of a wind turbine might provide insights about its structural integrity and potentially reveal signs of damage. These observations can aid decision-making concerning loss prevention. However, misinterpretation of these observations could result in unsuitable actions and subsequent losses.

To further understand the source of an observation, an experimental case study is set up. A simple pendulum serves as the focal point for this study. This physical system exhibits dynamic behavior over time, depending on various characteristics. These include the mass of the pendulum, its length, initial conditions, and damping. Although a pendulum may seem like an artificial system, it can represent a wide range of socio-ecological phenomena based on the chosen system parameters, such as climate changes, traffic fluctuations, and responses of structural and mechanical systems, among others.

The case study outlines a theoretical decision problem. The pendulum may or may not oscillate too much and the decision-makers can choose to stop the pendulum but at a cost. The information available may or may not be precise or even relevant. The question is then, how is it possible to make a decision based on an unknown state of information?

The following project sets out to classify different information obtained from the system. This information may be biased, or it may contain noise, or maybe the information does not even derive from the system which is model in the prior model.

1.1 Problem formulation

The problem at hand is to determine the state of a system, where limited information about the system's performance is available, making it challenging to define its exact state. The challenge is further compounded by the existence of multiple potential states, which can offer plausible explanations for the observed data. This may complicate decision-making as it may not be obvious where the observation from the system is derived from if they are biased, anomalous, outliers, or if they are from a completely different system.

In this context, Glavind et al. (2022) proposed a method for modeling and analyzing systems by utilizing big data techniques, where a probabilistic representation of the system and its performance indicators is created. However, this method assumes that all potential states of the system have been considered, which may not be practical or realistic. Therefore, the objective of this project is to classify system states that are in the probabilistic representation of the system for decision-making. To explore this problem, the following research question was asked:

1.2 Research question

How can classifications be utilized to identify information states of systems in a probabilistic representation of a decision?

The question essentially asks how a classification algorithm can be used to identify different kinds of information from observed data, and then how that classification can be utilized in probabilistic decision analysis.

The objective is to develop and exemplify a scheme for the optimal use of information types. To specify this the following sub-questions have been asked:

- How do the different types of information affect a system's performance?
- How can a classification scheme be integrated into the probabilistic representation of a decision?
- How can different types of uncertainties be incorporated into the probabilistic decision models?

- How can the output of the classification model contribute to decision support, such as enhancing the understanding of system behavior and aiding in risk management?

1.2.1 Decision problem

For the above questions to be answered and exemplified, the following assumed decision problem has been set up:

The pendulum (described in Chapter 3 System Identification) becomes fragile if it oscillates too much. If the pendulum oscillates beyond -42º, i.e., the mass of the pendulum is at an angular position of -42º or less, there will be excessive damage to the rest of the pendulum system. The pendulum should therefore not oscillate beyond -42º.

In order to avoid this, the decision-maker can perform one of two actions:

- They have the possibility to stop the pendulum, leading to the outcome of "a"
- They also have the possibility to do nothing.
    - If they do nothing there is a possibility for the pendulum to oscillate beyond the -42º(and thereby cause damage to the system), leading to outcome "b". Another possibility is that the pendulum will stay within the safe operating domain of -42º and do no harm to the system, leading to outcome "c".

Each of the three outcomes is associated with a cost, which defines the decision-maker's preference. The decision-maker prefers "a" over "b" and "c" over "a". This can be expressed as a function of utility by:

$$u(c) > u(a) \ > u(b) \qquad \textit{Eq. 1.1}$$

The decision problem is explored through Bayesian decision theory (presented in Chapter 4 Theory). The decision theory provides a framework for consistently account for uncertainties as well as the possibility to consistently rank different decision alternatives.

A classification algorithm (presented in Chapter 4 Theory) is then used to classify the information obtained from the pendulum. This classification provides knowledge about the state of the information, and thereby the state of the pendulum.

1.3 Reader's guide

Figure 1 provides a visualization of the work in chronological sequence to aid the reader's comprehension of the project. The project starts with a literature review of the available knowledge of how classification algorithms are used in structural engineering and decision-making in the literature. Based on these findings, the project was planned, hereby also which methods to use in terms of classification and how to include them in a decision-making framework.

When the topic and the boundaries for the knowledge domain are known, it is possible to identify the principal system for which the decision revolves. The system identification will present the pendulum as a case study while also presenting different taxonomies of information types and how the data is obtained and manipulated.

The description of methods can be seen as an extension of the literature review and the system identification. The identified method of classification will be presented along with a description of the Bayesian decision analysis framework.

When the system is identified, along with the methods, it will all be put together in the creation of the decision analysis. The decision model will be exemplified with the case study presented in the system identification. The results will be presented and evaluated followed by a short discussion of information types and caveats of the project before presenting the conclusion.

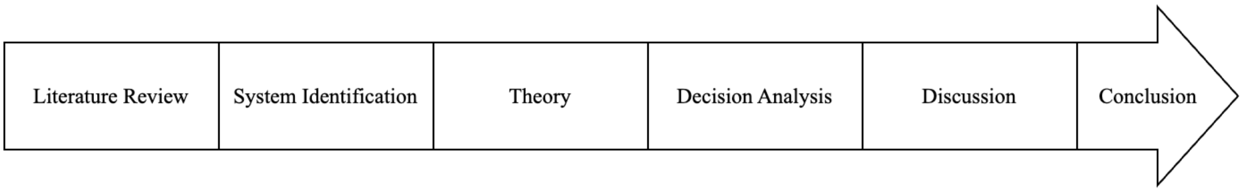


Figure 1 Structure of the project

2 Literature Review

A literature review of the knowledge domain of *"classification of information and system states"* is conducted to structure the preliminary research. The objective of the literature review is to explore and identify the current state-of-the-art literature. This exploration will contribute to a fundamental understanding of the inter- and multidisciplinary domain of classifying information states. The of Science (WoS) is used as a database to search for literature.

The literature review was conducted in two steps:

1. A bibliometric analysis utilizes a text mining application, VOS Viewer, to create a graphical network to visualize recurring keywords in the input text. This provides a greater overview of multiple different research items.
2. A state-of-the-art review is a systematic readthrough of literature to explore how different classification methods have been used previously. Finally, specific techniques for classifying information types are identified based on the bibliometric analysis and the state-of-the-art review.

2.1 Bibliometric analysis

The VOS-viewer produces graphical networks of clusters based on bibliometric data retrieved from WoS. The clusters consist of nodes representing keywords in the literature, and edges, representing the co-occurrence in the same or different literature. The thickness of the edges symbolizes the strength of the relationship or the degree of association between two nodes. For instance, a thicker edge denotes a higher frequency of co-occurrence. This interpretation of node and edge characteristics within the graphical networks allows for a comprehensive understanding of the underlying patterns and relationships in the bibliometric data (van Eck & Waltman, 2014).

The interpretation of the clusters is largely dependent on prior knowledge, effectively requiring an active engagement in the analytical process. Nevertheless, drawing from the Socratic principle of "following the argument where it leads," the interpreter can make sense of the clusters. For instance, if a cluster contains terms like "structural health monitoring," "damage detection," and "diagnosis," it can be interpreted as representing the knowledge domain of "Structural health monitoring." However, this interpretation relies on the context provided by the other connected clusters.

The analysis is an iterative process inspired by a circular hermeneutic approach. The process continues until a satisfactory list of literature representing the knowledge domain of the classification of information and system states is obtained. The process involves the following steps:

1. Identification of search terms and grouping them,
2. Apply the queries to Web of Science to get search results,
3. Visualizing the results in VOS-Viewer, and

4. Interpreting the graphs to analyze the findings.

The iterative process stops when the areas of interest are sufficiently displayed in the visualization or the count of items from the VOS Viewer is approximately less than 150. This makes it possible to evaluate the items individually in a state-of-the-art review.

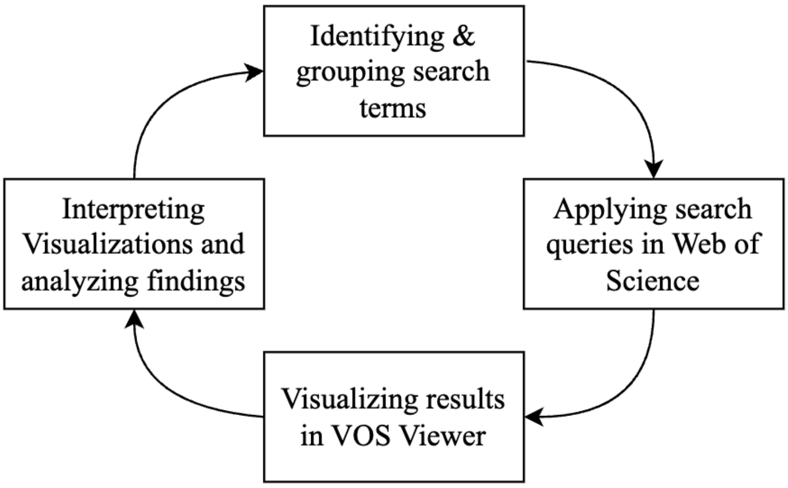


Figure 2. The iterative process of the bibliometric analysis

2.1.1 Identifying search terms and groupings

Firstly, search terms are selected through a discussion and brainstorming session with master student peers, i.e., study group, supervisor, and co-supervisor. Then, the search terms are chosen to describe an overall subject, e.g., for a search group of “Model uncertainty,” search terms such as “uncertainty modeling,” “aleatory,” and “epistemic” was used. Each search term in the group is paired with an OR-operator, allowing for items to be found if they include one or more search terms.

Another group called “Decision Support” is made to gain insights into decision-making research. Multiple groups can be paired to form search queries, e.g., “Decision support” can be paired with “Model uncertainty.” This search query can provide information on the combined knowledge domain of model uncertainty in decision support. Each group is paired with an AND-operator, which allows for items to be found only if they include at least one search term from each of the paired groups.

In Table 1. An overview of the different search groups can be found. The complete list of search terms corresponding to each of the groups can be found in Appendix 1, Search Words.

Group name	Group description
Model uncertainty	This category focuses on the quantification and modeling of uncertainty in probabilistic models. It includes techniques for characterizing different types of uncertainty and for incorporating uncertainty into risk assessments.
Probabilistic system representation	This category refers to methods for representing the behavior of systems in terms of probabilistic models. These models can be used to assess the risk of system failure and to identify potential sources of uncertainty.
Decision support	This category refers to methods for using probabilistic models and other data to support decision-making in complex systems. It includes techniques for assessing risks, identifying potential problems, and making informed decisions based on available data.
Outlier detection	This category refers to techniques and algorithms, such as classification, used to detect unusual or outlying information in data or systems. These techniques can be used in a wide variety of applications.
Reliability engineering & system safety	This category focuses on the analysis and design of systems to ensure their reliability and safety. It includes techniques for detecting and diagnosing faults, as well as methods for assessing the risk of system failure
Evaluation and comparison of classification algorithms	This category refers to methods for evaluating and comparing the performance of different classification algorithms. These methods can be used to assess the accuracy, effectiveness, and efficiency of different techniques, and to identify their strengths and weaknesses.

Table 1. Description of search groups

2.1.2 Apply queries to Web of Science.

Multiple search queries were established and used as initial input into the Web of Science. The different queries were built based on questions related to how classification is currently being used to address information states.

The project is interested in identifying research that addresses the use of classification algorithms for decision analysis while also considering the potential sources of model uncertainty associated with probabilistic modeling techniques. The model uncertainties are also included to represent any possible uncertainty quantification methods in engineering and system safety. This will contribute to identifying the interdisciplinarity of classification of information states and uncertainty quantification.

The following query focuses on identifying research on how classification of information states are used in conjunction with probabilistic modeling techniques to better represent the uncertainty and risk associated with systems:

Query 1: "Model uncertainty" AND "Probabilistic system representation" AND "Outlier detection"

The third query focuses on identifying research that evaluates and compares different classification algorithms in terms of their performance, accuracy, and effectiveness. This contributes to understanding the various methods and which classification algorithm might be the most relevant in the specific case example presented in this project. The third query is thereby established using the following groups:

Query 2: "Outlier detection" AND "Evaluation & Comparison of Classification Algorithms"

The third search query is established to identify research that explores how classification can be used in decision support within safety and reliability engineering. While this query is specific, it is the basis of this current project, as this is where this project is seen to be able to contribute. Whereas the previous queries contributed to the identification of the use of classification, this query focuses on how it can be used as decision support:

Query 3: "Decision support" AND **"**Outlier detection" AND "Reliability engineering & system safety"

Top 5 WoS categories for the four search queries		
Query 1 (n = 1340)	**Query 2** (n = 1329)	**Query 3** (n = 152)
Statistics Probability	Engineering Electrical Electronical	Engineering Electrical Electronic
Engineering Electrical Electronic	Computer Science Information Systems	Computer Science Artificial Intelligence
Computer Science Artificial Intelligence	Computer Science Artificial Intelligence	Operations Research Management Science
Computer Science Information System	Computer Science Theory Methods	Computer Science Information Systems
Computer Science Theory Methods	Telecommunications	Computer Science Theory Methods

Table 2. Top five categories from the Web of Science for each of the three search queries

In Table 2, most of the search results relate to the overall knowledge domain of "Computer Science" and, within that, the sub-domain of either "Information Systems" or "Artificial Intelligence." This is as expected as all the search queries contain the search group of "Outlier detection," which is derived initially from classification, thus statistics and computer science.

2.1.3 Visualizing results in VOS Viewer

Various parameters can be adjusted to create a graphical network using the VOS Viewer to display bibliometric data. Once the .txt-files from WoS are imported into VOS-Viewer, the parameters can be adjusted in any way believed to contribute to the optimal amount of information. For this report, a full counting option of term co-occurrence was chosen, meaning that each term has the same weight independent of the author(s), citations, or other bibliometrics and will not be fractionalized.

The VOS-Viewer lists co-occurring terms that can be modified for a better fit. However, it can often be found that terms are essentially the same and can be written differently. Sometimes it is because of abbreviations, and other times it is because of hyphens, e.g., “support vector machine,” “support-vector-machine,” and “SVM.” The VOS Viewer is unable to make the distinction that these terms refer to the same machine learning algorithm. Thus, this is changed manually to make a better fit for a graphical representation of the bibliometric data (van Eck & Waltman, 2014).

Some terms are left out as they do not contribute to an understanding of the knowledge domain, such as “model,” “algorithm,” and “data”. This can however affect the overall map. Therefore, it is essential to focus on the item count, which influences the number of links and the total link strength. In contrast, the minimum number of occurrences of a term and the item count affect the number of terms available to be selected and the number of clusters. The table below shows the number of queries, the chosen parameters, and their associated output.

Parameter settings for each of the three maps associated to Query 1, 2, 3						
	Item Counts	**Min. no. of term co-occurrences**	**Final number of terms selected**	**Number of clusters**	**Number of links**	**Total link strength**
Query 1	36	3	25	5	70	109
Query 2	32	5	25	4	136	488
Query 3	41	5	34	5	231	428

Table 3 The parameters of the VOS-viewer for the three different queries

Table 3 lists several different settings used in the queries.

- "Item counts" is the number of items used to make the visualization.
- The "minimum number of co-occurrences" is the smallest number of times terms need to appear together within the search result from the WoS.
- "The final number of selected terms" shows how many terms is selected after the list of items is adjusted.

The last three details, which are the number of clusters, links, and total link strength, are results of the process and can't be changed as they depend on the input data and parameters (van Eck & Waltman, 2014).

2.1.4 Interpreting visualizations and analysis of results

The following visualizations have been reiterated and only the final version of the visualization is presented. In addition, visualizations containing too much unstructured information have been reiterated, as described at the beginning of the literature review.

2.1.5 Query 1, Probabilistic system representation AND Outlier detection AND Model uncertainty

Query 1 focuses on how classification and outlier detection algorithms have been utilized with probabilistic modeling techniques to better represent uncertainty and risk. The visualization shows distinct clusters separated. The distance between clusters represents the similarity between clusters. Thus, some research domains are closer related than others, e.g., the concept of fisher information is far away from novelty detection; it can therefore be concluded that fisher information is sparsely utilized in novelty detection.

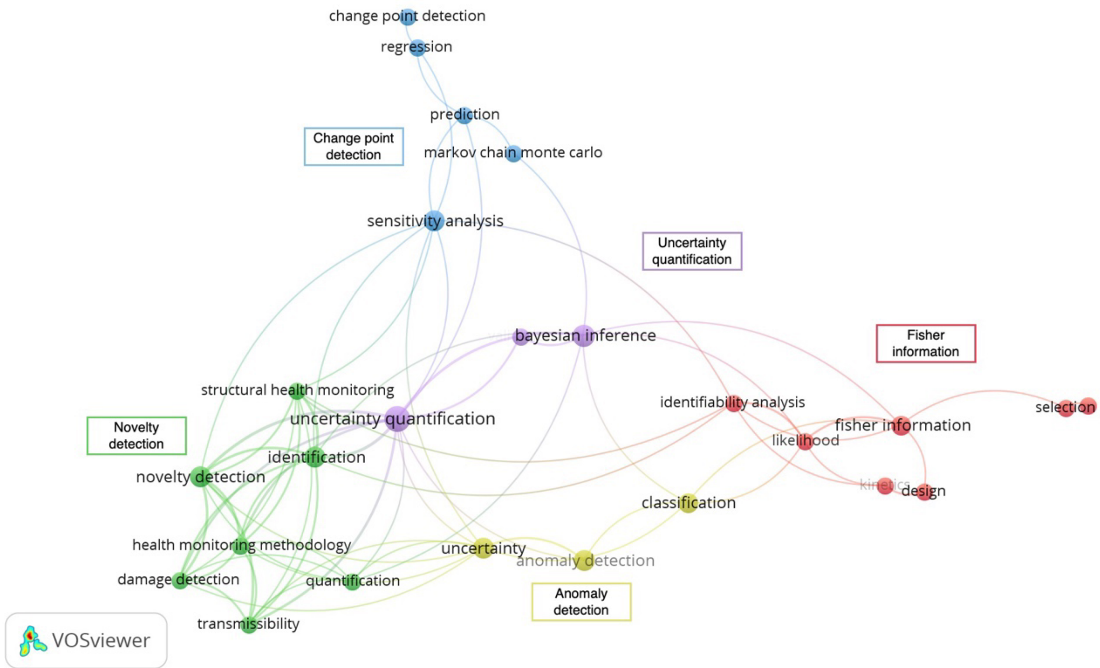


Figure 3 Query 1, Anomaly detection AND Probabilistic system representation AND Model uncertainty

The blue cluster is associated with change point detection. Change point detection is a statistical method used to identify a series of points or periods where there is a change in the underlying properties of a data set or signal. It can be inferred from the visualization that this is often a regression task rather than a classification task.

The red cluster contains terms related to the concept of fisher information. Fisher information measures how much information a statistical model contains about an unknown parameter. Therefore, it may also have applications in classifying information states.

The yellow cluster of Anomaly detection is related to classification. Anomaly detection often utilizes a classification algorithm to detect changes in data. This includes supervised, semi-supervised, and unsupervised learning problems, which will be described later in this chapter.

The green cluster primarily concerns novelty detection in structural health monitoring, i.e., completely new system states observed from the data. However, from the visualization, it can be observed that much of the novelty detection is related to detecting damage within physical structures. This suggests novelty detection has applications within structural engineering and the built environment.

The purple cluster of uncertainty quantification is placed in the middle of the visualization, suggesting that all the other clusters are related to uncertainty quantification. This relation contributes to understanding how uncertainty modeling, in terms of anomaly- novelty- or point change detection, can be applied to reduce model uncertainty, ultimately contributing to a better system understanding.

This visualization displays different algorithms to detect system changes, and it can be concluded that all the different detection algorithms have uncertainty quantification in common. This visualization also contributes to understanding how the detection algorithms contribute to probabilistic modeling. It provides the overall method for the different algorithms, e.g., classification algorithms for anomaly detection and regression algorithms for change point detection.

2.1.6 Query 2, Evaluation and comparison of Classification methods AND Outlier detection

The objective for the third query is to identify research that evaluates and compares different classification algorithms in terms of their performance, accuracy, and effectiveness. This will contribute to understanding the different possible algorithms and which method might be the most relevant to use in the specific case example presented in this project.

The red cluster consists of different terms related to IT security and the governance thereof, e.g., “cyber security,” “intrusion detection,” “smart grid,” and “internet of things.” All of which also are connected to “machine learning.” This connection may be made because machine learning can contribute to solving different IT-security-related tasks. This contributes to understanding the use of anomaly detection, e.g., that auto-encoders are applied in a cyber security domain. Still, it only slightly contributes to evaluating the different algorithms.

The yellow and blue clusters are closely related as both concern classification problems. For example, the blue item of one class classification is closely related to support vector machines. One class classification is an algorithm belonging to the collection of semi-supervised learning methods. The three items in the yellow cluster all share some of the same ways of measuring the distance between two data points in a high-dimensional feature space. At the same time, the blue cluster includes ensemble learning, a term often related to the algorithm of Random Forests where different decision trees are built. The output would be an average of the different decision trees, thus an ensemble of trees.

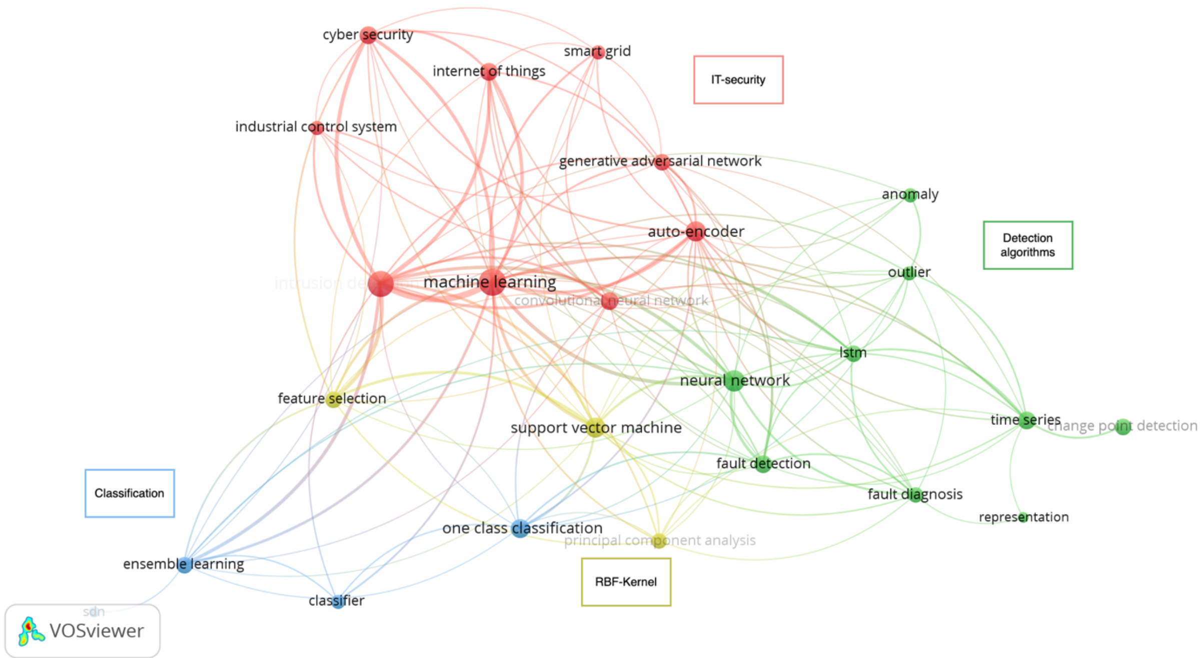


Figure 4. Query 2, Outlier detection AND Evaluation and comparison of methods

The green cluster contains many of the different terms relating to the actual task of outlier detection, e.g., "anomaly," "outlier," "change point detection," and "fault detection." In addition, many of the different detection categories relate to "neural network," which is often used as a classification algorithm and a clustering algorithm.

Query 2 does little to provide a better understanding of how to evaluate and compare the different detection methods but shows that the methods are primarily related to classification tasks, hereby mostly ensemble methods.

2.1.7 Query 3, Reliability Engineering & System Safety AND Outlier detection AND Decision support

The third search query is established to identify research that explores how outlier detection, thus classification, can be used in decision support within system safety and reliability engineering. This query is particular as it attempts to combine and display the intersecting domains this project contributes to. It accounts for the current applications of classifying information in a decision context relating to system safety and reliability engineering.

The green cluster contains items that relate to the diagnosis of engineered systems. It does so in terms of "diagnosis," "identification," and "detection." These items can be interpreted as means of decision support as they contribute to a better understanding of the system state. The cluster also contains neural networks and some detection methods, indicating machine learning use within those processes.

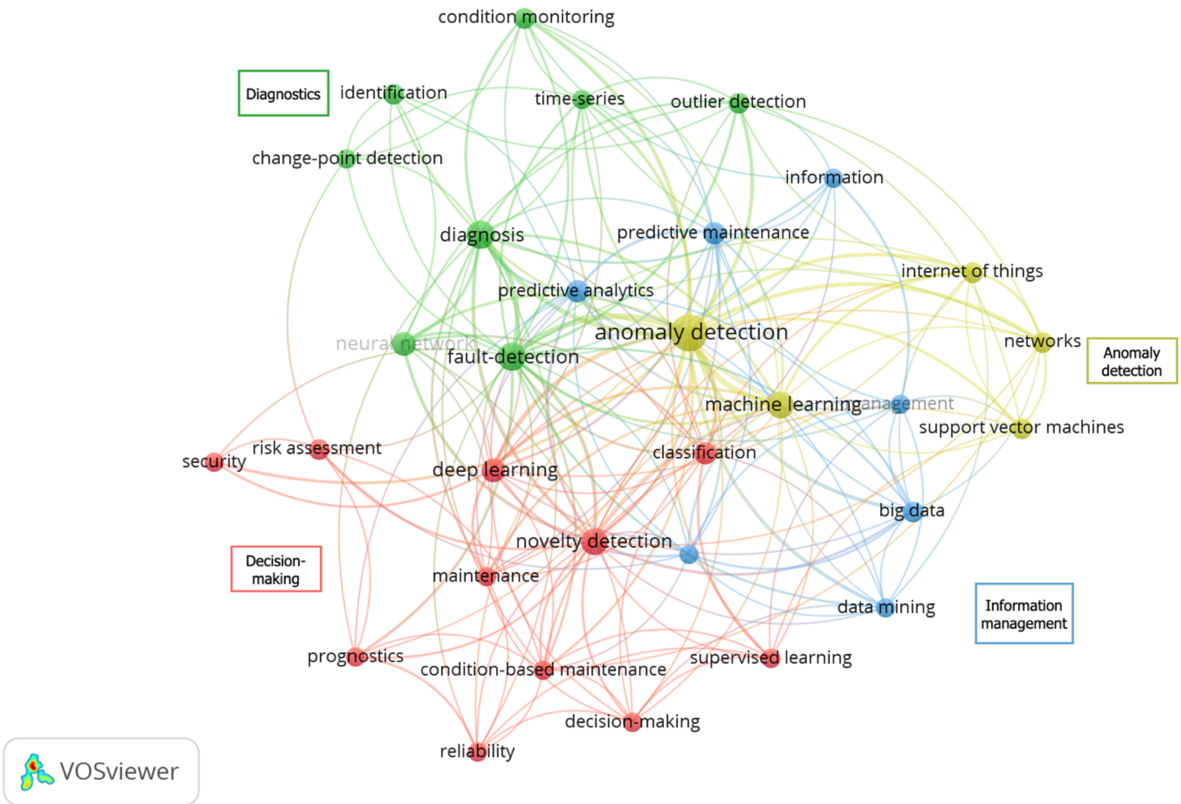


Figure 5. Query 3, "Outlier detection" AND "Reliability Engineering & System Safety" AND "Decision support"

The red cluster contains information on decision-making's role in engineered systems. Even though the actual node of "decision-making" is relatively small, many of the other nodes in the red cluster relate to the knowledge domain of decision-making in the context of engineered systems, e.g., "risk assessment," "maintenance," and "reliability." This cluster indicates that novelty detection for decision support in engineered systems has already been applied. Specific applications will be reviewed in the state-of-the-art review.

The yellow cluster is the smallest but contains one of the most influential nodes of "anomaly detection." Anomaly detection is a term that covers the detection of unusual information in a dataset, whether that is a novel system state or just outlying data. Machine learning is then closely related to anomaly detection but also shares the cluster with the support vector machine, suggesting a specific algorithm for the classification of information.

The blue cluster relates to the governance of data and the management of information. This is an important cluster to examine, as the governance of systems is highly influenced by the decision-making domain (Nielsen, 2007). Furthermore, it can be observed that the cluster is spread throughout most of the network. This suggests that the cluster's domain is interdisciplinary and that other clusters are influencing the domain. E.g., "management" is close to "machine learning" in the visualization.

The visualization of query 3 provides an understanding of the possible knowledge domains in which the classification algorithms can contribute to decision support, e.g., risk assessments, predictive maintenance, and condition monitoring. This is in accordance with what is expected based on the statements of Glavind (2021), where it is stated that anomaly detection is an important research domain to improve the knowledge of structural health monitoring.

2.1.8 Summary of the bibliometric study

Awareness of the inherent subjectivity of interpreting bibliometric visualizations is essential. The VOS Viewer provides an overview of trends and patterns in the literature. This makes it easy to grasp the integration of different knowledge domains, such as the intersection of the classification of information, probabilistic system representation, and model uncertainty.

From the first visualization, it can be concluded that classification algorithms are commonly used for outlier detection. In contrast, regression algorithms are more widely used in a change point series. The second visualization provides information on the importance of classification for detecting new, unseen information through ensemble methods. Finally, the third visualization provided examples of the possible knowledge domains in which the classification algorithms can provide decision support.

A criticism that can be put forth is the difficulty, if not impossibility, of stating the causality of the terms. E.g., even though it intuitively makes sense that machine learning must be managed, it cannot be concluded as the causality might as well be reversed for instance machine learning helps in managing systems.

The bibliometric study provides a starting point for classifying information types. In the next part of the literature review, a state-of-the-art review explores different possible algorithms for classifying information of systems is presented.

2.2 State-of-the-art Review

This state-of-the-art review assesses the current state of research on different possibilities of classifying information states, including different applications, challenges, and recent advances. In addition, the review will examine various techniques and approaches proposed in the literature along with their strengths, weaknesses, and limitations.

The visualization of query 3, Figure 5, provides an understanding of the possible knowledge domains in which the classification algorithms can provide decision support, e.g., risk assessments, predictive maintenance, and condition monitoring. This is in accordance with what is expected based on the statements of Glavind (2021), where he states that classification algorithms are an important research domain to improve the knowledge of structural health monitoring.

The state-of-the-art review is done by reading through all the articles of query 3 and structuring them by listing and comparing the research content, as seen in Table 4 (the entire table can be found in Appendix 2 Table for Literature Review). The table is created such that:

- The first column can be used to identify and backtrack the research. This is made possible with the digital identifier number and the article's name.
- The second column concerns the classification algorithm of the research. First, the names of the method are listed, along with a short description in the case of a novel method.
- The third column relates to the context of the proposed method. This row relates to the application domain of the research. This domain can be as broad as “structural health monitoring” and as narrow as “temperature monitoring of bearings.”
- The fourth column concerns the evaluation of the method. It is important to not just rely on terms such as “accuracy” and “precision”, as it is of essence to understand the outcomt of the algorithm to evaluate the performance. Therefore, exploring different evaluation methods to obtain the most desired results is relevant.
- The fifth column is regarding the decision-making aspect and how the methods can be used in a decision-making context. This is an essential aspect of the research in question, as classification must fit into a decision-making context and contribute to the facilitation of ranking of decision alternatives.
- Lastly, the sixth column is a column that can hold any additional relevant information.

Extract of state-of-the-art review table					
Article DOI	Classification method	Application domain	Evaluation of method	Decision-making	Other
The Robust Multi-Scale Deep-SVDD Model for Anomaly Online Detection of Rolling Bearings https://doi.org/10.3390/s22155681	Deep support vector data description, Deep-SVDD. Deep SVDD employs a deep neural network to learn a high-level representation of the data and then uses the SVDD algorithm to fit a hyper-sphere in the learned feature space.	Prognostics and Health Management (PHM) for rolling bearings	Anomaly score threshold based on the maximum value of the training data score.	Purely a technical article.	Possibility of implementing deep-svdd
…					
Anomaly Score-Based Risk Early Warning System for Rapidly Controlling Food Safety Risk https://doi.org/10.3390/foods11142076	Auto-Encoder (AE) based anomaly detection algorithms	Food safety risk assessments	Comparison with baseline models such as Isolation Forests, KNN, LOF, and K-Means. And then compared the fault detection rate, The false alarm rate, The area under the curve (AUC), accuracy.	Not much emphasis was being put on the decision-making part. A single sentence is referring to how this can be used by regulating authorities to strengthen the supervision of relevant food manufacturers.	Comparison of baseline models

Table 4. Extract of State-of-the-art literature review table

2.2.1 Results of the state-of-the-art review

When exploring the classification of information, it can be favorable to take a step back and start by looking at machine learning in general. This provides some background information that is beneficial to understanding how different methods relate to each other. The following provides an overview of the different classification techniques and methods and a short description of how it is used in the literature. To better illustrate the relationships between the various classification methods and machine learning categories, Figure 6 has been created.

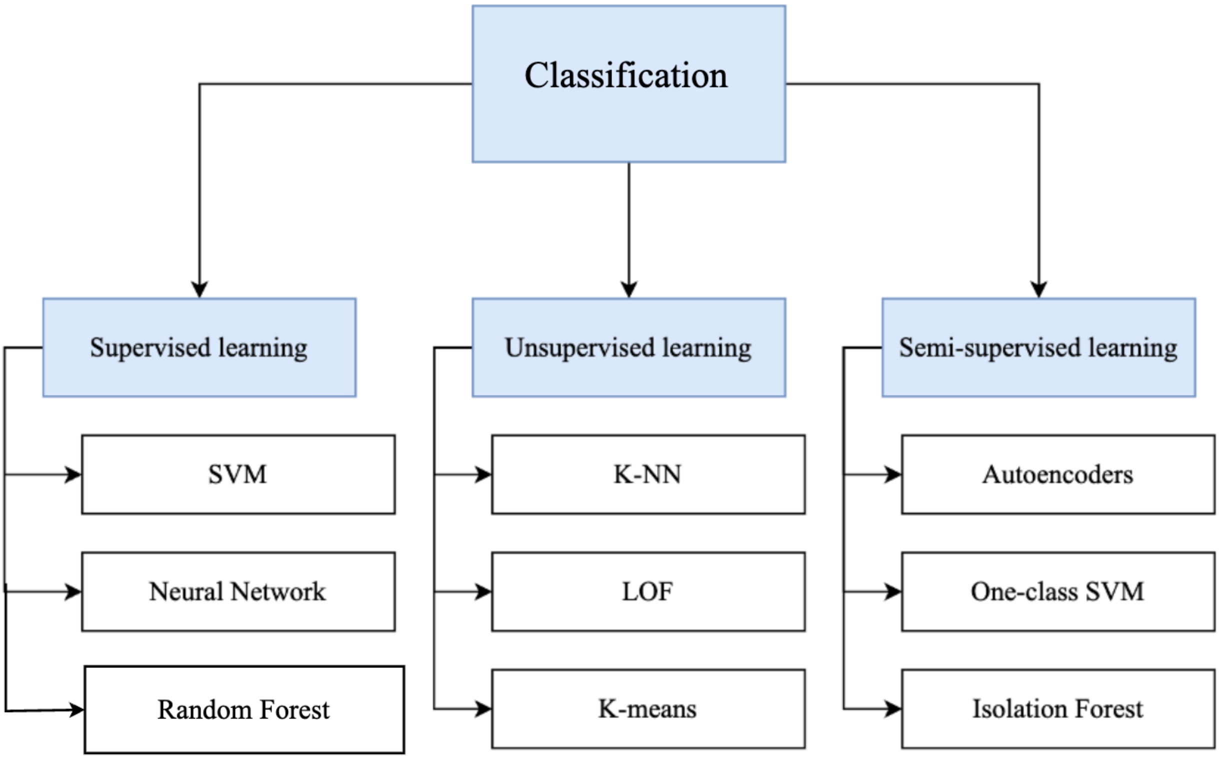


Figure 6. The different categories of classification within different machine learning domains

2.2.1.1 Supervised learning

Supervised learning problems refer to applications where the training data includes input labels and their corresponding output label. Support vector machines, SVM, Neural networks, and Random Forest are within the supervised learning domain. SVM is the most popular multi-class classification method (Sharma et al., 2022). The SVM seeks to find a hyperplane with the largest margin of separation between classes (James et al., 2021). SVMs have been found useful in the prognostics of rolling bearings and condition-based maintenance (Sharma et al., 2022).

Neural networks take an input vector and build a non-linear function to predict a response variable (James et al., 2021). As it is observed in the bibliometric study, Neural Networks are a prevalent technique for fault detection of systems. Within the state-of-the-art review, Neural Networks have been found to contribute to fault detection in rotating machinery (Crupi et al., 2004) and real-time anomaly detection for marine machinery (Velasco-Gallego, & Lazakis, I., 2022).

Random Forest is a machine learning model primarily employed for regression and classification problems. It operates by constructing numerous decision trees during training and outputting the mode of the classes for classification or the mean prediction for regression. The model effectively harnesses the power of 'the crowd,' hence the term 'forest,' to achieve robust and accurate predictions. The learning process involves using a method known as bagging, where different subsets of the data are used to train each tree. This process helps mitigate the risk of overfitting, thus improving the model's generalization to unseen data. (James et al., 2021). Random Forests have found usage in Structural Health Monitoring (SHM) systems, which have been deployed to monitor vital civil infrastructure, such as long-span bridges (Arul & Kareem, 2022). Zhou et al. (2012) utilized Random Forest for damage detection on an eight-story steel frame structure.

2.2.1.2 Unsupervised learning

Unsupervised learning problems refer to applications where the training data includes input vectors' observations but lacks corresponding output vectors. The algorithm must therefore identify patterns and structure in the data without explicit guidance on what the outputs should be. Further, all identified unsupervised learning techniques belong to the "distance-based outlier detection" category. Distance-based anomaly detection methods operate under the assumption that normal data points are located within a dense region while anomalies are far from their nearest neighbors. Although K-nearest neighbor, KNN, is originally a supervised machine learning approach, it is employed as an unsupervised technique for anomaly detection due to the absence of any predefined inlier or outlier class (James et al., 2021). Instead, KNN in an anomaly detection setting is based on a threshold-based approach. The central tenet of KNN is that similar data points tend to cluster together, while anomalies are situated far away from the clusters of similar data points. The KNN algorithm has been found to be useful for fault detection for reciprocating compressors (Patil et al., 2022).

The local outlier factor (LOF) is an unsupervised density-based anomaly detection technique that aims to identify local anomalies. Specifically, LOF determines a data point's local density variation based on neighboring points' density. Data points with a lower density than neighbors are classified as anomalies (Demetriou et al., 2022). LOF is generally not the best algorithm for anomaly detection compared to other easily accessible methods. However, it has successfully detected abnormal railway erosion (Accorsi et al., 2017).

K-means is an unsupervised method for grouping data points into distinct clusters based on similarity. The number of clusters, denoted by K, is predefined by the user. The objective is to assign each data point to a cluster such that the points within the same cluster share similar properties. Each cluster is associated with a centroid, representing the average of all data points assigned to that cluster. The algorithm iteratively updates the centroids until convergence (James et al., 2021). To detect outliers, a threshold value is added to the algorithm. A data point is considered an outlier if its distance from the nearest centroid exceeds the threshold value. The threshold value is chosen based on the specific problem and can be adjusted to increase or decrease the sensitivity to outliers. Zhang & Ma (2016) utilized a K-means clustering algorithm for fault detection as a part of a condition-based monitoring system for wind turbines. Zhang et al. (2019) also used the K-means algorithm for fault detection in a water source heat pump air-conditioning system.

2.2.1.3 Semi-supervised learning

Semi-supervised learning refers to problems where the training data comprises only normal data but lacks corresponding output vectors. The fundamental concept is to learn a model of the normal class and then identify output vector classes by detecting deviations from that model. The algorithm will thereby have to identify patterns in the data, but with the constraint that only the "normal" pattern is learned.

Autoencoders are a neural network consisting of an encoder mapping the input data to a low-dimensional representation, followed by a decoder that maps the encoded representation back to the original space. During training, the network learns to reconstruct the input data from the encoded representation, minimizing the difference between the reconstructed and original data (Baldi, 2012). Autoencoders have been found to detect anomalies in controlling food safety (Zuo et al., 2022) and fault detection in gas turbines (Luo & Zhong, 2017). Oliveira et al. (2022) found the method helpful in evaluating anomalies on a heavy-haul railway line.

The Isolation Forest algorithm randomly selects features and splits the data at each node of a decision tree, forming an isolation tree. Anomalies are more likely to be isolated in shorter trees than in normal data points, making them easier to detect. The algorithm computes an anomaly score for each data point based on the average path length in the tree. Isolation Forests are a fast and scalable algorithm that can handle large datasets and is not sensitive to the dimensionality of the data (Liu et al., 2008). These features of Isolation Forest make it a benchmark method for anomaly detection (Demetriou et al., 2022) (Zuo et al., 2022). During the literature review, Isolation Forest has also been found to contribute to condition-based maintenance for overhead transmission lines (Manninen et al., 2022).

One-Class SVM is a Support Vector Machine (SVM) algorithm for anomaly detection. It aims to learn a decision boundary that separates the normal data points from the anomalous data points. Unlike traditional SVM, One-Class SVM is a form of semi-supervised learning because only the normal data is provided during training. The algorithm constructs a hyperplane that encapsulates the normal data points and maximizes the margin between the hyperplane and the nearest data points. During inference, data points that fall outside the boundary of the hyperplane are classified as anomalies (Schölkopf et al., 1999). One-Class SVM has been useful for fault diagnosis as a novelty detection tool in DSL networks (Marnerides et al., 2015). Monroy et al. (2009) established fault detection and diagnosis systems based on One-Class SVM models.

2.2.2 Choice of method

When exploring the problem of classification of information states within a decision-making context, it is assumed that all labels of information can be modeled probabilistically. It is thereby possible to select either a supervised, semi-supervised, or unsupervised learning algorithm.

The following two principles, therefore, guide the choice of an algorithm:

The first principle is that of Box (1979), stating that *"all models are wrong, but some are useful."* Thereby saying that all of the different algorithms will provide some results which are not coherent or in accordance with the physical reality, but they may still be useful in the right context.

The second principle is Occam's razor, suggesting that a simpler model should be preferred over a complex when faced with different models.

With those guiding principles, the Random Forest algorithm is chosen because of its ease of computation and relatively low complexity in conveying and understanding the results The Random Forest will be explored in Chapter 4 of the Theory.

2.2.3 Summary of state-of-the-art review

The state-of-the-art literature review presented different possible classification algorithms with different uses based on the objective. Most of the methods reviewed were evaluated, in the respective literature, through the accuracy of their test data predictions. However, it is also argued that models should be thoroughly empirically validated, so more than a simple measure of accuracy is required.

The different methods have all been deployed in different contexts, and it can therefore be difficult to point at the best method since this is highly dependent on the problem at hand. Often the availability of training data, the type, i.e., dimension and format, are a large part of choosing the most optimal method for the given task (Pimentel et al., 2014).

The objective of this project is to explore how classification can contribute to a better understanding of information states in a system, thus opening different possibilities of either supervised, semi-supervised, or unsupervised learning algorithms.

Finally, the Random Forest method is chosen to explore the problem of how machine learning can be utilized to identify and understand information states which can be utilized in a decision-making scheme.

3 System Identification

Understanding the components of a system, its limits, and how these elements interact with the external environment as well as the future is crucial for decision-making (JCSS, 2008).

The current system identification is inspired by the approach taken by Nielsen (2020) based on Alexander (2002). The problem can be formulated as a design problem allowing for the utilization of the design methods of Alexander (2002). Alexander (2002) puts forth a paradigm for problem-solving used initially in architectural design. The paradigm, however, can be extended to cover many different domains, such as engineering, biology, and philosophy. The emphasis is put on the problem context and the hierarchal structure of design problems, which can be interpreted into any other problem (Nielsen, 2020).

Where Alexander (2002) focuses on the "goodness of fit," the Joint Committee on Structural Safety, JCSS, (2008) describes how decision-makers and engineers should consider the specific context, requirements, and constraints when trying to model decisions, ensuring the models are well-adapted to their intended purpose.

Just as Alexander (2002) advocates breaking down complex design problems into smaller, manageable sub-problems, engineers can divide a large-scale system into smaller components. This facilitates for better understanding, modeling, and simulation of the system and its interdependencies. JCSS (2008) also advocates for such a course of action by hierarchical systems modeling. Here the notion is that when assessing risks in a decision problem, the context of the decision problem should always be considered.

3.1 Matter Form Function

The basic concept of exploring information types through classifications presents a system design challenge. This challenge can be addressed by focusing on the three main elements of Alexander (2002), which are "matter," "form," and "function."

Matter, or material, is referred to as the building blocks for creating any form. However, Alexander recognizes that a design's success depends on how well the material properties are understood and utilized. This includes considering material strength, durability, aesthetics, and sustainability (Alexander, 2002).

The form is the primary focus of Alexander (2002). He contends that a *"harmonious"* form results from a well-adapted design solution that addresses various problems and constraints. He emphasizes the significance of hierarchical structures and patterns in achieving a coherent and meaningful form (Alexander, 2002).

"(...) there are good reasons to believe in the hierarchical subdivision of the world as an objective feature of reality." (Alexander, 2002).

The function is an essential aspect of design. According to Alexander (2002), a good design should serve the needs and requirements of its users and context. Functionality is closely related to the concept of "goodness of fit," where a successful design is highly adapted to the situation's contextual needs and constraints.

The following chapter presents a system identification based on the principles of JCSS (2008) systems modeling and that of Alexander (2002) of matter, form, and function.

3.1.1 Function

The function should serve the needs and requirements of its users and context. The functionality should be adapted to the system's specific needs, constraints, and context. The case study is designed to be a generalized replica of a dynamic system with a utility-based functionality of input to output.

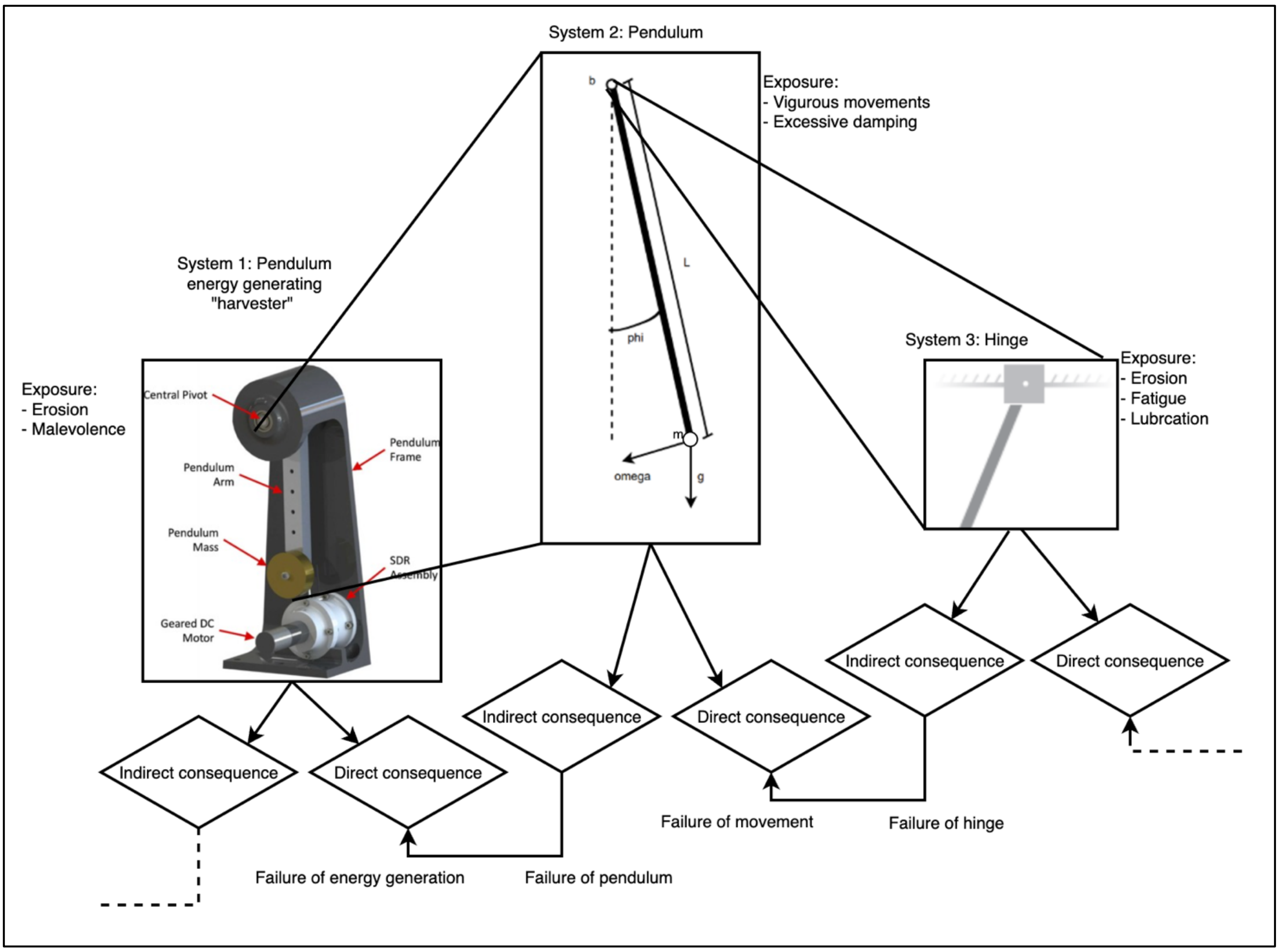


Figure 7. The pendulum system at different scales. Inspired by the JCSS (2008)

The system chosen to fulfill those requirements is a planar pendulum, which functions as a kinetic energy harvester based on input movement. Such systems can be found in water wave energy conversion, energy harvesting from vehicle maneuvering, and human motion (Graves & Zhu, 2022).

The system representation outlined in Figure 7 is only meaningful when considered in the context of constituents, exposures, and consequences. This can be straightforwardly done through a generic approach offered by the JCSS (2008) (a principal visualization of this can be seen in Figure 8). The exposure events can be identified either quantitatively, through databases, sensor data, or qualitatively through expert interviews or hazard identification workshops. This project only considers the very limited exposure event of excessive oscillation, which is modeled by a mathematical model of the physics of the pendulum. This is elaborated further in the "Form".

As stated in Glavind et al. (2020), there are known exposures and unknown exposures. Whereas it is of course easier to model and handle well known exposures. These are depicted as the yellow circles, in the upper part of Figure 8. The known exposures may however interact and cause exposure events, which are previously unthought of, or neglected. For now, the pendulum system only considers the yellow exposures, such as vigorous movements, excessive damping, and external object interfering with the pendulum movement.

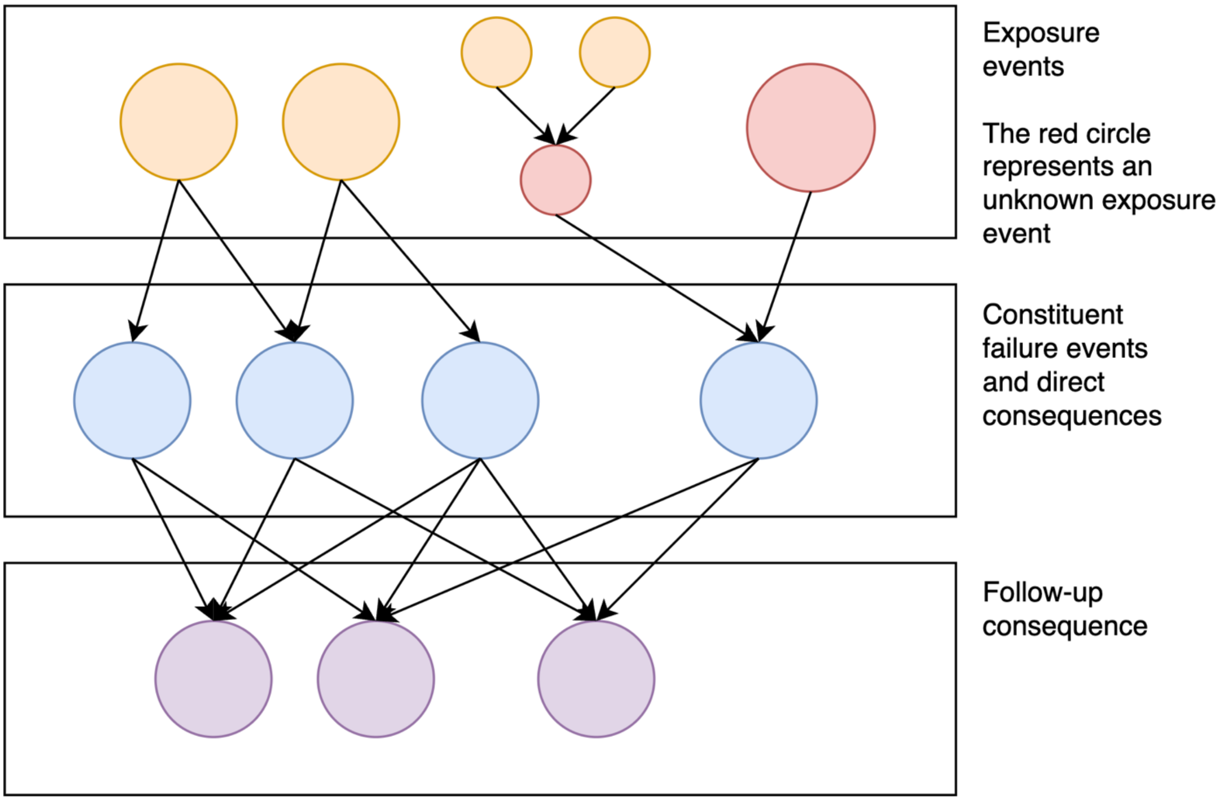


Figure 8. A generic representation of systems in risk assessments (JCSS, 2008). Notice that the red circle in the upper third of the figure represents an unknown exposure event, which is difficult, if not impossible, to appreciate in the risk assessment fully.

Direct consequences refer to the harm caused to the individual components of a system. In contrast, indirect consequences encompass any outcomes beyond these direct effects, such as those linked to the loss of the system's functions. Ultimately, the failure of the hinge, leads to failure of the pendulum. Thereby, a direct consequence to the hinge may lead to an indirect consequence and failure to generate electricity in the generator (JCSS, 2008).

One of the criteria when formulating this system identification is to ensure that the system model can facilitate a ranking of decision alternatives. Furthermore, the ranking should be consistent with the available knowledge and offer the possibility of being continuously updated and revised according to future knowledge. This becomes increasingly important for the current system case as the project results need to be incorporated into the system representation (JCSS, 2008)..

3.1.2 Form

The design process should break down complex problems into smaller sub-problems, as shown in the previous subchapter of the function. An essential part of defining the form of the design problem is the form of the system considered.

The case study system of the pendulum is based on a mathematical model. This means that the observed output data is calculated through a Monte Carlo simulation of the pendulum. This is possible by solving a set of ordinary differential equations, ODEs. There are multiple different methods for solving ODEs; common for most of them is the trade-off of complexity vs. accuracy. A popular choice is the fourth-order Runge-Kutta, RK4, because of the following (Kaw, 2010):

- Higher accuracy, compared to more straightforward methods like Euler's method,
- Ease of implementation. The RK4 method is relatively easy to understand and implement.
- High efficiency. While RK4 requires more function evaluations per step than lower-order methods, its higher accuracy means that fewer steps are needed to achieve the desired level of accuracy.

The RK4 aims to approximate the solution to an ordinary differential equation, represented as:

$$y' = f(x, y) \qquad \textit{Eq. 3.1}$$

To apply RK4, each step is divided into small intervals with a fixed step size h. Throughout each interval, the method calculates the solution at four distinct points combined to generate an updated estimate for the solution at the interval's conclusion. The updated estimate is determined through a weighted average of these four estimates, denoted $k1, k2, k3$, and $k4$, derived by evaluating the given function f at different points within the interval (Kaw, 2010).

Specifically, the method computes four intermediate values of y, denoted $k1, k2, k3$, and $k4$, as follows:

$$k1 = h * f(x, y) \qquad \textit{Eq. 3.2}$$

$$k2 = h * f\left(x + \frac{h}{2}, y + \frac{k1}{2}\right) \qquad \textit{Eq. 3.3}$$

$$k3 = h * f\left(x + \frac{h}{2}, y + \frac{k2}{2}\right) \qquad \textit{Eq. 3.4}$$

$$k4 = h * f(x + h, y + k3) \qquad \textit{Eq. 3.5}$$

The updated estimate of the solution is then given by:

$$y_{new} = y + \left(\frac{1}{6}\right)(k1 + 2*k2 + 2*k3 + k4) \qquad \textit{Eq. 3.6}$$

An example of a pendulum swing is calculated in the Appendix 3 Runge-Kutta methods and Appendix 4, Runge-Kutta single iteration.

When using the Runge-Kutta method to solve the ODE of the pendulum, the following assumptions are made:

- Constant length: The method assumes that the length of the pendulum rod remains constant throughout the swing. This is a reasonable assumption for most physical pendulums.
- Time-invariant system: The method assumes that the system is time-invariant, meaning that the equations of motion do not change over time. This means the damping coefficient and other parameters remain constant throughout the swing.

While these assumptions are not always strictly valid for real-world pendulums, they are often reasonable approximations and can provide reasonable results for the system case (Kaw, 2010).

The form, according to Alexander (2002), should also address the solution to the problem. This will also be done in a mathematical domain at the intersection of engineering and data science to address the challenge of the classification of information states in a decision-making context. During the literature review, the Random Forest algorithm was found to offer a possible solution to the challenge. The Random Forest algorithm will be thoroughly presented in Chapter 4 of the Theory.

3.1.3 Matter

Where matter is referred to as the building blocks for creating any form, Alexander (2002) recognizes that a design's success depends on how well the material properties are understood and utilized. This is interpreted as the characteristics of the pendulum itself, along with the external forces influencing it. This is, however, limited to the influences possible to model under the constraints of the RK4 solution.

For ease of computation, the considered model is an idealized planar pendulum. The idealized planar pendulum is also called a simple pendulum (Pook, 2011). Such pendulum is built on the following assumptions, which make the analysis more tractable:

- The pendulum is planar in a two-dimensional space. Therefore, it is not possible for any three-dimensional oscillations.

- The entire mass of the rod is gathered in an infinitely small point at the bottom of the pendulum rod.
- The hinge is, in principle, frictionless, and the rod does, in principle, not encounter any wind resistance. However, a damping coefficient is added to the model to simulate model these real-world discrepancies.

The pendulum is assumed to have the following attributes, which are used as input to the RK4:

Attribute variable	Abbreviation	Value
Mass of pendulum (in kg)	m	0.1
Acceleration due to gravity (in m/s^2)	g	9.81
Length of pendulum rod (in meters)	L	Gaussian distribution, μ= 0.01, σ=0.001
Initial angle (in radians)	phi	0.79
Initial angular velocity (radian/s)	omega	Gaussian distribution, μ= 0, σ=0.01
Time step (in seconds)	dt	0.1
Duration of simulation (in seconds)	t_max	10
Damping coefficients (kg/s)	b	Gaussian distribution, μ= 0.1, σ=0.02

Table 5. Attributes, abbreviations, and values of the input variables for the RK4

The pendulum's mass, **m**, situated at an infinitely small point on the very bottom of the rod, is assumed to have a mass of 100 grams.

The acceleration due to gravity, **g**, is set at 9.81. This value is the standard approximation used in physics calculations (Atkins & Escudier, 2013).

The initial angle in radians, **phi**, the angle at which the RK4 starts, is set at 0.79 which is 45.263^{o}. This can essentially be set to anything between 0 and 365.

The duration of each timestep, **dt**, is set to 0.1 seconds for a total time, **t_max**, of 10 seconds. Thereby resolving in 1000 timesteps for each iteration of RK4. Meaning that the RK4 will reiterate 1000 times before a timeseries is returned to the array of Monte Carlo simulations.

The damping coefficient, **b**, acts as different external and internal system influences. E.g., external forces acting on the pendulum, such as wind resistance, gravitational pull, vibrations, and other outside disturbances, as well as internal forces, such as bearing friction on the hinge, erosion, etc. As seen in Table 5, the damping coefficient is modeled through a Gaussian distribution, with a mean, μ,= 0.1, and a standard deviation, σ, =0.02. This means that for each simulation in the Monte Carlo simulation, a new value drawn from this distribution is used to solve the ODEs.

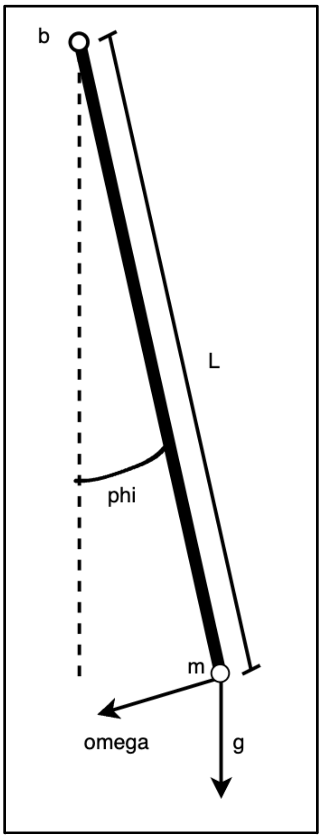


Figure 9. A closer display of the pendulum and the different forces acting upon the system's behavior.

To simulate the random movement of the pendulum, which it will be subject to, either through water waves, vehicle maneuvering, or human motion, the initial velocity, **omega**, also is modeled by a Gaussian distribution, μ= 0, σ=0.01. This value will also change in each simulation iteration.

The length of the pendulum rod, **L**, will initially be set to 10 cm. This will, however, be modified in an experiment later in the project. During the experiment, the length of the rod will be modeled by a Gaussian distribution, μ= 0.1, σ=0.001

3.1.4 Summary of the Function, Form, and Matter

A case system is identified through the interpretations of the problem of information classification as a design problem, which will serve as a generalized example of any system with equivalent properties of dynamics and transformations of input to output. First, in the chapter on system identification, the case system is presented in terms of function, i.e., the input and output functionalities. Then the form of the system is in this case interpreted through the fourth-order Runge-Kutta scheme. Finally, the matter was presented as the system's characteristics through the variables needed to solve the fourth-order Runge-Kutta scheme.

3.2 Information States

When exploring solutions to the problem statement, an important part of the exploration is breaking down the components of the problem, just as Alexander (2002) and the JCSS (2008) prescribe. Therefore, three overlapping taxonomies of information are presented and their similarities presented.

Firstly, two distinct categories of information are proposed by Raiffa & Schlaifer (1961): Perfect and imperfect information.

Perfect information, as the name suggests, pertains to information that accurately mirrors the true state without any degree of uncertainty. This is an idealized condition that is valuable to consider for theoretical discussions and understanding, despite not typically being attainable (Raiffa & Schlaifer, 1961).

In contrast, imperfect information corresponds to scenarios where additional details related to a given situation are not direct or may be mixed with random disruptions or noise. This is often the reality in real-world scenarios such as in conducting experiments or collecting samples, where information gathered may not precisely reflect the true conditions due to the potential interference of various factors or inaccuracies (Raiffa & Schlaifer, 1961).

Feltham (1968, as cited in Zhang et al. 2021) introduces two different types of errors: variability and bias. Both of which are imperfect information. Variability is linked to random disturbances or noise that can affect the accuracy of measurements. It is typically represented using a normal distribution with a mean of zero, illustrating that these disturbances are just as likely to be positive as they are to be negative, with no overall directional trend.

On the other hand, bias is the discrepancy between the anticipated value of measurement and the actual outcome due to possible systematic errors in the measurement process. These systematic errors, are consistent and predictable in their effect, leading to measurements that are systematically off-target. When a measurement can be deemed as an unbiased process, the zero-mean normal distribution can be used to model the randomness of the measurement error (Feltham, 1968, as cited in Zhang et al. 2021).

In their 2019 research, Nielsen et al. discuss a methodology for decision-making. It is argued that the quality and availability of information play a crucial role in shaping the outcomes of the decisions taken. Thus, for optimizing the decision-making process, it becomes crucial to appreciate the types of information at hand. Accordingly, Nielsen et al. (2019) propose a taxonomy for different types of information:

1. The information is relevant and precise.
2. The information is relevant but imprecise.
3. The information is irrelevant.
4. The information is relevant but incorrect.
5. The flow of information is disrupted or delayed.

Even though the first type of information is relevant and precise, it is often found in decision-making subject to uncertainties. This type of information may contain some natural uncertainty but is still precise and relevant for the given decision. Such situations can be found in different domains, including structural engineering, where constructions withstand the calculated loads. (Nielsen et al., 2019).

The second information type considers imprecision. E.g., the information with a high level of uncertainty associated. This may include measurement errors and ambiguity in observations. This can be caused by either aleatory[i] or epistemic[ii] uncertainties in the observations (Nielsen et al., 2019).

The third category contains irrelevant information. When obtaining information, it is crucial to beware of the source of the information. Observed data can originate from different unconsidered systems or system states. This may cause the information to be irrelevant and thereby cause the decision, which is based on the information, to be suboptimal. The solution to this is to account for all system states and alternative systems that may provide the same information. The probability that the information is derived from the considered system, or an unknown system must be represented consistently in a decision analysis (Nielsen et al., 2019).

In the fourth scenario, the information is relevant yet inaccurate. Gross mistakes, such as those made through misconduct in gathering or processing information or by malicious intent, can result in inaccurate information. The use of inaccurate or defective equipment and the incorrect interpretation of data from tables are two examples of massive mistakes. It can also be said that information type 4 is related to epistemic uncertainties (Nielsen et al., 2019).

Information type 5 may arise during the transmission of information. This may also have an impact on the state of the system as well as the generation of consequences. Different possible technical failures can cause this type of information. These interruptions and delays in communication are frequently seen as the direct results of natural disasters, cyber-attacks, or other possible accidents (Nielsen et al., 2019).

Even though there are multiple views on different types and states of information, some common ground can be found between Nielsen et al., (2019), Feltham (1968, as cited in Zhang et al. 2021), and Raiffa & Schlaifer (1961). Generally, it can be said that perfect information may be too idealistic, but relevant to consider, while imperfect information is more important, as it is easier to achieve.

The category of imperfect information can be split into multiple subcategories. The two subcategories suggested by Feltham (1968, as cited in Zhang et al. 2021), can be included in the information types presented by Nielsen et al. (2021). As observations generally tend to contain some noise, information type 1 and information type 2 can be considered as noise (variability error), while information type 4 relates to gross mistakes and systematic errors such as bias.

The current project will therefore include these two types of imperfect information: Noise and bias. Further it is assumed, that any and every observation made will contain noise, the project will

[i] Aleatory uncertainties refer to changing values due to random nature or an inherent variety of physical phenomena. Aleatory uncertainty is irreducible as it appears random (Nielsen, 2007).
[ii] Epistemic uncertainties refer to changes in values that could be known but are not. It is often possible to reduce epistemic uncertainty by acquiring more knowledge about the given phenomenon (Nielsen, 2007).

therefore extend the category of bias to also include noise. In the next subchapter, a description of how noise and bias is implemented in the data.

3.2.1 Presentation of the data

The following subchapter concerns the data which will be analyzed later in this project. The data is derived from the RK4 method described in the Form above. Then, the data is mutated to contain noise for some observations, as well as bias and noise for other observations. This is done to reassemble the different information types described previously in the Information States. To get a better understanding of the data for the case study, the dataset is presented. A flowchart of the data preparation can be found in Figure 12.

To this extent, it is important to appreciate that the pendulum model is used as a prior systems model but is also the same model used when assumed observations of the real physical system is being obtained. It may be helpful to think about the prior model as a digital twin of the physical reality, and the term "observations" is referring to information obtained from the physical reality. However, these "observations" may still contain errors as described in the Information States and it is thereby not possible to be sure that they depict physical reality precisely.

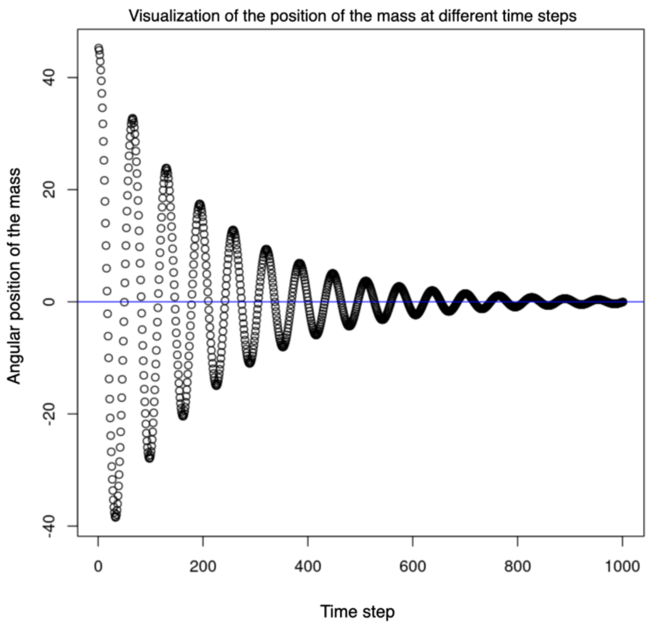


Figure 10. The position of the pendulum mass at different time steps

The data preparation is described through the following five steps:
The first step is to simulate 10 000 Monte Carlo simulations based on the prior probabilistic model of the physics of a pendulum, described in the Matter and the Form. Each timestep from the RK4 is stored as data point which represents the angular position of the pendulum mass at different timesteps i.e., a 10 000 by 1000 data frame. If the rod of the pendulum is exactly vertical, then the mass is considered to be at 0º. The data frame contains values between 46.264 and -43.957, i.e., the

angular position of the mass of the pendulum is maximally 46.264º to one side and maximally -43.957º to the other side.

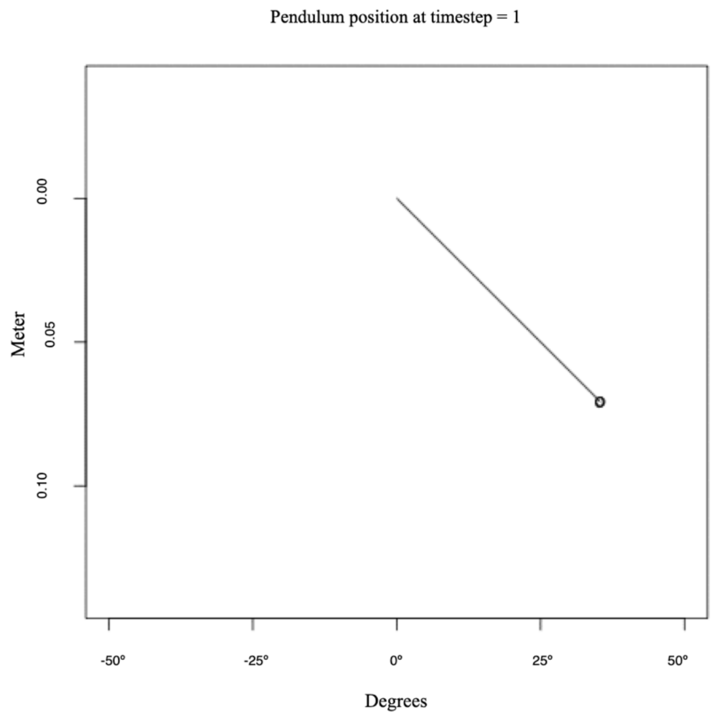


Figure 11. A graphical representation of the pendulum at timestep 1. This should be seen as a "snapshot" of the pendulum, and not a graph. On the x-axis is the angular position of the pendulum mass in relation to hinge. A still vertical hanging pendulum mass is said to be at a position of 0º.

Step 2. The data frame is copied into two identical data frames which can be manipulated individually. This approach provides the flexibility to perform the transformations on each data frame separately. While it still maintains a reference point in the original data, facilitating comparisons of the outcomes derived from each individually manipulated data frame and the tracing back of transformations if required.

Step 3.a. One data frame should be influenced by noise that each individual data point of each feature has been augmented by adding a random sampling drawn from a Gaussian distribution with a zero mean and a standard deviation of 0.1.

Step 3.b. The other data frame consists of features that should simulate to be influenced by bias and noise. For this, each observation of each feature has been added a constant bias of 1, as well as a random draw of a Gaussian distribution with a mean of 0 and a standard deviation of 0.1.

Step 4. The pair of data frames are merged into one data frame, accompanied by an identifier denoting the method of feature manipulation. This results in a single data frame with dimensions of 20,000 by 1,001.

In the final, fifth step, the data frame is randomly partitioned into a test dataset and a training dataset. The division proportions are such that the training dataset contains 70% of the total data, whereas the test dataset contains the remaining 30%. This partitioning strategy ensures an adequate balance for both model training and evaluation of the Random Forest, which will be explained in the following chapter of the Theory.

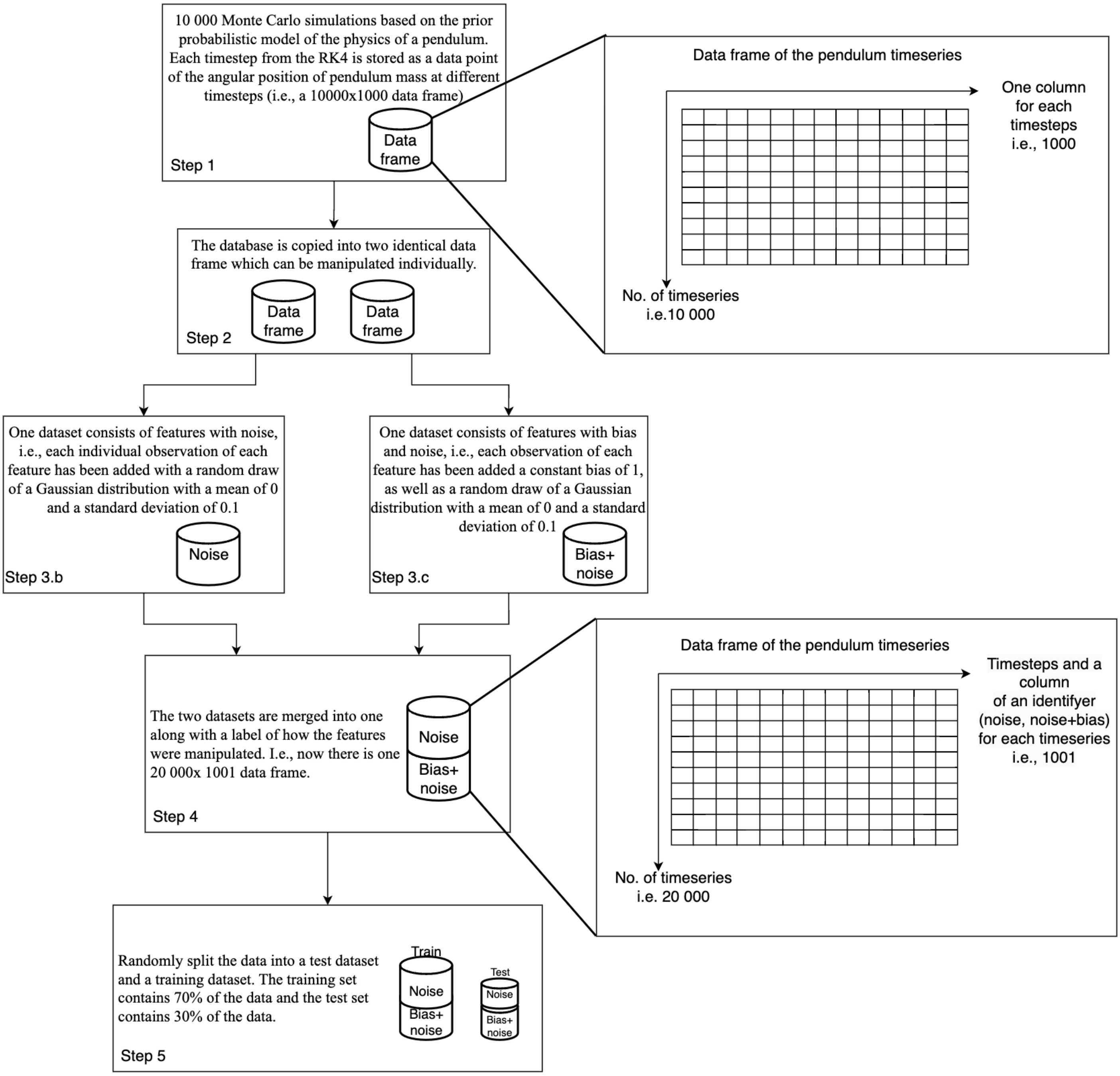


Figure 12. Flowchart of the data preparation

4 Theory

In the following chapter, Bayesian decision theory is presented, elaborating on its fundamental principles. Following is a presentation of the machine learning method of Random Forest. This includes an exploration of its underlying mathematics and practical implications.

4.1 Decision theory

The objective of decision theory is to identify the most optimal decision. This is, however, constrained to various conditions, e.g., preferences and uncertainties. Therefore, it is of interest to develop a systematic approach that consistently treats information to optimally utilize information and rank decision alternatives.

Nielsen et al. (2019) outlines five steps to making decisions subject to uncertainty. Only steps 1, 2, and 3, are considered in this project, while steps 4 and 5 concern calculating the robustness of the decision. Steps 1, 2, and 3 are as follows:

1. Based on available knowledge the decision-maker should identify and represent all possible systems.
2. The decision-maker should formulate and execute a prior decision analysis for the ranking of decision alternatives. This should also account for different possible underlying systems.
3. Based on the prior analysis, the decision-maker should formulate and execute a pre-posterior decision analysis for the ranking of decision alternatives. This should be done so it incorporates the collection of additional information as well as commissions for additional strategies of information collection.

Glavind & Faber (2018) set five different criteria for such an approach and named it a probabilistic system representation. The probabilistic system representation is a model of reality in the context of decision-making. A good model in this context should facilitate a possible ranking of different decision alternatives. Furthermore, it should be able to:

- Represent multiple different, potentially competing systems.
- Include a probabilistic representation of the constituting elements, hereby also the aleatory and epistemic uncertainty.
- The parameters of the underlying models' constituents should be described probabilistically (this also includes the epistemic and aleatory uncertainty).
- Facilitate the inclusion of information derived from system observations and experiments.
- As well as consistently represent any statistical uncertainty brought on by a lack of data, information, or knowledge.

Decisions are often susceptible to incomplete or uncertain information because the available knowledge is uncertain or unknown, e.g., wind loads, material properties, future operational circumstances, and degradation processes.

A system model offers a mapping from input to output in a decision context, conditional on a utility-based choice of option. Since system performance is typically unpredictable, the best option must be chosen in line with Bayesian decision theory (Raiffa & Schlaifer, 1961) and the axioms of utility theory (von Neumann & Morgenstern, 1953) this can be done by optimizing the expected utility to find the optimal action “a*”, that is:

$$a^{*} = \arg\max_{a} E'[U(a)] \qquad \textit{Eq. 4.1}$$

Where E'[U(a)] is the prior expected utility, which may be calculated as:

$$E'(U) = \sum [P(x)U(x)] \qquad \textit{Eq. 4.2}$$

The expected utility represents a weighted average of the utilities of all possible outcomes, where the weights are the probabilities of the different outcomes.

Bayesian decision theory refers to the application of Bayesian principles and methods to decision-making under uncertainty. It incorporates the concepts of prior and posterior probabilities, to guide decision-making in the face of incomplete data or uncertain information (Raiffa & Schlaifer, 1961).

The four axioms of utility theory should also be respected in a probabilistic system representation. The axioms outline the assumptions of stakeholder preferences and decision-making. The axioms are as follows (von Neumann & Morgenstern, 1953):

1. Completeness: The completeness axiom states that a decision-maker can compare and ranking all possible outcomes and alternatives. i.e., given any two alternatives, the decision-maker can always express a preference for one over the other or consider them equally preferable.
2. Transitivity: The transitivity axiom says that if a decision-maker prefers alternative A to alternative B and alternative B to alternative C, then the decision-maker must also prefer alternative A to alternative C. This axiom ensures the logical consistency of preferences.
3. Continuity: The continuity axiom assumes that preferences are continuous, meaning that if a decision-maker prefers alternative A to alternative B, there exists a range of probabilities at which the individual would be indifferent between receiving alternative A or a lottery with some chance of receiving alternative B. This axiom implies that preferences do not abruptly change with small changes in probabilities.
4. Independence: The independence axiom, sometimes referred to as the independence of irrelevant alternatives, states that if a decision-maker prefers alternative A to alternative B, then introducing a third alternative C that is completely independent and unrelated to A and B should not alter the decision-maker’s preference between A and B. In other words, the decision-maker’s preference should be solely based on A and B and not influenced by the presence of an unrelated alternative.

It can often be helpful to visualize decisions in the form of a decision tree. Even though decision theory can solely be formalized in mathematical terms, it can help to understand and not least convey the meaning of the decision.

Figure 13 illustrates a decision event tree that depicts the process of a simple prior decision analysis. The concept of prior and posterior decision analyses differs in the information available to decision-makers during the decision-making process. This simple decision tree does not allow for a modeling of an underlying system, but only allows for a choice of action, “a” to be modeled in terms of expected utility of the state of nature, “x” given the action.

In a prior decision analysis, decision-makers rely on existing information to evaluate the decision alternatives and their associated uncertainties. The analysis is conducted before any new data or evidence becomes available. Decision-makers assess the potential outcomes, assign probabilities and utilities, and use this information to make an initial decision.

The case where extra information has been obtained, the probabilistic decision models may be modified. This corresponds to the posterior decision analysis. The basis for the probabilistic modeling of uncertainty has been strengthened, but otherwise the posterior decision analysis is conceptually equivalent to the prior decision analysis.

During a pre-posterior analysis, decision-makers make use of their existing assumptions, knowledge, and beliefs to assess the potential outcomes and associated probabilities of the experiment or data collection. They consider the implications of different scenarios and make judgments or decisions based on their current understanding. I.e., when an experiment is planned but the outcome is still uncertain, the decision analysis using such "unknown" information is characterized as the pre-posterior analysis.

The decision-maker has a set of options or "decision alternatives" to choose from. Each alternative has potential outcomes that are uncertain, and each outcome is associated with a certain level of utility or value.

To determine the best decision, the decision-maker uses Eq. 4.2 to calculate the expected utility for each alternative. The alternatives are then ranked based on their expected utility, and the alternative with the highest expected utility is considered the optimal decision.

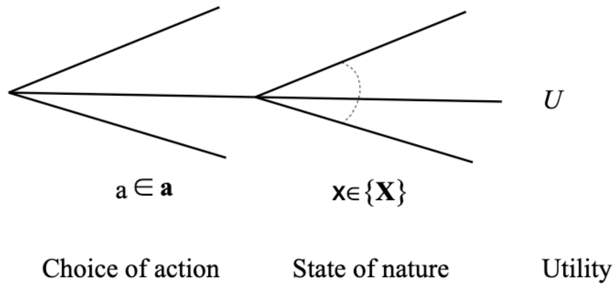


Figure 13. Principal visualization of a simple decision, the dotted line indicates that the outcome of the choice of action associated with uncertain.

However, the true underlying system is often unknown. The fundamental idea is that the properties of the system that is analyzed are unknown, i.e., the true current state of the system for which the decision is concerned is not known. In the case where the underlying system is unknown, the decision tree can be visualized as in Figure 14.

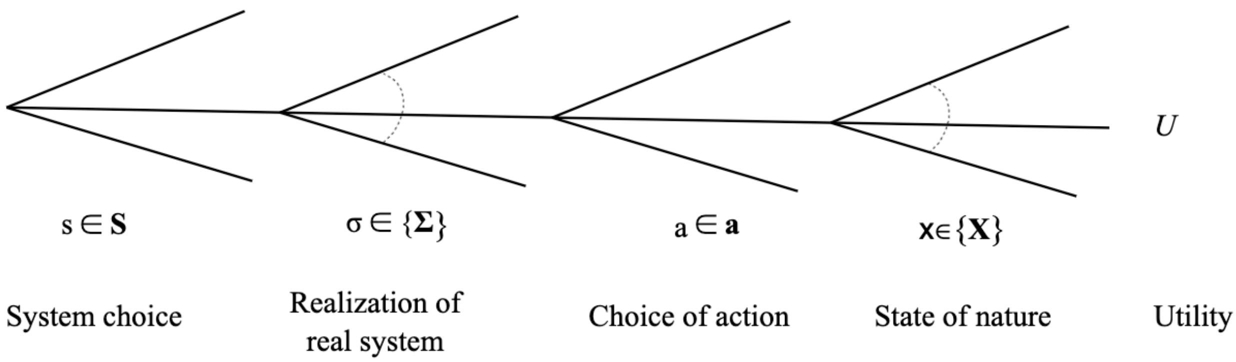


Figure 14. Principal visualization a decision with an unknown underlying system state

A system, “s”, is chosen from a set of potential systems “**S**”. This is what the decision-maker believes to be the true state of the system. The outcome is however associated with uncertainty and an actual system is realized, σ, independently of the choice of system. A decision, “a”, is made independent of the realization of the system. The utility U(a, x) is obtained in accordance with the realization of the state of nature “x”, which is dependent on all prior decisions and occurrences (all of this is represented in Eq. 4.3. It's important to appreciate how the system chosen affects both the action, a, and the result of nature x (Faber, 2011).

Following Figure 13, the realization of the system, “σ”, is related to the probabilistic modeling of “**X**”. This indicates that “**X**” is conditioned on “σ”. This is also why, later in the analysis it can be seen that different states of “x” has different values based on the realization of the system, “σ”. However, the probabilistic modeling of the decision must be based on a selection of a specific system “s”. When the utility is calculated, it is assumed that the chosen system, “s”, coincides with the realization of the real system “σ”, i.e., s = σ. In this case the optimal action “a*” can be identified based on a prior decision analysis from:

$$a^* = \arg\max \left(E'_{\mathbf{X}|\mathbf{a}}[U(\mathrm{a},\mathbf{X})]\right) \qquad \textit{Eq. 4.3}$$

As it can be seen, Eq. 4.1 and Eq. 4.3 closely reassembles each other. The only change is that in Eq. 4.3 the expected utility of a relies of "**X**" which is conditioned on the choice of the system.

It is however not always the case that the decision-maker knows which system is the correct system, i.e., $s \neq \sigma$, and therefore an additional des-utility term must be included to correspond to all other decisions, which may have been correct, but was not chosen. i.e.,

$$E'_{\mathbf{X}|\{\mathbf{\Sigma}\backslash s\}}[U(a^*, \mathbf{X})] \qquad \textit{Eq. 4.4}$$

Because the realization of the true system, "σ", is unknown, the optimal choice of the system, "s*", as well as the optimal choice of action, "a*", will have to be in accordance with the expected utility of all possible system, as shown in the equation from Faber & Maes (2005, as cited in Nielsen et al. 2019):

$$(s^*, a^*) = \arg\max_{s}(P(\mathbf{\Sigma}=s) \arg\max_{a} (E'_{\mathbf{X}|a}[U(a, \mathbf{X})] + E'_{\mathbf{\Sigma}\backslash s}[E'_{\mathbf{X}|\{\mathbf{\Sigma}\backslash s\}}[U(a^*, \mathbf{X})]]) \qquad \textit{Eq. 4.5}$$

Equation 4.5 can be read as follows: The objective is to maximize the expected values of "s" and "a". To achieve this, the variable a needs to be optimized. This involves adding up the probability of a realization of the system "σ" multiplied by the expected value of the best course of action, as well as the expected value of the same action, but considering a different realization of system, $s \neq \sigma$, that did not actually occur. This calculation is performed for all system options, and the one that results in the highest expected value can be selected as the optimal choice.

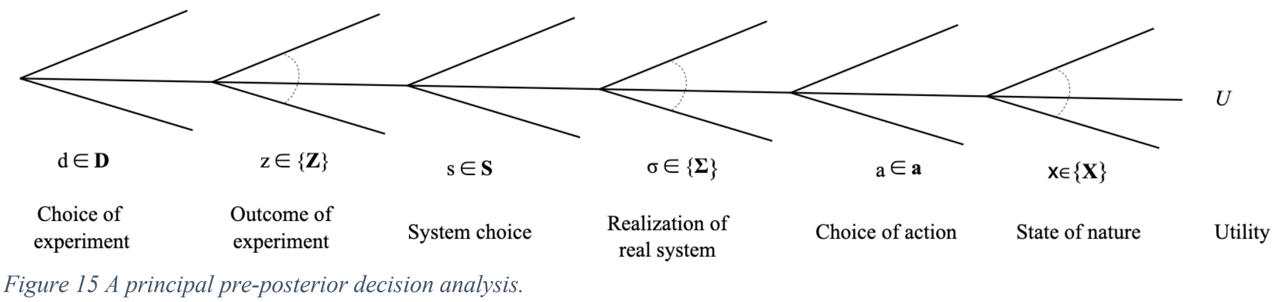


Figure 15 A principal pre-posterior decision analysis.

Before the decision is made, it is possible to obtain new information by conducting an experiment, "d". The outcome of the experiment "z" is associated with uncertainty, as the outcome cannot be known before the choice of conducting the experiment. This is visualized in the principal decision tree of Figure 15.

E´´ is the computation of expected values, which are performed according to updated probability allocations for possible states of "**X**". In other words, P´´(**X**|s) is equal to the previous probability of "**X**" given "s" and but also the outcome of the experiment "z", i.e., P´´(**X**|s, z). Here it may be noted that ´denotes posterior, while ´´denotes pre-posterior.

The optimization which considers the optimal experiment "d", as well as the optimal choice of system "s", and the optimal action "a" may be formulated as:

$$(d^*, s^*, a^*) = \arg\max_{d} E'_z\left[\arg\max_{s}\left(P(\boldsymbol{\Sigma}=s|\mathbf{z})\ \arg\max_{a}\left(E''_{\mathbf{X}|a}[U(a, \mathbf{X})] + E''_{\boldsymbol{\Sigma}\backslash s}[E''_{\mathbf{X}|\{\boldsymbol{\Sigma}\backslash s\}}[U(a^*, \mathbf{X})]]\right)\right)\right]$$

Eq. 4.6

4.1.1 Summary of decision theory

The decision theory provides the boundaries regarding which the decision problem can be analyzed within. Most importantly are the five criteria outlined in the beginning, namely:

- The decision model should be able to facilitate the representation of multiple different systems.
- The model should include a probabilistic representation of the different constituting elements.
- The parameters of the underlying models' constituents should be described probabilistically.
- The model should facilitate for the inclusion of new information derived from system observations and experiments.
- As well as consistently represent any statistical uncertainty brought on by a lack of information.

The criteria above are met if the decision analysis is made in accordance with Raiffa & Schlaifer (1961), and von Neumann & Morgenstern (1953). The theoretic foundation for conducting a prior and pre-posterior analysis presented will be used and exemplified with the decision problem of the pendulum in the analysis.

4.2 Random Forest

Random Forest is a supervised machine learning algorithm that can be used for both classification and regression tasks. However, its primary use is in classification problems. It operates by constructing multiple classification trees during training and outputting a binary response variable. For classification problems this is the mode of the response for all trees. The process of making multiple models and combining them is known as an ensemble model. A Random Forest model is an ensemble of classification trees.

It is important to distinguish a classification tree in machine learning literature from the decision trees of decision theory. The current chapter will only focus on the classification trees from machine learning and will later refer to the ensemble of classification trees as Random Forest.

Supervised learning algorithms is shortly described in the literature review. However, some more context is helpful to understand the specific mathematics and intuition behind the algorithms. It is useful to know, that when building the model, the response labels of the entire dataset is essentially known. To build and evaluate a supervised learning model, the data is randomly split into two datasets: a training dataset and a testing dataset. The model is then built on the training data, where the information regarding the response variable is available. The model is then used to predict the response variable of the test data, where the response is known but hidden from the model. This makes it possible to evaluate the true positives, true negatives, false positives (type 1 error), and false negatives (type 2 error) of the test data (James, et al. 2021).

4.2.1 Classification trees

A classification tree is a fundamental machine learning structure designed to classify an instance into one of several classes. The tree is constructed by repeatedly splitting the instances in the training data into subsets based on the values of the predictor variables.

In Figure 16, each split in the data denotes a test, ($t_1$to t_4), on one of the two features, pendulum position at time step 20 on the x-axis and pendulum position at time step 40 on the y-axis. This partitions the feature space into multiple subspaces, each belonging to either R_1 and R_2 , which may represent either noise or noise and bias.

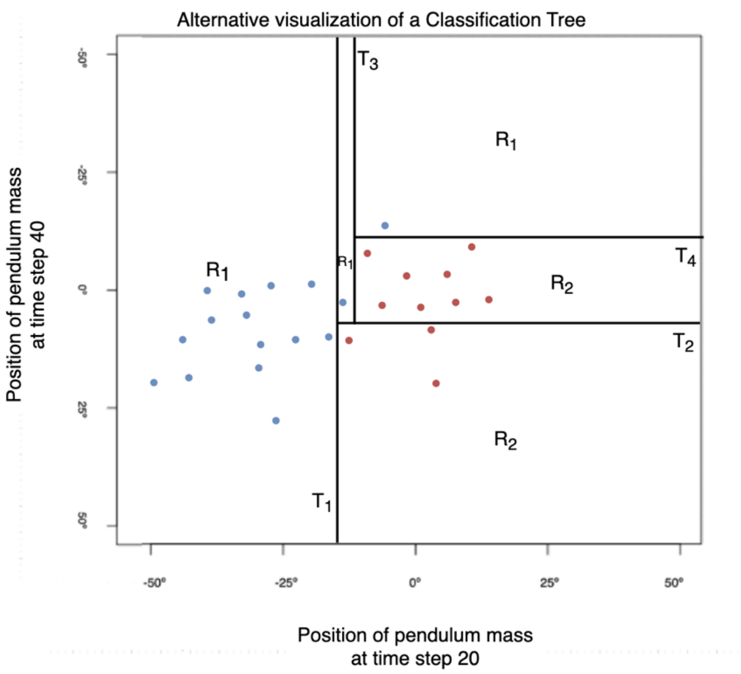


Figure 16. An example of a classification tree. Blue dots may represent observations with "noise" awhile red dots may represent "noise and bias".

Each split, and thereby each test, is based on the Gini impurity, which is a measure used to determine the best attribute to split on. It represents a probability of a random sample being classified incorrectly if a label is randomly picked according to the distribution in a branch. I.e, if there are K=2 classes (in the current case this can be interpreted as either “noise” or “noise and bias”) and at each split of the data the proportion of instances that belong to class i, is p(i), then the Gini impurity of that node is calculated as:

$$\text{Gini Impurity} = \sum_{i=1}^{K} P(i)*(1-P(i)) \qquad \textit{Eq. 4.7}$$

The Gini impurity ranges from 0 to 1 - 1/k, where 0 indicates that all the instances of one side of a split belongs to a single class (i.e., the side of the split is "pure"), and 1 - 1/k represents a pure class distribution (i.e., the maximum impurity).

Each test is made by choosing a split point in the data. For continuous features, the classification tree algorithm generally sorts the values and considers splitting at the value of which lies lies between two adjacent values, i.e., if one feature contains the following values: 1, 9, and 12, then there be a calculation of the Gini Impurity of a split at 6, and 10.5. Whichever of those two splits have the lowest Gini Impurity is going to be the split performed.

In a Random Forest, the default behavior is typically to grow each decision tree to its maximum depth, meaning each tree in the ensemble is grown until all subspaces are pure and have a Gini impurity of 0. It is of course possible to stop the trees from being built too large as the process can be computationally heavy for larger data sets. The rationale for building trees until the Gini Impurity is 0, is that even though a fully grown tree may be overfit to its sample, the process of averaging the predictions of many trees that are each overfitted to different samples can still result in a model with good generalization performance (Hastie et al., 2009).

Once the tree is constructed, it can be used to make predictions of the test data, and finally used on new previously unobserved data.

4.2.2 Building a Forest

Random Forest is an ensemble of the classification trees above. The number of trees in the Random Forest is decided by the user. In general, the more trees the better, but it can be computationally expensive to calculate many trees. Often the result of the classification tends to converge between 100-1500 trees in a forest (Breiman, 2001).

To not just build the same tree 1500 times, a method known as bootstrapping is used. For a given dataset of N observations, a bootstrap sample is generated by randomly selecting N instances from the dataset, with replacement. Because of the "with replacement" part, this means that some instances may be selected multiple times and included in the bootstrap sample, while others might not be selected at all (James et al., 2021)

For each bootstrapped dataset, a classification tree is built. But, unlike the standard classification tree algorithm, which considers all features in the dataset for each split, Random Forest only considers a subset of features picked at random. The idea behind this is that by considering only a subset of features, the trees in the forest become de-correlated. Breiman (2001), the creator of the Random Forest algorithm suggests using a subset of features corresponding to m = sqrt(p), where m is the randomly selected features and p all possible features.

Once all classification trees are trained, they are combined to make predictions. For a classification problem, the mode of all trees is taken as the final prediction. i.e., as a default, predictions of new data is made with the decision criteria, that it should have at least 50% of the votes of all trees.

The number of trees and the number of subsets of features randomly selected may affect bias and the variance of the model. Here, the bias refers to the error due to the model's assumptions about the data. A high-bias model oversimplifies the data, leading to underfitting, and it might not capture important patterns, resulting in a higher error on both the training and test data. The variance refers to the error due to the model's sensitivity to fluctuations in the training data. A high-variance model overcomplicates the data, leading to overfitting, and it captures not only the underlying patterns but also the noise in the training data, resulting in a low error on the training data but a high error on the test data (Hastie et al., 2009).

Random Forests are a way to reduce the variance of a single classification tree. A classification tree, especially one with many splits, can have low bias but high variance as it might overfit the training data. By creating a Random Forest and following the mode as a final prediction for a new observation, the Random Forest algorithm reduces the variance while keeping the bias approximately the same, leading to a more robust model (Hastie et al., 2009).

In Random Forests, each tree is built independently from a bootstrap sample of the data, and only a subset of the features is considered at each split. This diversity among the trees decreases the correlation between them, further reducing the variance of the overall model. However, Random Forests can still suffer from high bias if the individual trees are biased. This can happen, for example, if the trees are not grown deep enough to capture the complexity of the data (Hastie et al., 2009).

Random Forests also offer an easy way of controlling the model in terms of statistical errors. To adjust the type 1 and type 2 error, it is possible to either adjust the decision criteria as well as adding weights to the Gini impurity.

The decision threshold can be based on the costs and benefits of different types of misclassifications. For example, if the cost of a false negative is high, it is possible to lower the decision threshold, so it is more likely to predict the positive class and less likely to miss any positive instances.

The Gini impurity can be manipulated by adding class weights which affect how the individual classification trees are built. When determining where to split the data, the algorithm will favor splits that lead to a better classification of the higher-weighted class.

4.2.3 Summary of Random Forest

The Random Forest algorithm is a robust tool for prediction, using a group of decision trees for its operation. Each of these trees is formed by making binary splits in the data, guided by the optimal Gini impurity. Once these trees are constructed, they are collectively used for prediction. The final prediction is determined by finding the most common output across all the trees.

Finally, it is possible to adjust the Random Forest by changes in the number of trees, number of random features used to build the tree, weights on the Gini impurity as well as changing the decision threshold. The latter two will become important in the case study in the analysis, where it will be possible to determine the expected value of type 1 and type 2 errors in a decision analysis.

5 Analysis

The analysis will combine the pendulum system, information types, decision theory, as well as classification. Starting with a summary of the decision problem. This will be related to the decision theory, where a prior analysis will be made. Then a pre-posterior analysis of the decision problem will be made with the introduction of the Random Forest applied as a classification scheme to predict information types.

As described in the introduction, the pendulum is generally preferred to oscillate. However, the pendulum becomes fragile if it oscillates beyond -42º, i.e., the mass of the pendulum is at an angular position of -42º or less. This is assumed to cause excessive damage to the rest of the pendulum system. The pendulum should therefore not oscillate beyond -42º.

The system identification defines two possible states for the physical pendulum. The first, termed "noise," represents a standard state subject to minor variations. The second, labeled as "noise and bias," depicts an anomalous state, also subject to some variability. Importantly, these states don't just pertain to the physical reality of the pendulum, they also extend to the observed data from the system, which could potentially contain noise, or both noise and bias.

To achieve this, the decision-maker can perform one of two actions:

- The decision-maker can stop the pendulum. This leads to an outcome where the pendulum is stopped and must start again. Thereby wasting time as well as requiring energy to be stopped. The outcome is denoted as “x.1”.
- The decision-makers also have the option to do nothing and let the pendulum oscillate. This action has two associated outcomes, namely:
    - “x.2” is the possibility that the pendulum oscillates beyond the -42º and thereby causes damage to the system.
    - And outcome “x.3”, where the pendulum does not oscillate beyond -42º. This means, that the pendulum generates electricity, and the associated utility is therefore positive.

Thereby the preferred outcomes can be expressed in terms of utility in accordance with the transitivity axiom as (von Neumann & Morgenstern, 1953):

$$u(x.3) > u(x.1) > u(x.2) \qquad \textit{Eq. 5.1}$$

5.1 Explanation of the decision variables

To put the decision problem into the context of the information types, the different variables are explained below along with a visualization of a prior decision tree.

System choice, s ∈ **S,** is the choice of which system is believed to be realized in the physical reality. The decision-maker must make a decision based on their beliefs of the state of the pendulum. It can either be in a normal state with some variety, or it may be biased. Figure 17 depicts the decision as a decision tree, here the system choice, s.1, corresponds to the information being with noise. s.2

corresponds to the information type that is with noise as well as bias.

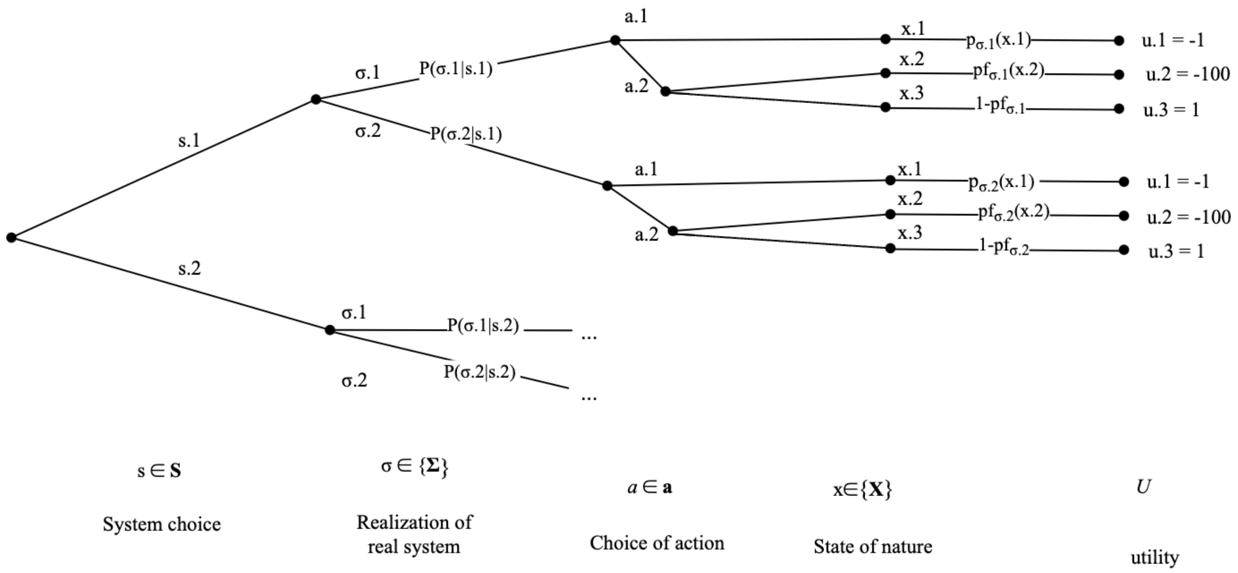


Figure 17. The structure for the decision problem

The realization of systems, σ ∈ {**Σ**}, is the true state of the system. This can be interpreted as the probability that the system is in the state that it is believed to be according to the choice of system.

Following, there are two possible actions, as explained in the decision problem. The choice of action corresponds to either, a.1, which is the possible action to stop the pendulum, and a.2, which is the possible action to do nothing.

If a.1 is chosen, the state of nature x.1 is immediately realized, and the pendulum stops. If a.2 is chosen, then there are two possible states of nature, either x.2 or x.3. Where x.2 corresponds to the pendulum exceeding the threshold and causing damage to the system, and x.3 is corresponding to the pendulum not exceeding the threshold.

The probability of failure for each of the information types is based on calculations from the prior model of the physics of the pendulum. Firstly, by identifying all instances of time series which has at least one instance of a value of less than -42. Then dividing those instances by the total number of simulations. This is done both for the "noise" and for the "noise and bias".

The utility corresponds to the state of the pendulum after an action has been made and the true state of nature is realized. The utility is assumed to be measured on an arbitrary ordinal scale. If the pendulum is stopped by the decision-maker, i.e., through a.1 and thereby x.1, then no electricity is generated, and there is a cost of restarting the system of -1. If the state of nature of x.2 is realized, then the pendulum will cause damage to the electricity generating system and a negative utility of -100 is realized. If the pendulum is realized in a state of nature of x.3, then the pendulum generates electricity and provide a utility of 1.

5.2 Prior

For the prior analysis, the objective is to rank possible decision alternatives, such that the optimal decision can be made of whether to stop the pendulum from oscillating or not. This optimization relies on different information types, which may or may not be true for the real system. The ranking of decision alternatives is therefore dependent on an optimization of the choice of system, which corresponds to the different information types. It can therefore be said that the objective is to optimize and rank the different choices of system states.

Firstly, the expected value of each of the actions and their corresponding outcome is calculated according to Eq. 4.2. Here the different realizations of the system are relevant to consider. For instance, if the realized system only contains noise, then the true state of nature of outcome x.2| σ.2, s.1 = 0.007, while if the same state of nature is also biased, then outcome corresponds to x.2| σ.2, s.2 = 0.055.

The expected value for each of the actions is calculated according to Eq. 4.2. For all possible actions of σ.1, the expected value is calculated as:

$$a.1|\sigma.1 = 1*(-1) = -1 \qquad \textit{Ea. 5.2}$$

$$a.2|\sigma.1 = 0.007*(-100)+0.993*(1) = 0.643 \qquad \textit{Eq. 5.3}$$

Then all possible actions relating to the system realization of σ.2 is calculated in the same manner:

$$a.1|\sigma.2 = 1*(-1) = -1$$

$$a.2|\sigma.2 = 0.055*(-100)+0.945*(1) = -1.805$$

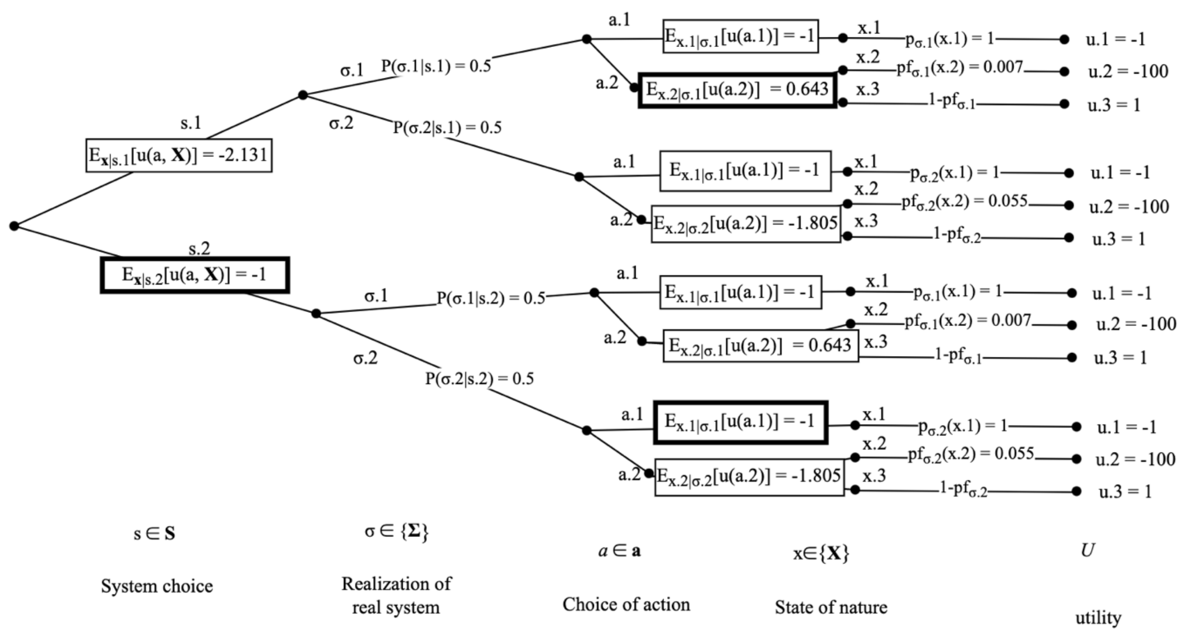


Figure 18 An extended visualization with probabilities, for σ and x, and expected values for s and a.

As described in the Chapter of the Theory, the expected value of the chosen action should be evaluated in relation to the system choice. Thereby the associated des-utility should be calculated as the utility of the same action but different system realization. I.e., the des-utility for action a.2 in σ.1, is the same as the utility for a.2 in σ.2.

For the prior analysis, no information is available regarding whether the true state is "noise" or "noise and bias". Therefore, a non-informative prior of 0.5 is provided for both possible realizations.

In the prior analysis for s.1, σ.1, the highest expected value of the two possible actions is a.2 (marked with a bold line in Figure 17, and 18). The product of the expected value for σ.1, a.2 is then added with the product of the des-utility of σ.2, a.2. This can then be said to be the expected value of s.1.

$$0.5*(0.643) + 0.5*(-1.085) = -2.131 \qquad \textit{Eq. 5.4}$$

The same is then done for s.2, where the optimal action is to stop the pendulum. Again, the des-utility is realized as the same action, but for the alternative system realization. This is then calculated to be:

$$0.5*(-1) + 0.5*(-1) = -1 \qquad \textit{Eq. 5.5}$$

In accordance with Eq. 5.6 the expected values can then be arranged. First by obtaining the expected value of all possible actions and then all possible system choice and the highest value may be chosen as the most optimal choice.

$$(s^*,a^*) = \arg\max_{s}(P(\mathbf{\Sigma}=s)\arg\max_{a}(E'_{\mathbf{X}|a}[U(a,\mathbf{X})]+E'_{\mathbf{\Sigma}\setminus s}[E'_{\mathbf{X}|\{\mathbf{\Sigma}\setminus s\}}[U(a^*,\mathbf{X})]])) = 0.5*(-1) + 0.5*(-1) = -1$$

Eq. 5.6

The prior model is established to show what the decision-making process looks like without any form of classification scheme, but only following observations directly of what is believed to be the system. The two different actions of whether to stop the pendulum or not is evaluated and ranking of the different alternatives can be made, i.e., the optimal choice of system is s.2, which leads to a choice of action of a.1 resulting in stopping the pendulum.

5.3 Pre-posterior

The pre-posterior model expands the prior with the classification scheme. The classification acts as an indicator for the different information types. It thereby contributes with probabilities of observing the specified information type conditioned on the observation. This is essentially an update of the probabilities of the system realizations.

The pre-posterior decision analysis for the optimization of system choice, is the optimization of choice of the experiment. To do this, an "experiment plan" must be formulated, from which it is possible to obtain new unknown information as well as what to do with this new information. The question to be answered in the pre-posterior analysis is: Should an observation be made, and at what price? This is answered by the difference in the expected value of making an experiment to the expected value of not making an experiment.

The choice of the experiment, i.e., when, what, and where to observe the system, is denoted by “d”. Notice that it is always an option to do nothing and just keep to the prior analysis. The outcome of the experiment is associated with uncertainty, denoted by “z”. For the given case, it is assumed that the decision-maker can choose to observe the system at timestep 20.

5.3.1 Random Forest

The classification happens in accordance with the Random Forest described in Chapter 4 of the Theory. The different experiment designs can be described as different strategies of classification, e.g., different features as predictors, weights for the Gini impurity, as well as adjustments of the decision threshold. It should also be noted that the Random Forest model is based on the probabilistic model of the pendulum, i.e., not observations from the physical reality.

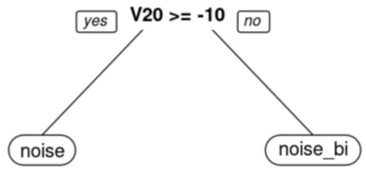


Figure 19. A single classification tree with a single split in of the training data. The test is whether a training data point has a value of more than -10. If the test is true, then it will be labeled as noise is the test is false, then it will be labeled as noise and bias.

To not over-fit the model, a single predictor is chosen, this should be the same feature as the decision-maker would observe. E.g., if the decision-maker can observe the angular position of the mass of the pendulum at timestep 20, then it should be the same feature used for building the Random Forest model.

The algorithm starts with the decision-maker choosing which values to use for the number of trees to build, the number of features to consider at each split, and possibly also the decision threshold as well as weights for the Gini Impurity. For the current example, only 100 trees are built, and only one feature is considered. The decision threshold is set at 50% and there are no added weights for the Gini Impurity.

A bootstrap sample is generated based on the training data set. The new bootstrapped sample consists of 14 000 random data points, sampled with replacement, from timestep 20. Timestep 20 is denoted V20 in the data (Which can be seen in Appenix 4 Data). Then the Gini impurity is calculated based on the best possible split, i.e., after calculating each possible split for the training data, the algorithm returns the first split, i.e., the first test, to be whether the value of the data point is ≥ 10.

The Gini impurity of the split in Figure 19 can be calculated by following eq 4.7. Keep in mind, that this is just one of many splits. First, the probability of instances belonging to class one of the two classes is calculated by first identifying how many instances of data points satisfy the statement of V20≥ -10. From the data, it can be identified that 7896 data points satisfy the statement while 6104 do not. The sample probability is then calculated by dividing each node by the total number of observations in the training set.

$$\frac{7896}{14000} = 0.564 \qquad \textit{Eq. 5.7}$$

$$\frac{6104}{14000} = 0.436 \qquad \textit{Eq. 5.8}$$

These numbers are then used in the Gini impurity formula:

$$0.564 * (1 - 0.564) + 0.436 * (1 - 0.436) = 0.246 \qquad \textit{Eq. 5.9}$$

The Random Forest overfits each classification tree by splitting the tree multiple times until the Gini Impurity is 0. Meaning that the above calculation is done reiteratively many times. Then after the tree is built, a new tree is built, and so on until 100 trees have been built.

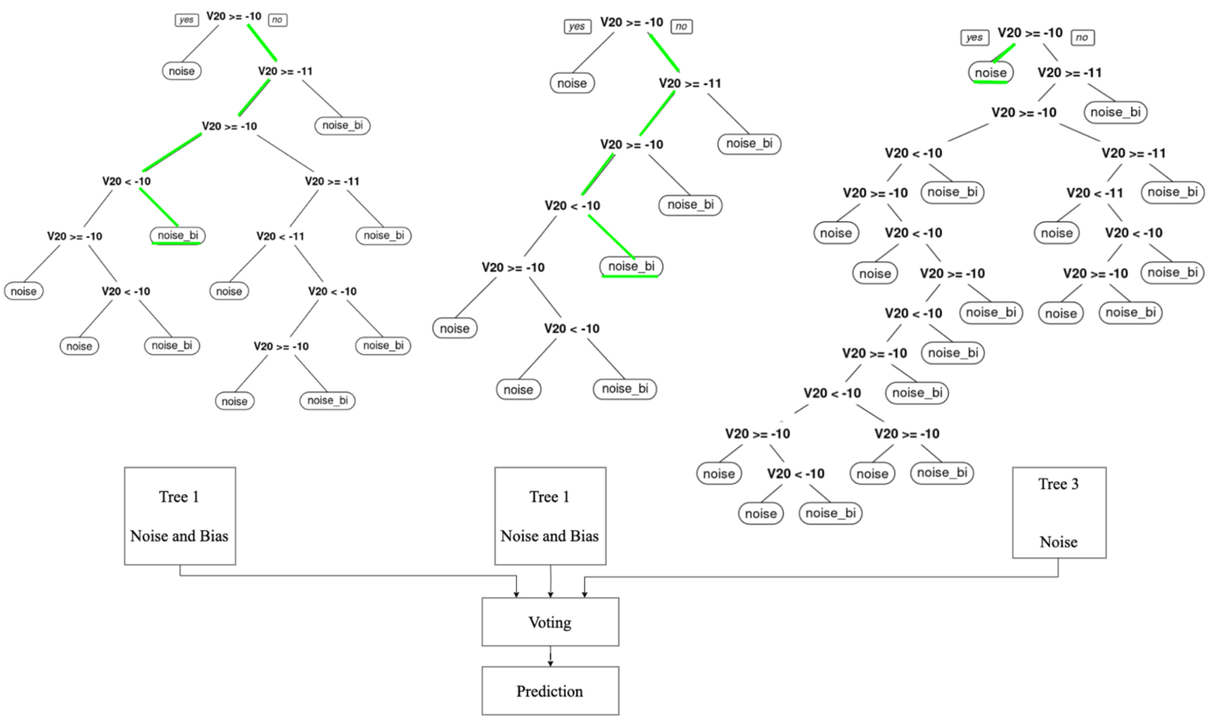


Figure 20. An example of a Random Forest with only 3 trees

The Random Forest is then tested on the test dataset, where each data point is attempted to be predicted. As explained in the theory, the prediction happens by a data point being introduced to the sub-divided space and the is classified according to the label for the given subspace. Then all the different trees "vote", i.e. the mode of the prediction of all trees is the final prediction.

In Figure 20, an example of a Random Forest with only 3 trees is visualized, to grasp how different trees may look in the same Random Forest, even with only one feature as the predictor. Notice how even though the split is almost always at the value 10, different requirements in terms of the mathematical operator (, >, <, ≥, ≤, =) are used, to make different tests. For the classification of information types 100 trees is grown.

When the test set have been predicted, the prediction can be compared to the true labels in a confusion matrix. The confusion matrix is then used to calculate the probabilities of correct and incorrect classifications. The probability matrix can be seen in Table 6.

	True state of information		
Predicted state of information		*Noise*	*Noise and Bias*
	Noise	*0.7598269*	*0.2401731*
	Noise and bias	*0.2624729*	*0.7375271*

Table 6 Probability matrix of the classified information states.

These probabilities may now be used as input to the pre-posterior decision analysis as probabilities for σ.1 given s.1, σ.2 given s.1, σ.1 given s.2, and σ.2 given s.2.

5.3.2 Pre-posterior decision analysis

The pre-posterior analysis builds upon the prior analysis, by updating the probabilities associated with the system realizations. The utility and the probabilities of the different states of nature have not changed, as well as the expected values of the different actions have not changed.

However, instead of the non-informative prior, it is possible to update the probabilities of the system realizations, σ. This influences the expected value of the system choice, which again influences the ranking of the different experiments, d. For the sake of illustration, the experiment d.1 is only associated with one possible outcome and is thereby set to realize with 100% certainty. The mapping of the pre-posterior decision analysis can be seen in Figure 21.

To calculate the expected utility of the pre-posterior system choice, the product for the new probability of σ.1 and the expected value of a.2 is added with the des-utility. Again, the des-utility is the utility for the same action, but different system realization and can be seen in the second term of Eq. 4.5. This can be calculated to be:

$$E''[U(s.1)] = 0.76*(0.643) + 0.24*(-1.805) = 0.055 \qquad \textit{Eq. 5.9}$$

and,

Eq. 5.10

$$E''[U(s.2)] = 0.262*(-1)+0.738*(-1) = -1$$

As it can be seen, the highest expected utility has now shifted from s.2 to s.1. Thereby suggesting the decision-maker to choose system 1 if no other information is available. It is important to remember, that no observation of the system has yet been made. If an observation is made, then the information state should be predicted by the Random Forest model and the choice of system should be made accordingly. However, in terms of expected utility, it can be said, that the decision-maker should choose s.1.

Because the experiment outcome is certain, the maximum expected value is the same as the maximum value for all system choices. Thereby, when considering the possibility to do nothing, the optimal action can be expressed following Eq. 5.11 as:

$$(d^*, s^*, a^*) = \arg\max_{d} E'_{z}[\arg\max_{s}(P(\boldsymbol{\Sigma}=s|\mathbf{z}) \arg\max_{a} (E''_{\mathbf{X}|a}[U(a, \mathbf{X})] + E''_{\boldsymbol{\Sigma}\backslash s}[E''_{\mathbf{X}|\{\boldsymbol{\Sigma}\backslash s\}}[U(a^*, \mathbf{X})]])] = 1*(0.76*(0.643) + 0.24*(-1.805)) = 0.055$$

Eq. 5.11

The pre-posterior analysis of the pendulum plays a crucial role in this decision-making problem. It demonstrates that, given the availability of a relevant classification scheme, it is possible to update the probabilities of the information state. Consequently, this allows the decision-maker to choose the system with a higher expected utility.

The expected value of information is usually calculated by subtracting the expected value before obtaining the information from the expected value after obtaining it. However, in this project it is not the information gained through the new observation, which is of interest, but rather the information gained from classification of that information. It is hereby possible to calculate the expected value of the classification. This can be done by subtracting the difference in the posterior expected value of s. 1 from the posterior expected value of s.2. i.e.,

$$E''[U(s.1)] - E''[U(s.2)] = 0.055-(-1) = 1.055. \qquad \textit{Eq. 5.12}$$

This value represents the gain in expected value when the classification is implemented in the decision-making.

5.3.3 Decision-making based on statistical errors.

Including a classification scheme to classify the different information types in a decision-making process, can also facilitate the decision-makers preferences regarding type 1[iii] and type 2[iv] errors. It may be, that the decision-maker would like to be more cautious and would therefore try to minimize

[iii] Type 1 error is known as a false positive. It occurs when the model incorrectly predicts a positive outcome when the actual outcome is negative. i.e., the model wrongly classifies a negative instance as positive.

[iv] Type 2 error is also called as a false negative, occurs when the model incorrectly predicts a negative outcome when the actual outcome is positive. I.e., the model wrongly classifies a positive instance as negative.

the type 2 error. This can be interpreted as the decision-maker preferring type 1 errors rather than type 2 errors.

In such instances, the parameters of the Random Forest can be changed. As described in the theory, the statistical errors can be changed by tuning the decision threshold for the voting process and by applying weights to the Gini Impurity.

Considering the decision problem of the pendulum. The probability for a type 2, is at error found to be at 0.262 which is slightly higher than the probability of a type 1 error at 0.24. The decision-maker may change their risk appetite to reflect more cautiousness and would therefore like the model to reflect their risk appetite. It is assumed, that the decision-maker would like a lower type 2 error.

By changing the decision threshold from being what the majority vote decides, it can now discriminate between the “noise” class and the “noise and bias” class. To exemplify, an arbitrary threshold has been decided such that for a given observation to be classified as “noise”, it needs at least 40% of the votes from the Random Forest, and to be classified as the “noise and bias”, it needs at least 60% of the votes. If neither of these conditions is met, the observation will be classified as the class that gets the most votes. By following this approach, the following probability matrix of Table 7, is obtained:

	True state of information		
Predicted state of information		*Noise*	*Noise and Bias*
	Noise	*0.6756253*	*0.3243747*
	Noise and bias	*0.1851026*	*0.8148974*

Table 7. Probability matrix of the classified information states, with a preference for low type 2 error.

It is obvious to see, that the shift in type 2 error comes at a cost. Now the true positive, which can be interpreted as choosing system s.1 and the realization of the system, is σ.1, is down to a probability of 0.676. This of course also means that the type 1 error is much higher, the probability of system 2 realizing if system 1 is chosen is 0.324.

Based on these preferences it is then possible to calculate the pre-posterior expected values for “noise” and “noise and bias” respectively:

$$E''[U(s.1)] = 0.676*(0.643) + 0.324*(-1.805) = - 0.15 \qquad \textit{Eq. 5.13}$$

$$E''[U(s.2)] = 0.185*(-1) + 0.815*(-1) = -1 \qquad \textit{Eq. 5.14}$$

It can be seen that the expected value of choosing system s.1 decreases. The expected value of choosing system s.1, is however still more than the expected value of choosing system s.2. Now, the value of this classification is found to be 0.85.

Even though this value is lower, it is more aligned with the preferences of the decision-maker. The probability of wrongfully classifying information as "noise" when it is actually "noise and bias" is much lower. The value of this is difficult to show in the current example because there is only one outcome associated with a.1. Because of this, the expected value of s.2 is still -1.

5.4 Summary of the analysis

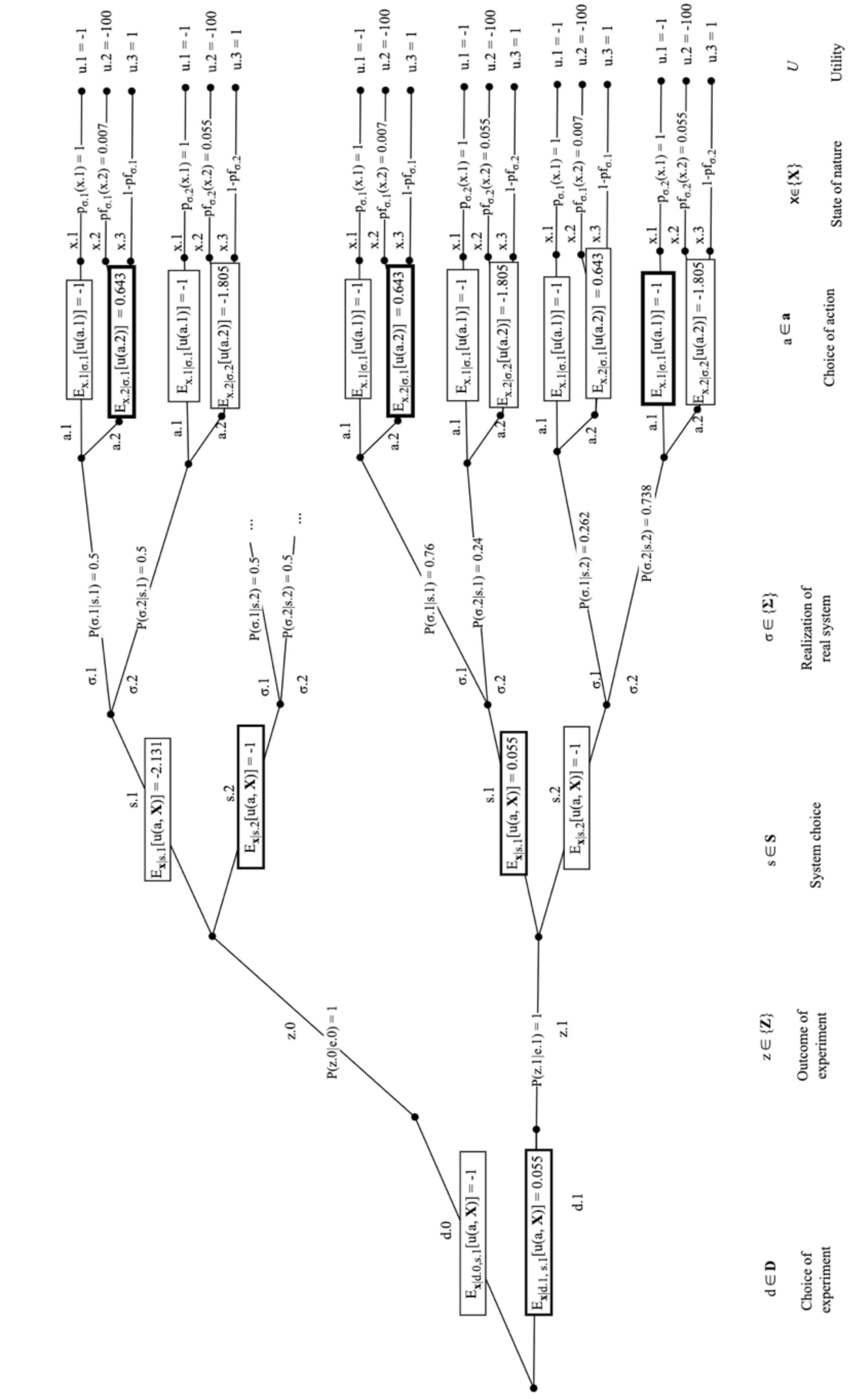
The analysis combines the pendulum with the different information states with the decision theory, and the classification. A prior analysis is made with the result of ranking s.2. higher than s.1., i.e., based on the prior analysis, the decision-maker is recommended to believe the information state as being "noise", when only considering the expected value of the system state.

Then, for the pre-posterior decision analysis, the classification scheme of Random Forest is introduced. The Random Forest classifies by randomly building multiple decision tree based on parameters specified by the decision-maker. The outcome of the Random Forest is a probability matrix, which is used as updated probabilities for the realization of the system states.

The pre-posterior analysis is then conducted with the updated probabilities and new expected values for the different system choices are found. The expected values for the choice of system suggests that it is reasonable to change the choice of system from s.2 to s.1. This change allows for a new term to be coined called the "value of classification". This shows the difference in value of correctly making a classification to an incorrect classification.

Finally, the concept of employing a classification scheme in the decision analysis is used to model the decision-maker preferences of statistical errors.

It is important to remember, that it is the information types that is being classified and not the state of the pendulum in the physical reality, but rather the state of the information which may be obtained from the pendulum.



d ∈ D
Choice of experiment
z ∈ {Z}
Outcome of experiment
s ∈ S
System choice
σ ∈ {Σ}
Realization of real system
a ∈ a
Choice of action
x∈{X}
State of nature
U
Utility
d.0
d.1
z.0
z.1
P(z.0|e.0) = 1
P(z.1|e.1) = 1
s.1
s.2
σ.1
σ.2
a.1
a.2
x.1
x.2
x.3
P(σ.1|s.1) = 0.5
P(σ.2|s.1) = 0.5
P(σ.1|s.2) = 0.5
P(σ.2|s.2) = 0.5
P(σ.1|s.1) = 0.76
P(σ.2|s.1) = 0.24
P(σ.1|s.2) = 0.262
P(σ.2|s.2) = 0.738
u.1 = -1
u.2 = -100
u.3 = 1

6 Discussion

The discussion will consider to main subjects, information states, and limitations. During the analysis, it has become clear that the representation of information is essential for the probabilistic modeling of decisions. To this extent, more information types can straightforwardly be introduced in the classification scheme presented and, of course, also in the modeling of the decision.

6.1 Perspective on information states

The current project only considers two different types of information. However, as presented in the system identification, different types exist. This opens not only the discussion of information types but also about what information is in general. Thereby entering the domain of Information theory.

Before presenting different information types, Nielsen et al. (2021) ask, "What is information?" This is followed by a discussion of truth and information supported by different philosophical opinions. Nielsen et al. (2019) point out that sometimes information is referred to as evidence. This description contributes to the understanding of the effects of information. I.e., knowledge can be understood as information that is "collected and processed over time." Thereby, "information" can be interpreted as evidence of some "difference." Nielsen et al. (2019) ultimately accept Bateson's (1972, as cited in Nielsen et al., 2019) definition of information, stating that "The elementary unit of information – is a difference which makes a difference." Simplified, it can be said that information is a "difference." This notion of information facilitates the possibility of Bayesian methods and information ranking. This also means the information types can be integrated into a probabilistic system representation.

Another, but not the contrary, view can be found in information theory, data science, and structural engineering. Within these domains, a model known as Data-Information-Knowledge-Wisdom has been proposed to present the relationship between different categories of "information" (Zhang et al., 2021). The four different information types should be understood hierarchically, where "data" is the lowest in the hierarchy and "wisdom" is at the top.

Data can be seen as raw, unprocessed measurements without any added interpretation or analysis. It is the most basic level in the hierarchy and is the building block for the other levels. Information comes from processed data. I.e., data that has been given meaning through relational or contextual connection.

If the difference in information can be quantified, then this may be used as a weight in the classification model, thereby allowing for a hierarchical difference of information. For instance, wisdom might be given more weight than knowledge, knowledge more than information, and information more than raw data, reflecting their respective values and relevance in decision-making.

However, it is important to note that not all knowledge can be quantified or easily classified within this model, and the borders between each level of the DIKW hierarchy are often blurred and not strictly defined. The challenge lies in adequately capturing and modeling these nuances within a decision-making context.

Ultimately, this highlights the complexity of modeling decisions, where a variety of factors come into play, including the types of information considered, the way this information is classified and valued, and the decision-making processes employed.

6.2 Limitations

One caveat of the project is that the pendulum is not constantly in motion. This has a great influence on the classification model because the data used in the modeling process is very homogenous. Because of this, the Random Forest was eminent at predicting the information states. In the physical reality, it may not be the case, that a classification scheme will work that well with only one predictor. However, this may be resolved by including more features as predictors and tuning the parameters to closely fit the preferences of the decision-maker. This still does not render the usefulness of the concept of integrating a classification scheme into a decision analysis.

This ties into another criticism which is that, when obtaining new information, it is essential that it is directly from the time step for which the classification model is built. Otherwise, the classification will not yield very good results. This can be tested by classifying observations of different timesteps with the current model.

7 Conclusion

This project identifies how a classification algorithm can identify different information states in a probabilistic system representation of a decision. This is done by integrating a classification scheme into the decision analysis process offered by Bayesian decision theory.

To do this, a theoretical decision problem is set up with an oscillating pendulum that is not allowed to exceed a specific threshold. The objective of the decision-maker is to maximize the expected utility by optimizing the choice of system. Therefore, the decision problem should be seen as a proxy for other systems with the basic properties of mapping an input to an output.

The literature review, consisting of a bibliometric analysis and a state-of-the-art review, reviewed how classification methods and uncertainty modeling have previously been used in structural engineering and decision-making.

A relevant classification scheme is identified in the literature review. It is found that a suitable algorithm is that of a Random Forest. The algorithm builds multiple different decision trees. When new data is presented to the Random Forest, it is labeled by the majority prediction of all the trees.

The pendulum system, as well as different taxonomies of information types, is then presented. This also provides the possibility of modeling the data physics of the pendulum and applying error and variety, i.e., noise and bias. Then the Bayesian decision theory is presented, followed by a description of the Random Forest algorithm, with this also how to tune the parameters to facilitate the decision-makers preferences.

All the above is then put together in the analysis, where the decision problem is solved by integrating the Random Forest into a Bayesian decision analysis. The first step is by making a prior decision analysis, showing the expected value of without any classification or additional information. This is followed by a pre-posterior analysis, where the Random Forest is integrated, and the new expected value is obtained. The Random Forest contributes to modeling the uncertain information regarding the system information state. This contributes to solving the challenge of defining information states out of a set of competing information states when only limited information is available.

The project shows that it is possible to integrate a classification scheme into a probabilistic system representation. By classifying information types, it is possible to inspect the effect of the different information types on the system's performance. As a principal example, the classification of information types of the pendulum showed the value of classification. The concept of integrating a classification scheme into a Bayesian decision can be extended to multinomial classification for including additional information types. This may be utilized in any decision scenario, which can be modeled according to Glavind and Faber (2018), where the information states are unknown. The classification of information state shows that classifying information in a decision context can yield a higher expected value for the system choice and not least the choice of experiment.

8 List of Literature

Accorsi, R., Manzini, R., Pascarella, P., Patella, M., & Sassi, S. (2017). Data Mining and Machine Learning for Condition-based Maintenance. *Procedia Manufacturing*, *11*, 1153–1161. https://doi.org/10.1016/j.promfg.2017.07.239

Alexander, C. (2002). *Notes on the synthesis of form* (17. printing). Harvard Univ. Press.

Arul, M., & Kareem, A. (2022). Data anomaly detection for structural health monitoring of bridges using shapelet transform. *SMART STRUCTURES AND SYSTEMS*, *29*, 93–103. https://doi.org/10.12989/sss.2022.29.1.093

Atkins, A. G., & Escudier, M. P. (2013). *A dictionary of mechanical engineering* (1st ed). Oxford University Press.

Baldi, P. (2012). Autoencoders, Unsupervised Learning, and Deep Architectures. *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, 37–49. https://proceedings.mlr.press/v27/baldi12a.html

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Crupi, V., Guglielmino, E., & Milazzo, G. (2004). Neural-Network-Based System for Novel Fault Detection in Rotating Machinery. *Journal of Vibration and Control*, *10*(8), 1137–1150. https://doi.org/10.1177/1077546304043543

Dang, V.-H., Vu, T.-C., Nguyen, B.-D., Nguyen, Q.-H., & Nguyen, T.-D. (2022). Structural damage detection framework based on graph convolutional network directly using vibration data. *Structures*, *38*, 40–51. https://doi.org/10.1016/j.istruc.2022.01.066

Demetriou, P., Becker, I., & Hailes, S. (2022). Who Watches the Watchers: A Multi-Task Benchmark for Anomaly Detection: *Proceedings of the 8th International Conference on Information Systems Security and Privacy*, 579–586. https://doi.org/10.5220/0010915000003120

Faber, M. H. (2011). On the governance of global and catastrophic risks. *International Journal of Risk Assessment and Management*, *15*(5/6), 400. https://doi.org/10.1504/IJRAM.2011.043698

Glavind, S. T. (2021). *Information-consistent systems modeling and analysis: With applications in offshore engineering*. Aalborg Universitetsforlag. https://doi.org/10.54337/aau443238866

Glavind, S. T., Sepulveda, J. G., & Faber, M. H. (2022). On a simple scheme for systems modeling and identification using big data techniques. *Reliability Engineering & System Safety*, *220*, 108219. https://doi.org/10.1016/j.ress.2021.108219

Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed). Springer.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning: With Applications in R*. Springer US. https://doi.org/10.1007/978-1-0716-1418-1

Joint Committee of Structural Safety (JCSS), F. & MH. (2008). *Risk assessment in engineering*. Joint Committee of Structural Safety.

Kaw, A. (n.d.). *Runge-Kutta 4th Order Method for Ordinary Differential Equations*.

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation Forest. *2008 Eighth IEEE International Conference on Data Mining*, 413–422. https://doi.org/10.1109/ICDM.2008.17

Manninen, H., Kilter, J., & Landsberg, M. (2022). A holistic risk-based maintenance methodology for transmission overhead lines using tower specific health indices and value of loss load. *International Journal of Electrical Power & Energy Systems*, *137*, 107767. https://doi.org/10.1016/j.ijepes.2021.107767

Marnerides, A. K., Malinowski, S., Morla, R., & Kim, H. S. (2015). Fault diagnosis in DSL networks using support vector machines. *Computer Communications*, *62*, 72–84. https://doi.org/10.1016/j.comcom.2015.01.006

Monroy, I., Escudero, G., & Graells, M. (2009). Anomaly detection in batch chemical processes. *Computer Aided Chemical Engineering*, *26*, 255–260. https://doi.org/10.1016/S1570-7946(09)70043-4

Neumann, J. V., & Morgenstern, O. (1953). *Theory of Games and Economic Behavior*. Princeton University Press.

Nielsen, L. (n.d.). *Toward a unified theory of risk, resilience and sustainability science with applications for education and governance.pdf* (Vol. 2020). Aalborg Universitetsforlag. Retrieved April 23, 2023, from https://vbn.aau.dk/ws/portalfiles/portal/405076852/PHD_LNN_E_pdf.pdf

Nielsen, L., Tølbøll Glavind, S., Qin, J., & Faber, M. H. (2019). Faith and fakes – dealing with critical information in decision analysis. *Civil Engineering and Environmental Systems*, *36*(1), 32–54. https://doi.org/10.1080/10286608.2019.1615476

Nielsen, M. H. F. (2007). Risk and Safety in Civil Engineering: Lecture Notes. In *Risk and Safety in Civil Engineering* [Report]. ETH. Eidgenössische Technische Hochschule Zürich. Swiss Federal Institute of Technology Zurich.

Oliveira, D. F. N., Vismari, L. F., Nascimento, A. M., de Almeida, J. R., Cugnasca, P. S., Camargo, J. B., Almeida, L., Gripp, R., & Neves, M. (2022). A New Interpretable Unsupervised Anomaly Detection Method Based on Residual Explanation. *IEEE Access*, *10*, 1401–1409. https://doi.org/10.1109/ACCESS.2021.3137633

Patil, A., Soni, G., & Prakash, A. (2022). A BMFO-KNN based intelligent fault detection approach for reciprocating compressor. *International Journal of System Assurance Engineering and Management*, *13*(2), 797–809. https://doi.org/10.1007/s13198-021-01395-2

Pimentel, M. A. F., Clifton, D. A., Clifton, L., & Tarassenko, L. (2014). A review of novelty detection. *Signal Processing*, *99*, 215–249. https://doi.org/10.1016/j.sigpro.2013.12.026

Pook, L. P. (2011). Variations on Simple Pendulums. In L. P. Pook (Ed.), *Understanding Pendulums: A Brief Introduction* (pp. 27–41). Springer Netherlands. https://doi.org/10.1007/978-94-007-1415-1_3

Raiffa, H., & Schlaifer, R. (n.d.). *Applied Statistical Decision Theory*.

Schölkopf, B., Williamson, R. C., Smola, A., Shawe-Taylor, J., & Platt, J. (1999). Support Vector Method for Novelty Detection. *Advances in Neural Information Processing Systems*, *12*. https://proceedings.neurips.cc/paper_files/paper/1999/hash/8725fb777f25776ffa9076e44fcfd776-Abstract.html

Sharma, J., Mittal, M. L., & Soni, G. (2022). Condition-based maintenance using machine learning and role of interpretability: A review. *International Journal of System Assurance Engineering and Management*. https://doi.org/10.1007/s13198-022-01843-7 van Eck, N. J., & Waltman, L. (2014). Visualizing Bibliometric Networks. In Y. Ding, R. Rousseau, & D. Wolfram (Eds.), *Measuring Scholarly Impact* (pp. 285–320). Springer International Publishing. https://doi.org/10.1007/978-3-319-10377-8_13

Velasco-Gallego, C., & Lazakis, I. (2022). RADIS: A real-time anomaly detection intelligent system for fault diagnosis of marine machinery. *Expert Systems with Applications*, *204*, 117634. https://doi.org/10.1016/j.eswa.2022.117634

Zhang, W., & Ma, X. (2016). Simultaneous Fault Detection and Sensor Selection for Condition Monitoring of Wind Turbines. *Energies*, *9*(4), Article 4. https://doi.org/10.3390/en9040280

Zhang, W.-H., Lu, D.-G., Qin, J., Thöns, S., & Faber, M. H. (2021). Value of information analysis in civil and infrastructure engineering: A review. *Journal of Infrastructure Preservation and Resilience*, *2*(1), 16. https://doi.org/10.1186/s43065-021-00027-0

Zhou, Q., Ning, Y., Zhou, Q., Luo, L., & Lei, J. (2013). Structural damage detection method based on random forests and data fusion. *Structural Health Monitoring*, *12*(1), 48–58. https://doi.org/10.1177/1475921712464572

Zuo, E., Du, X., Aysa, A., Lv, X., Muhammat, M., Zhao, Y., & Ubul, K. (2022). Anomaly Score-Based Risk Early Warning System for Rapidly Controlling Food Safety Risk. *Foods*, *11*(14), Article 14. https://doi.org/10.3390/foods11142076