TU DELFT

MASTER'S THESIS

Determine and explain confidence in predicting violations on inland ships in the Netherlands

Author:
Paul Bakker
4326091

Supervisor:
Dr. Tintarev

A thesis submitted in fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

COMPUTER SCIENCE
TRACK DATA SCIENCE

Web Information Systems
Department of Software Technology
Faculty EEMCS, Delft University of Technology

Abstract

For real-world problems even the most complex machine learning models can only achieve a certain accuracy. This makes it important to understand why a specific prediction is made. Explanations can provide human decision support by allowing human experts to assess the reasoning of the model as well as the correctness. Specifically, in this thesis, we consider the problem of predicting violations on inland ships in the Netherlands to help inspectors in the Dutch government deciding which ship to inspect. The main contribution is determining confidence in a prediction separately from probability and using this confidence estimation for deciding which prediction of violation to select as well as to explain.

With the limited number of inspectors and a large number of inland ships in the Netherlands, the global performance on all ships is less relevant. Instead, deciding the most qualitative predictions is more useful. Therefore, a measure of model confidence is determined to improve upon the traditional ranking based on probability. In the evaluation of this approach, no significant difference is found between the ranking based on probability for complex ensemble models. However, for simpler, more interpretable models, there is a significant improvement in using model confidence to re-rank.

The determination of confidence is further used to create explanations from the context of confidence. The goal of these explanations is to help an inspector in deciding whether to inspect an inland ship. This novel explanation approach justifies the confidence in a prediction by expressing features contributing towards the confidence. We perform a human-grounded user study evaluation to identify the *task effectiveness*, *perceived usefulness* and user trust compared to the explanations from the traditional context of probability. The results of the user study suggest the explanations of the confidence to be particularly useful for problems with a lower accuracy.

Thesis committee:

Prof.dr.ir. G.J.P.M. Houben,	TU Delft, chair
Dr. N. Tintarev,	TU Delft, supervisor
Dr J. Urbano Merino,	TU Delft

Preface

This report concludes my thesis project as well as studying at the TU Delft. Looking back makes me realize how much I have learned over these years and during this project.

I would first like to express my gratitude to my supervisor Dr. Nava Tintarev, for her continuing support throughout the project. Also, I would like to thank Dr. Emily Sullivan for her advice and guidance in the set-up and execution of this project as well as Dr. Oana Inel for her feedback on this report. Thanks should also go to the people in the Epsilon group for their feedback on my presentations.

I would also like to thank all the people at IDlab for their help and suggestions during the user study as well as including me in the larger discussions within the group. Specifically, I want to express my gratitude to Jasper van Vliet and Margje Schuur. This thesis would not have been possible without their enthusiasm and support. I also want to thank all the inspectors who participated in the study.

Last but certainly not least, I cannot begin to express my thanks to my family and friends. My deepest thanks go out to my parents and brother for always believing in me.

Paul Bakker
Delft, November 2020

Contents

Abstract **i**

Preface **ii**

1 Introduction **1**
1.1 Inspecting inland ships . . . 1
1.2 Confidence versus probability . . . 1
1.3 Ranking with confidence . . . 2
1.4 Explaining complex models with model confidence . . . 2
1.5 Overview . . . 3

2 Research context and framework **4**
2.1 Research context . . . 4
2.1.1 Inspecting inland ships . . . 4
2.1.2 Motivation . . . 5
2.2 Preliminaries . . . 5
2.2.1 Probability versus model confidence . . . 5
2.2.2 Interpretable models versus post-hoc justification . . . 6
Model-agnostic versus model-specific . . . 6
Global versus local explanations . . . 6
2.3 Requirements . . . 6
2.4 Research framework . . . 7
2.5 Datasets used . . . 8
2.5.1 Inland ships . . . 8
Preprocessing of the data . . . 9
2.5.2 Other datasets . . . 10

3 Related work **11**
3.1 Model confidence . . . 11
3.1.1 Confidence intervals . . . 11
3.1.2 Conformal Prediction . . . 12
Non-conformity function for Random Forest models . . . 13
3.1.3 Use cases for confidence score . . . 13
Confidence-weighted online learning . . . 13
Information retrieval . . . 14
Recommender systems . . . 15
3.2 Confidence displays . . . 16
3.3 Explainable Artificial Intelligence . . . 17
3.3.1 Interpretable models . . . 18
3.3.2 Post-hoc justification . . . 19
Model approximation . . . 19
3.3.3 Feature contributions . . . 20
SHAP . . . 21

Random Forest specific feature contributions . . . 21
3.4 Evaluation of explanations . . . 22
3.4.1 Evaluation metrics as proxy . . . 22
3.4.2 Human-grounded evaluation of SHAP . . . 23
3.5 Research gaps . . . 23

4 Predicting model confidence **25**
4.1 Definition of model confidence . . . 25
4.2 Conformal Prediction Framework . . . 27
4.2.1 Assumptions . . . 28
4.2.2 Prediction sets . . . 28
4.2.3 Conformal predictions . . . 28
Hypothesis testing . . . 29
4.2.4 Non-conformity function . . . 31
4.2.5 Transductive versus Inductive . . . 31
4.3 Experimental design . . . 32
4.3.1 Experiment 1: prediction set across significance levels . . . 32
4.3.2 Experiment 2: Meta conformity approach . . . 34
4.4 Results . . . 35
4.4.1 Prediction sets on binary classification . . . 35
4.4.2 Meta-Conformal classification . . . 37
4.5 Conclusions . . . 38

5 Ranking with model confidence **39**
5.1 Taxonomy Ranking algorithms . . . 39
5.2 Experimental design . . . 40
5.2.1 Experiment 1: Correlation between error and additional metrics . . . 40
5.2.2 Experiment 2: Pointwise ranking with confidence measures . . . 41
Pointwise ranking with confidence measures . . . 41
5.3 Results . . . 43
5.3.1 Correlation between error and additional metrics . . . 43
5.3.2 Pointwise ranking with confidence measures . . . 44
Using confidence to re-rank . . . 45
5.4 Conclusion . . . 46

6 Explanations based on confidence **48**
6.1 SHapley Additive exPlanations . . . 48
6.1.1 The Shapley Value . . . 49
6.1.2 SHAP value . . . 49
Compared against other additive feature attribution methods . . . 50
6.1.3 Approximation method . . . 51
KernelSHAP . . . 51
6.1.4 SHAP Interaction values . . . 52
6.1.5 Types of explanations . . . 52
6.2 Implementation of SHAP . . . 52
6.3 Exploration SHAP values between two contexts . . . 54
Difference between SHAP values . . . 54
Global feature importance . . . 55
Individual predictions . . . 55
Combining individual predictions . . . 57
Interaction between features . . . 57

6.4 Conclusion . . . 59

7 A human-grounded evaluation of confidence based explanations **60**
7.1 Evaluating explanations . . . 60
7.2 A human-grounded evaluation . . . 61
7.2.1 Research hypotheses . . . 61
Explanation effectiveness . . . 61
Explanation usefulness . . . 61
User trust . . . 62
7.3 User study . . . 62
7.3.1 Inspection Task . . . 63
7.3.2 Task ordering . . . 64
7.3.3 Experiment details . . . 64
Model used . . . 64
Instances used . . . 65
Participants . . . 65
Quality assurances . . . 65
7.3.4 Results . . . 66
Hypothesis 1: Explanation effectiveness . . . 66
Hypothesis 2: Explanation usefulness . . . 67
Hypothesis 3: User trust . . . 71
Results on individual inspection tasks . . . 73
Written reflections of the participants . . . 75
7.4 Conclusion . . . 75

8 Conclusions **77**
8.1 Summary . . . 77
8.2 Contributions and recommendations . . . 78
8.3 Conclusion . . . 78
8.4 Limitations and Future Work . . . 79
Conformal Prediction . . . 79
Ranking with confidence . . . 79
Evaluation explanation . . . 80

Bibliography **81**

A Determining confidence **87**
A.1 Prediction set behaviour . . . 87
A.1.1 Inland dataset . . . 87
A.1.2 Churn dataset . . . 89
A.1.3 Adult dataset . . . 91
A.1.4 Spambase dataset . . . 93
A.2 Correlation between error and metrics . . . 95
A.3 Comparison pairwise, listwise and pointwise approach . . . 96
A.4 Confidence reranking . . . 97

B Explaining based on confidence with the *churn* dataset **99**
B.1 Difference in SHAP values on *churn* dataset . . . 99
B.2 Global feature importances . . . 100
B.3 Single prediction forces . . . 101
B.4 Interaction values . . . 101

C Additional results user study **103**
C.1 Agreement between participant and prediction 103
C.2 QQ plots accuracy of both contexts 103
C.3 Nonparametric hypothesis tests 104
C.4 Usefulness features 105
C.5 Power analysis 108
C.6 Time taken between contexts 110

Chapter 1

Introduction

In recent years machine learning models are increasingly used in real-world situations. Simple models are predicting whether mail is spam and complex models tag photo images for easier search. The models are increasingly used by people who have no knowledge about the inner working of these models. While simple models can be made understandable with some text and visualizations, for complex models, their behaviour is difficult to understand. Even the maker of these models can have difficulty in understanding the output of these models. In real-world situations, it can be useful, even necessary, to understand why a certain prediction is made [1, 24, 31]. For example, a machine learning model used as an additional tool for a human-decision problem can incorporate explanations to assess the reasoning for a model and show the limitations of the prediction.

1.1 Inspecting inland ships

In this study, we specifically look at the real-world problem of inspecting inland ships in the Netherlands in cooperation with the Human Environment and Transport Inspectorate of the Dutch government, in Dutch "Inspectie Leefomgeving en Transport" (ILT).

Inspectors working for the inspectorate have to decide which inland ships they inspect. With the limited number of inland ship inspectors and at least 5.000 inland ships with a Dutch flag, this decision focuses on the ships most likely to be in violation. Currently, this decision is based on human expertise together with an application for retrieving information about a ship available to the inspectorate. The IDlab, a group of data analysts within the inspectorate, is currently working on a system predicting violations on inland ships. The long-term goal is to provide the prediction of violations in real-time to the inspectors as an additional aid. However, even the best performing models have a relatively modest predictive performance. Only showing the prediction can, therefore, quickly erode trust in the system, as a large number of predictions will be incorrect. Providing additional explanations of the predictions might help the inspectors in deciding which ship to inspect.

This thesis contains two main parts: (1) ranking evaluation with the inclusion of confidence on several machine learning models and datasets and (2) a human-grounded evaluation on the usefulness of explanations for inland ship violations from the context of confidence.

1.2 Confidence versus probability

With the advent of machine learning models present in everyday life, it is important to understand the limitations or uncertainty of these models and their predictions. The limitation of the model is often expressed with a probability. For these probabilistic models, a probability distribution is estimated, upon which classification is determined [66].

Model confidence, as described in this study, tries to determine the quality of this probability distribution estimation. Confidence is determined by expressing the prediction as a range with certain guarantees on error rate instead of the single point prediction of probability. A good illustration of the difference between the probability and confidence is expressed by the model predicting the outcome of the US 2020 presidential elections by The Economist [72]. This model predicts a 97% probability of a candidate winning the election. However, besides this single point prediction, a range of predictions in the number of electoral votes is also given. This range expresses 95% confidence the true outcome will lay within this range (259-415), with an average point prediction (356). Model confidence can be determined by looking at the size of this range; a smaller range gives more confidence in the average point prediction being correct. In this thesis, we use the Conformal Prediction framework, which uses conformity between instances, to predict these ranges with the use of prediction sets.

1.3 Ranking with confidence

Given that the number of inland ships in the Netherlands always vastly exceeds the number of inspectors, the overall accuracy of the model is less important than selecting the ships most likely to be in violation. The selection of the most qualitative predictions of violations is, therefore, an important contribution of this thesis. To achieve this selection the problem is modelled as a ranking problem.

The goal of ranking for the specific problem of violations on inland ships is sorting a list of predictions of violations in such a way that the violating ships are more likely at the top of the list, while non-violating ships are lower on the list.

The ranking of predictions in previous research focuses almost exclusively on information retrieval problems, such as ranking with document and query pairs or recommender systems [94, 98, 59, 16]. Evaluation of these approaches is therefore done with specialized datasets in information retrieval. In this research, several traditional binary classification datasets are used, as well as the real-world dataset of violations on inland ships in the Netherlands. A primary contribution is a novel approach of ranking with the inclusion of the estimated model confidence based on conformity. For the evaluation of this approach, the baseline of sorting based on the probability is used.

1.4 Explaining complex models with model confidence

The accuracy of machine learning models for real-world problems can be low, as is the case for the real-world problem of predicting violations on inland ships in the Netherlands. Even with high accuracy, just providing the prediction does not guarantee trust in the system. Therefore, in the field of eXplainable Artificial Intelligence (XAI), a large number of approaches are discussed to explain the model or justify the output of these models [26, 31].

With the determination of a measure of model confidence, the second part of this thesis looks at using this additional measure to explain the predictions of violations on inland ships from the novel context of confidence. Instead of justifying the probability determined by a Random Forest model, the confidence as determined by the Conformal Prediction framework is justified. For both the context of probability as well as confidence, the SHAP framework justifies the predictions by determining feature contributions.

With the numerous approaches of explaining predictions of complex models, evaluation to determine the quality of these explanations becomes more important [53, 24]. Evaluations of explanations are difficult due to the human-centred nature of providing explanations. A main contribution of this thesis is a human-grounded evaluation of the explanation from the context

of confidence. For this evaluation, a user study is performed where participants are shown explanations for both the context of probability as well as confidence. Participants have to decide whether or not to inspect the ship based on the prediction and explanation given. During the user study the *task effectiveness*, *perceived usefulness* and *user trust* is used to evaluate the quality of the explanation from the two contexts.

1.5 Overview

Next, the research context is laid out in Chapter 2. The problem we want to solve is described together with the requirements for the specific approach. From the context and requirements, the research questions are defined. Finally, the datasets used to answer these questions are discussed.

In Chapter 3, a literature review is given with two main parts. The first part looks at research determining the confidence of a prediction separately from probability. Also, research into the use cases of this confidence in several applications is discussed. The second part of the literature review gives an overview of the topic of eXplainable Artificial Intelligence (XAI) with an overview of the different approaches to create explanations as well as techniques for evaluating the explanations.

In Chapter 4, the Conformal Prediction framework is used to determine the confidence of individual prediction, together with two experiments to determine the behaviour of this approach and whether model confidence would be useful in the ranking of instances.

In Chapter 5, the model confidence is used to rank the predictions for several datasets and classifiers. The correlation with the confidence of the model and the error rate of this model is also determined.

In Chapter 6 and 7, confidence is used as a new context for explaining the complex random forest model predicting violations in inland ships in the Netherlands. Chapter 6 contains an introduction into the workings of the SHAP explanation framework, together with a data analysis comparing the SHAP values between the context of confidence and probability. In Chapter 7, the contexts are evaluated by a human-grounded evaluation via a user study.

Finally, in Chapter 8, a summary of all the work is given together with the limitations, future works and conclusions.

Chapter 2

Research context and framework

In this chapter, the research context is first described, followed by the research questions as well as the research framework used to formulate and answer these questions.

2.1 Research context

The main contribution for this research is evaluating the usefulness of incorporating model confidence into ranking predictions and explanations for these predictions. Ranking is used to select the prediction the system is most confident in and evaluate the model confidence. To evaluate the explanations based on the model confidence a user study is performing together with the IDLab of the Human Environment and Transport Inspectorate of the Dutch government, in Dutch "Inspectie Leefomgeving en Transport" (ILT).

2.1.1 Inspecting inland ships

The inspectorate has the task of performing inspections on inland ships. There are several criteria checked; working conditions of the crew, correct documentation of the ship, storage safety, etc. The inspectorate does keep a risk assessment related to violations for ships it has inspected in the last three years. However, a large number of inland boats have no estimation about the potential of such violations. For these ships, inspections are performed based on the expert opinion of the inspectors of the inspectorate. The goal of the IDLab is to use data available related to these ships to help the inspectors with the decision on which ship to inspect.

At the moment, the inspectorate uses the tool InspectieVIEW. This system allows the inspector to log inspections. Inspectors can log basic information about the ship, whether a violation took place and action taken to resolve the non-compliance. Most importantly, inspectors can write some additional information about the inspection with the use of a simple text box. Therefore, this system is not a pro-active system; it just provides information. The accuracy of inspectors finding violating ships is currently below 40%.

This research is built upon prior research of the IDLab. In this prior research, Random Forest Tree models were used to predict if 'unknown' inland boats passing through the Netherlands would violate the law. This pro-active system can therefore advise inspectors on which ship to inspect and is no longer deciding based on their expert opinion.

The limited number of inspectors cannot inspect all ships. Therefore, it has to be determined which ship to inspect. Or more general, a selection has to be made in the test instances. Furthermore, this model is rather complex, and the inspectors are not able to fully understand why the prediction is as it is. Although a decision tree is relatively easy to explain, the large number of trees necessary in the Random Forest model reduces the understandability of the model. Making this complex model understandable to inspectors without any background into computer science is not straightforward. However, the explanation can aid the inspector in understanding which information the model bases its decision on and help the inspector in the decision-making process.

2.1.2 Motivation

There are two primary motivations for the determining of model confidence for the problems of ranking and explaining the predictions of the real world-problem of violations on inland ships in the Netherlands.

- Firstly, previous research found specific approaches of determining confidence separately from probability improved the ranking for the problem of document ranking in information retrieval [94, 98], the ranking of recommendations [16, 59] and ranking in an online learning setting [39]. More details on these studies can be found in the related work in Section 3.1.3. However, instead of determining model confidence based on variation in the probability distribution, we use the conformity between instances, which has the same guarantees in error rate [77]. Additional benefits in this approach are the possibility to determine how new instances conform to the training data and contrastive information about the class not ultimately predicted. A motivation is determining whether this approach in estimating model confidence improves the ranking similarly to previous research in traditional classification problems.
- Secondly, previous research found the displaying of confidence in explaining predictions improved user satisfaction and user agreement [61, 60]. However, explaining the reasoning behind the confidence estimation is not performed. This, while numerous approaches exist for the explaining of complex models [26, 31]. For classification models, this means explaining the probability distributions estimated. More details on the research into confidence displays and techniques explaining complex models can be found in Section 3.2 and 3.3.

  For this study, we will use explanation techniques on the determination of model confidence. This creates another context from which to explain the prediction of violation on inland ships. We want to determine whether this other context improves the explanation from the traditional context of probability.

2.2 Preliminaries

Before moving towards the requirements of this research some preliminary definitions are described.

2.2.1 Probability versus model confidence

With the advent of machine learning models present in everyday life, it is important to understand the limitations or uncertainty of these models and their predictions. The limitation of the model is often expressed as with a probability. In the next sections, a brief distinction between this probability and model confidence is described.

Using machine learning models ranging from simple models like linear regression to more complex models like neural networks have been used extensively for classification problems. These models almost always are probabilistic, meaning that the output of the model is the probability of a certain class $P(y|x)$. The probability of an instance belonging to a specific class is determined with a probability distribution. These probabilistic models are used in practice because realistic decision making often necessitates recognizing uncertainty. With the incorporation of this probability, uncertainty can be measured, for example, the chance of rain in a weather forecast. In this study, we look only at binary classification problems, making the probability distributions in this study Bernoulli distributions. Model confidence uses these determined probability distributions and tries to determine the quality of these distributions. Or, in other words, it tries to determine how confident the model is in the correctness of the probability distribution. There

are several approaches for this determination of this confidence. Examples are prediction interval determination and conformal prediction. In section 3.1, a review of these approaches is given.

2.2.2 Interpretable models versus post-hoc justification

The field of eXplainable Artificial Intelligence (XAI) researches techniques for making complex machine learning models understandable towards the maker of the model as well as to a potential user of the model. There are two main distinguishable groups of approaches.

Interpretable models. These are explanations for models which are intrinsically understandable. Simple visualizations of the model can explain why a prediction was made. Examples of these models are linear models or decision trees. Research of explanation of these model look at the best approach to make the model understandable to the user.

Post-hoc justification For more complex models, it is not possible to easily understand why a prediction is made. Post-hoc justification techniques do not try to explain the model itself. Instead, it tries to justify the predictions by the model.

Model-agnostic versus model-specific

For both the determination of model confidence as the creation of an explanation model-agnostic or model-specific approaches exist [77, 6]. Model-agnostic approaches work on any classifier, as these approaches see this classifier or model as a black box. These approaches work based on the input and output of such a black-box model, not the structure or properties of a specific model. Model-specific approaches instead do leverage additional information about the model. For example, a model-specific approach could look at the paths of the individual decision trees in a tree ensemble to decide a confidence score or explanation.

Global versus local explanations

The final distinction between approaches is local versus global explanations or interpretation [53, 6]. Global explanations propose to explain the global working of a certain model. An example of a global explanation is the general feature importances to make a prediction. Local explanations propose to explain individual predictions or smaller subsets of predictions. An example of a local explanation is the features most contributing to an individual probability prediction.

2.3 Requirements

For the overall selection and explanation of the overall research a number of requirements can be defined.

Predicting violations. The predicting of violations on inland ships has the following attributes:

- *Binary classification*. There are several categories of violations on inland ships. However, due to the limited amount of information available the problem is reduced to a binary classification problem; *in violation* or *not in violation*
- *Data sparsity*. As data is obtained from several sources, not all having the same ships available, for a lot of inland ships data is missing. This makes the combined dataset sparse.
- *High dimensional*. With the combining of several sources, the dimensionality of the data is high, with 252 features for each instance.

Selecting predictions. A selection on ships the model is most confident in has to be made.

- *Model-agnostic*. There are several classification problems researched in the inspectorate. The selecting of most confident predictions should work on any machine learning problem.
- *Combinations*. Model confidence should not be the only criteria in selecting ships. The quality of the prediction should be combined with other factors such as predicted probability.
- *Improve accuracy*. The selection of ships should increase the relative performance of the predictions compared to the overall performance of the system.

Explaining predictions. The prediction of violation has to be explained.

- *Justifying the results*. The difficulty of this particular problem requires complex models to achieve the best performance. In order to explain the model to novice users, the predictions should be justified, instead of explaining the model itself.
- *Model-agnostic*. There are several classification problems researched in the inspectorate. The creation of an explanation should not be specific to a model.
- *Local explanations*. The decision on boarding a ship means the explanation should be tailored to the specific ship, not a global explanation of the model.

To summarize; the system must perform binary classification on highly dimensional and sparse data. A selection in predictions should be model-agnostic, allow for multiple measures of quality and improve relative performance. Explaining the system must justify individual predictions in a way that is model-agnostic.

2.4 Research framework

Given the research context and requirements, the main research question of this thesis is as follows:

How can we predict the confidence of a complex model to select predictions and provide inspectors with local model-agnostic explanations of this confidence?

In order to answer the main research question, four additional research questions are formulated.

The determination of confidence is often conflated in research with a number of different definitions. Confidence is often used interchangeably with probability. Other definitions would be risk, uncertainty or reliability. The first research question is therefore:

How can we predict the confidence score separately from class probability?

In order to answer this question, relevant literature is reviewed, looking at different approaches defining this notion of model confidence. It was decided to use the Conformal Prediction framework to determine local model-agnostic predictions of confidence and credibility.

Due to the limited number of inspectors and a large number of inland ships in the Netherlands the overall performance of the model on the whole test data is less important than the relative performance of a selection of the test set. A simple approach of deciding which ships to inspect is sorting the predictions by probability and only taking the most probable instances according to the model. With the determination of model confidence for individual predictions, sorting can be based on this measure. In order to determine the usefulness of the additional quality measure, the problem of selecting ships to inspect is modelled as a ranking problem. This created the second research question:

Can model confidence predictions improve the ranking of predictions?

This question is answered with an evaluation of a number of different approaches to rank the predictions of violations. These approaches are compared to the baseline of sorting based on probability. The ranking is also evaluated for several other datasets to test the general usefulness of the inclusion of model confidence.
With an improvement found in using model confidence in sorting the predictions of violations of inland ships, the second focus of this thesis is using this model confidence to explain individual predictions. This created the third research question:

How can confidence prediction be used to generate model-agnostic local explanations?

In order to answer this question, a literature review was performed in the field of eXplainable Artificial Intelligence (XAI). From this review, the SHAP framework is selected for this particular problem. The model-agnostic Kernel-SHAP is used to approximate a random forest model with a linear model. This simple interpretable model creates local explanations with feature contributions. Data analysis is performed to determine how the model confidence determined by the Conformal Prediction framework differs from the probability of the base machine learning model.

The explanations based on confidence need to be evaluated. The goal of these explanations is helping the inspectors in their decision whether or not to inspect a ship. In order to evaluate this, the final research question is as follows:

How are explanation based on confidence received by users?

A user study is performed to answer this question. To evaluate the quality of the explanations, a human-grounded approach of evaluation is used. The reason for this approach is to make the results generalize to other problems as well as making it possible to include more people in the study.

2.5 Datasets used

A number of experiments with additional measures of *confidence* and *credibility* are performed with several different datasets. These are the dataset of violations of *inland ships* as well as the popular machine learning datasets *churn*, *adult* and *spambase*.

2.5.1 Inland ships

The data set used in this research comes from previous research done by the Human Environment and Transport Inspectorate. In this previous research data from multiple sources within the

Dutch government was collected and combined to help inspectors in choosing which ship to inspect. This data consists of a total of 274 variables for 7214 inspections. Inspection data of the inspectorate was used to link separate databases together; for example, the European Number of Identification (ENI number) and the registration for the Chamber of Commerce to link ships and companies. Next, a brief overview of the different datasets is described.

Basic features of the ship A number of the variables of the dataset contains basic information about a ship. This is, for example, the type of ship, when it was built, the size of the ship, in which country was it built, etc.

Previous behaviour and the cargo of the ship Information about the previous behaviour of the ships is also included in the dataset, as well as variables describing the cargo on the ship. The historical location of a given ship is also contained. An example of variables describing the cargo is whether it is labelled as dangerous.

Fuel information Based on the fuel pass used by ships in the Netherlands, there is a lot of information about the fuel history of the ship. Examples of such a variable are the locations and time the ships were filled up or the amount of fuel.

Information about the owner of the ship A number of variables in the final dataset are related to the owner of the ship, as well as the previous owners. These give information about the certificates the owner has, but also how many ships the owner has and how many times those were in violation. Other variables describe how long the company has been active, what its main activity is and other activities of the company.

Historic information of the inspectorate An important dataset used was the historical data of the inspectorate. This data was used to derive feature on the ships such as the number of previous violations, when was the ship last inspected and what kind of violations previously took place on the ship.

Aggregated features With all the different information sources of inspection, aggregated features were also determined for the individual inspection instances. An example of such an aggregated feature is the number of previous violations on a ship. In this case, all previous inspections of the ship in question are used to determine this feature. Other examples are the average time the ship takes to fuel or the time between the inspection in question and the previous inspection.

Preprocessing of the data

Based on all the variables already collected, the preprocessing for this research will limit itself to feature selection. The main reason as to why feature selection is performed in this research is to get a better understanding of the model and the data. This understanding is useful for the subsequent part of the creation of explanations. Feature selection, and more specifically, feature importance is closely related to popular explanation methods [73, 78].

Feature selection means determining which features contribute the most to the prediction of a machine learning model. This can be done either automatically or through manual inspection on the different features [65]. Feature selection is a useful strategy for a number of goals; reducing the complexity of a model for high-dimensional data, improving the generalizability of the model, improving the performance of the model or making the model and data more interpretable [50]. In order to determine which feature contributes the most to the predictions, many approaches

have been developed. Most machine learning framework such as Scikit-Learn has built-in feature importance scores for a large number of machine learning models, for example.

In Random Forest models and decision trees in general, every node in the tree contains a condition where samples are split based on a single feature value. The goal is to get samples with similar values of that specific feature to end up in the same set, where all instances contain the same class label. The condition is determined based on the impurity, which in the case of a default random forest model is the Gini impurity (or entropy) [13]. During the training of a model it is possible to keep track of how much each feature contributes toward decreasing this Gini entropy. This is the technique Scikit-Learn uses to calculate feature importances. This means that there is no additional step of calculation needed to determine the feature importances when using this technique. A drawback of this approach to determining feature importance is that it tends to favour predictor variables that are continuous or categorical with a large number of levels [5].

There are also other more advanced techniques to estimate the feature importances, an example being permutation accuracy importance. The permutation accuracy importance looks at the effects of shuffling a single variable in the overall accuracy of the model. This makes it model-agnostic and relatively efficient. A significant drawback of this approach is that it tends to overestimate the importance of correlated predictors [84]. There is a large correlation between the different features in the dataset of this project, so it was chosen not to use this approach. Instead, it was chosen to use a drop-out procedure to remove bias in the estimation of feature importances. This procedure drops a single variable and looks at the effect this has on the performance of the overall model. The reason for choosing this method is that it was found to be unbiased in the selection of features [83]. And, while this technique is computationally inefficient due to having to retrain the model for each dropped feature, due to the relatively small dataset of this project, this is not a problem.

Using the drop-out technique, 22 features were determined to have no impact on the performance of the Random Forest model. To speed up the evaluations in this thesis, these features were not used, resulting in a total of 252 features.

2.5.2 Other datasets

Besides the inland ship dataset looking at a real-world problem, this section uses three additional datasets across all the different evaluations. The reason is twofold; firstly, to evaluate the generalizability of the results. Secondly, the traditional machine learning have nice properties, such as less noise, strong i.i.d. assumption etc. No preprocessing was performed in these datasets.

Adult dataset The first tradition machine learning dataset is the *adult* dataset from Kohavi et al. [41]. This multivariate dataset contains 14 features for 48842 instances. The goal is to determine based on 1994 US Census data like age, education and race to determine if the person makes over 50K a year, making it a binary classification problem.

Churn dataset A similar dataset to the *adult* dataset is the *churn* dataset [11]. The multivariate dataset contains 20 features for 5000 instances. The goal is to predict whether the customer churned based on telephony account features like the monthly charges, years on the contract and age of the customer.

Spambase dataset The final dataset is the *spambase* dataset [33]. The multivariate dataset contains 57 features for 4601 instances. No categorical data is contained in this dataset. The goal of this dataset is to predict if an email is spam.

Chapter 3

Related work

This chapter surveys relevant literature for the two main topics of this thesis. One goal of this literature review is describing the current approaches in determining the confidence of predictions and describing the current situations in which these are used. With this understanding, the confidence determination will be used in selection and ranking problems for traditional machine learning problems. The second goal is reviewing the topic of eXplainable Artificial Intelligence (XAI). The different types of approaches are reviewed in order to determine which approach is most suitable for explaining the predicted confidence. Evaluation methods are also described to determine how to evaluate this new explanation approach.

3.1 Model confidence

In order to determine the validity of a prediction made by a machine learning model, an important notion is the confidence in its prediction. The definition of confidence is sometimes conflated with the probability given by such a model. This is not unreasonable, as the probability of a prediction does incorporate the notion of uncertainty. However, in this review, the notion of confidence can most easily be described as a quality measure of the prediction. We do not expect a model to determine the *true* probabilities of a problem perfectly. This can be due to the model being simple to be understandable or due to limitations of the dataset. Instead of answering the question of *How probable is an event?*, the question we try to answer is *How confident are you in the prediction?*. Note that the prediction in the case of a probabilistic model is the probability.

The current research into quantifying or estimating confidence (also called uncertainty, reliability or risk in some papers) there are two main approaches described in this chapter. The first approach is using the variance of predictions as a measure of confidence expressed with a confidence interval. The second approach is the Conformal Prediction framework, which looks at how well test samples conform to the training data and determine a confidence score based on this conformity.

3.1.1 Confidence intervals

A concept used to quantify the uncertainty of the predictions from this difference in data distribution is confidence intervals. There have been numerous approaches for estimating these confidence intervals for different machine learning models [79]. These intervals are an estimation on the interval in which data points will lay with a certain probability. In other words, confidence intervals cover new observations with high guaranteed probability. This differs from the standard probabilistic machine learning approach; instead of the mean of the probability distribution, a range of the probability is determined. For example, 95% confidence intervals will guarantee with at least 95% probability the true probability will lay in the ranges determined. This does not mean that for a single prediction and range there is a 95% probability.

The interval computed from a given sample either contains the true probability, or it does not. Instead, the level of confidence is associated with the method of calculating the interval.

The confidence coefficient is simply the proportion of samples of a given size that are expected to contain the true probability. Meaning that for a 95% confidence interval, if many samples are collected and the confidence interval computed, 95% of these intervals contain the true probability.

A 100% confidence interval can trivially be determined by simply including the whole probability space. However, there is no use for this interval. The goal and evaluation for the prediction of these intervals is making the intervals as small as possible while still guaranteeing the percentage of observations laying within the interval.

Intervals are easily calculated for simple linear regression models. However, for more complex models, such as Random Forest, more complex methods have been proposed to estimate these intervals. Next, three of such approaches is given.

Quantile Regression Forest. A popular method for estimating the confidence interval in a Random Forest model is the Quantile Regression Forests [63]. In this study, the aim was to determine if the random forest can give information on the conditional distribution of the response value.

As the name suggests, Quantile Regression Forest uses the concept of quantile regression in a random forest model. While standard regression tries to estimate the conditional mean of a response variable given a certain input, quantile regression looks at keeping more information about the conditional distribution of the model. An example of such information could be the dispersion of observations around the predicted value.

Normally, a random forest model only keeps the conditional mean in the leaves of the different trees [13]. In the proposed Quantile Regression Forest, the value of all the different observations in this leaf node are kept, not just their mean. This is done in order to assess the conditional distribution [63]. Using this distribution, it is trivial to determine the confidence interval; the range between the preferred percentiles of the distribution of the response variables in the leaves. The width of the confidence interval gives the variation of new observations around the predicted values. The smaller the width, the more confident the model is in its prediction.

Monte Carlo estimation. In another study, a different method was proposed to determine the confidence intervals of the Random Forest model according to a Monte Carlo approach [17]. This means that a number of random forest models are parameterized by resampling the dataset. In these models, the mean and variance of the prediction across all tree levels for each observation are kept, not the whole distribution. For each of the models, a hold out sample is kept and used to approximate the confidence interval of the overall system.

Jackknife estimation. A study by Wager et al. [91] looked at noise when trying to estimate the confidence intervals. Monte Carlo bias was found to be the prominent factor in confidence intervals getting too large. In this particular study, the Jackknife and Infinitesimal Jackknife estimators were used to estimate the confidence interval by estimating the variance of the distribution in the leaves of the random forest tree. The basic idea of the jackknife is to omit one observation and recompute the estimate using the remaining observations, commonly called the leave-one-out approach.

3.1.2 Conformal Prediction

Besides the use of confidence intervals, there is another approach to estimate model confidence; measuring conformity to determine a confidence level. A conformity measure tries to indicate how typical a given data sample is. The basic idea of this measure is defined in a paper by Shafer et al. [77]. Separate confidence levels are determined for individual samples based on this conformity measure, with the only assumption being that the training and test samples are drawn independently from the same distribution. These confidence levels have the same guarantees as

the confidence intervals discussed earlier. To determine the conformity one uses each class label as the prediction of the new sample. For each of the labels one looks at how well the new sample conforms to the training data. In other words, it estimates how typical the test sample is compared to the other samples from the training set.

For each of the labels assigned to the individual sample, a p-value indicating this conformity is determined . The label with the highest p-value is the predicted label for the individual sample or a prediction set containing all labels having a p-value above a certain significance level is returned. The second highest p-value gives the probability of another label being the actual label. So the confidence of the prediction can be defined as one minus the second-largest p-value among the potential labels [64]. The most difficult part of the conformity framework is the determination of the p-value for each of the class labels. This determination can be done with the use of model-agnostic or model-specific non-conformity functions. These functions determine the p-value for individual samples. The model-agnostic functions do not rely on any structure of the underlying model; these only look at the input and output. A comparison of these functions was performed by Johansson et al. [34].

Non-conformity function for Random Forest models

A number of different model-specific non-conformity functions for a random forest classifier have been proposed, with a couple of examples being [23, 92, 9]. A study by Bhattacharyya looked at evaluating these different functions with each other [9].

Four different non-conformity functions were compared, with the first being a function based on the proportion of the trees in the ensemble that votes for the actual true class, as defined in [23]. The non-conformity score is simply one minus the proportion of trees that vote for the actual class label. This measure of conformity (and therefore confidence) still looks at the variance of the predictions. However, the main difference compared to previously discussed methods is that the variance is now determined between the different weak learners in the ensemble method. This is different from the variance estimations discussed earlier, as they looked at the variance of the predictions when using different sub-samples of training data.

The second function works on a proximity basis, as defined in [92]. This is a measure of closeness of samples in the meta-space and can therefore be seen as a k-nearest neighbour-based non-conformity function. The neighbours, in this case, are the different samples in the training data. The non-conformity scores are defined as the ratio of the sum of k-nearest neighbour distances in the label space with samples of the same class to the sum of k-nearest neighbour distances with samples of the other classes [92]. The main idea behind this function is that the nearer to the samples of a certain class the test sample is, the higher the chance of the test sample belonging to this particular class. The distance between samples is calculated as the agreement between the trees in the ensemble. The agreement is expressed by how many times the path in a weak learner is the same for two samples [92]. The other two functions are slight alterations of this second function, focused on class imbalance and data sparsity.

3.1.3 Use cases for confidence score

In the previous section two main approaches for the determination of confidence were laid out. In this section, research into the current use cases for this confidence score is discussed.

Confidence-weighted online learning

The concept of confidence in the field of online learning has been actively studied in recent years. Online learning algorithms are unique in the fact that they operate on a single instance at a time,

updating the rules or weights each iteration. These updates are simple, fast and make few assumptions about the underlying data. These algorithms are popular in the field of Natural Language Processing (NLP) due to the fact that these algorithms can process its input piece-by-piece in a serial fashion. This means that there is no need to have the whole input from the start in order to start processing. Due to similarity in concept between this single-instance learning and the bagging method of ensemble learning methods studies related to confidence in the field of online learning is discussed in this section.

The use of confidence as a weight in these kinds of algorithms was proposed by Dredze et al. [25]. In this study parameter confidence information was added to a linear online learning algorithm. This parameter confidence information is determined by modelling a diagonal Gaussian distribution. The standard deviation of this distribution represents the confidence of the mean parameter value. This distribution is used as memory for the NLP task in order to determine commonality of features, as the confidence of the weight of the features being correct increases with more samples of this feature. These estimates of confidence are then used to influence parameter updates. Instead of equally updating every feature weight for the features present in an instance, the update favours changing more low-confidence weights than high-confidence ones. This updating of the weights and the determination of the distribution is modelled as an optimization problem.

The implementation of the confidence-weighted linear classifier was improved in the following year by the same researchers [19]. The rule that the distribution over the parameter vector is updated each round causes aggressive updates in the original paper. This can cause over-fitting when the data is not linearly separable. The new proposed algorithm differs in the fact that it makes decoupled updates of the mean and confidence parameters of the distribution and softens the hard constraint of CW related to when to update.

These algorithms are used in research to solve a large number of problems. Examples are the classification of phishing emails [8], the detecting of malicious URLs [57] and text categorization [18].

A study by Khalid et al. [39] used the confidence-weighted classifier concept in a bipartite ranking algorithm. In a traditional bipartite ranking problem, instances come from only two categories, positive or negative. The goal is to learn from these samples a ranking function that ranks future positive instances higher than negative ones. In this study, an online bipartite ranking function is adapted with the updating of the ranking function being done with a confidence weighted learning approach from [93]. This also uses the variance of the Gaussian distribution of each feature as a metric for confidence and updates the ranking algorithm more aggressively for low confidence instances.

Information retrieval

In the field of information retrieval, a primary topic is the retrieving of the most relevant documents for a user's information needs. To effectively decide which documents to retrieve and which are most useful, the probability ranking principle is most often used. This principle tries to rank the documents in decreasing order of relevance, assuming that this maximizes the effectiveness of the system.

In a study by Wang et al. a method for ranking of documents based on a mean-variance paradigm was proposed [94]. Instead of relying only on the mean relevancy probability estimated based on the estimation of the relevancy of the documents in the ranking, it is assumed that the estimation of the relevancy of the individual documents has their own probability distributions. From these distributions the variance in the estimation can be determined. This is in the paper

described as the risk of the prediction made. An evaluation was done with a number of risk levels. In other words, weighing the variance higher or lower when determining the overall ranking. Metrics used for evaluation of the ranking the commonly used normalized discounted cumulative gain, precision and Mean Reciprocal Rank. The proposed method was compared against a user-based, item-based and Probabilistic Latent Semantic Analysis. The user-based and item-based rankers are based on collaborative filtering. It was found that the proposed ranking method based on the mean-variance paradigm outperformed the other basic rankers. A possible explanation given for this improvement was the fact that the proposed method looks at correlations between documents, similarly to conformity measures discussed before.

A study by Zuccon et al. [98] built upon the proposed method by Wang et al. [94] with three main differences; Firstly, the relevancy probability is not estimated by a point-wise estimator, it is determined by taking the mean of the probability distribution. Secondly, instead of substituting the variance in the estimations obtained for a document with the variance of the scores of the documents already ranked, it is again taken directly from the probability distribution of the single document. And finally, instead of approximating covariance between the relevance distributions of documents in terms of the correlation between documents features, the covariance is computed between distributions associated with different documents.

A similar evaluation was done when compared to the study by Wang et al. [94], and similar improvements were found when compared to traditional ranking algorithms. However, the performance of the two ranking algorithms based on the mean-variance paradigm was not compared directly against each other.

Recommender systems

The effect of including a measure of confidence has also been used to improve the performance of recommender systems in recent years. A study by Mazurowski [59] looked at a number of measures for defining confidence for a collaborative filtering recommender system. The following measures representing confidence were compared; the number of ratings by the user, the number of ratings for the particular item, the variance in ratings for the particular rating and three measures which look at the variation in the predicted rating when trained on different training data via the RESAMPLE method, similarly to [91]. All these measures can be used for all the different approaches of collaborative filtering, as they do not rely on any specific part of the algorithm.

Determining the confidence of a recommendation system was proposed in a study by Cleger-Tamayo et al. [16] to improve the ranking of the recommendations made for a movie recommendation system. It uses new input features taken from the original collaborative filtering model; the entropy of the ratings of a given movie, the average rating of neighbours and the number of neighbours that did not rate the particular item. The entropy looks at how concentrated the different rating are for a particular item. A separate model for determining the confidence in a certain rating was proposed; a decision tree. The binary determination of confidence was evaluated with a ranking problem.

3.2 Confidence displays

Over the years, there have been numerous studies looking at using confidence displays for helping users of different computer systems understanding and trusting the system. People with no background in machine learning or other prediction models would and are using these powerful tools in numerous decision processes. Examples are the severe weather predictions, polls of government elections or recommendation systems. For a lot of people using these models without any background in statistical modelling or machine learning these models are a black-box. This can lead to users not trusting the output of the system or overly relying on this output without understanding the limitations of the predictions made. In this section, studies are discussed which look at displaying the confidence of a system to a user and evaluate the effects.

A study by McNee et al. [61] showed the confidence of a recommendation from a recommender system of movies. For recommendation systems, collaborative filtering is the default method for making these kinds of recommendations. The idea behind collaborative filtering is having users rate items and predict the rating of that item for a new user based on the similarity between the users. A drawback of this method is the sparsity of ratings for obscure items. As the prediction is based only on these ratings and no meta information of the item itself, an item with only a few ratings is going to result in a risky recommendation. The simple metric of the number of ratings of an item is used as a confidence score [61]. This confidence score was presented alongside the recommendations made for a number of tasks in an A/B test. This evaluation was task-based where users were asked to perform three movie selection tasks in different risk scenarios. The selection was logged in order to see if users' behaviour changed when the confidence score was shown. After each task, a number of questions were asked related to user satisfaction and acceptance. In the experiment, it was found that adding the confidence display increased the user satisfaction overall. Similarly, the showing of confidence scores altered the users' behaviour. In the risk-averse scenarios, users were more likely to avoid low confidence predictions while in the risk-seeking scenarios users were more likely to choose predictions with low confidence.

A training phase was also presented to a certain number of users. When measuring user satisfaction between users trained in the confidence display and users just are shown the confidence displays without any explanation, mixed results were found. For new users of the system, user satisfaction increased when trained in the confidence display, while it decreased for experienced users. The possible explanation given for this result is that the training phase could plant a *seed of doubt* by increasing the awareness of the experienced users that the recommendations have varying amounts of accuracy [61].

The effect of a confidence display for a system aiding a human expert in a decision process was also evaluated in a study by McGuirl and Sarter [60]. The human experts in this case are fighter pilot using a system based on a neural network which predicts icing situations. The metric for the confidence used is the probability of the binary classifier, which is categorized into three groups (low, variable, high). Similarly to the previous study, an A/B test was performed with only half of the participants in the experiment being shown the confidence display. The behaviour of the pilots was again logged, and a survey was held in order to determine user satisfaction and trust.

Pilots who were shown the confidence display were more likely to comply with the decision made by the system predicting icing events and perform mitigation techniques when it predicted an icing event [60].

We found only one study looking at evaluating the interpretation of confidence intervals [32]. In this study, a single bound of the interval was presented to participants of a user study and compared to the traditional approach of showing the mean percentage. An example of this approach

would be "At least 60% confident", this being the lower bound of the confidence interval instead of the single probability score of "73% confident".

The studies laid out in this section have shown that using confidence display can lead to users trusting the system and being more satisfied with the predictions. It could also increase the overall performance of a task when a system is used as a decision aid. This while the metrics used for the confidence prediction is in both case quite basic. Therefore, more comprehensive measures of confidence, as laid out in section 3.1, is a worthwhile direction for further research specifically for systems with user interaction.

3.3 Explainable Artificial Intelligence

A major topic of recent interest in the field of artificially intelligent systems is the notion of explaining the decisions, recommendations, prediction or action made by this system. As discussed earlier, these kinds of systems are commonly used by people without any background in the underlying working of the system. Examples are the recommendation systems for music or movies, product recommendation by Amazon or severe weather forecasts. Most of these systems suffer from opacity, meaning that it is difficult to get an understanding into their internal workings, especially for systems with deep learning methods. Furthermore, while the output of probabilistic models offers clear, direct, numerical probabilities of events, for example 30% chance of rain, if only this probability is shown the value undergoes interpretation into a subjective sense or feeling and influences how people act upon these events [29]. How probabilities change over time or between instances also influences the subjective interpretation of the probabilities [58]. The concept of Explainable Artificial Intelligence (XAI) proposes to move to more transparent AI. The concept is used in a number of techniques to produce more explainable models without impacting the performance of the predictions made.

Explanations for decisions made by AI systems first appeared in the context of rigid rule-based expert systems, such as decision trees [1]. The aim was to show the decisions based on the rules created by the system. An example of such an explanation framework was given in 1983 by Swartout et al. [86]. The explanations generated with this technique can be seen as intrinsic in the rule-based models. For example, the rule-based model of a decision tree can be explained by following the path in the tree for a particular instance.

In the machine learning community, the concept of explanations started with simple visualizations to assist the designers of this model in understanding the behaviour of the system. Recent examples of these visualization techniques are proposed for convolutional neural networks by Zeiler and Fergus [97] and recurrent neural networks by Karpathy et al. [37]. While useful for machine learning experts, for users of such systems without the background knowledge it is not useful additional information.

Motivation for explanations The explanations for different problems have a number of motivations, often specific to the problem at hand.

A common motivation is *assessing the reasoning* of the model [24, 2]. The reasoning can show models focused on spurious relationships in the training data or other incorrect reasoning for a given prediction. With these information improvements for the model can be determined to improve the reasoning of the model. The reasoning of the model can be used as additional decision support to determine the accuracy of the prediction and whether or not to follow it [53, 24].

Using explanations to *inform the human decision-maker* is another motivation in the creation of explanations [24, 2]. This can give additional insights into the specific problem.

Human decision-makers without a background in machine learning are less likely to trust predictions of a complex model, even if the accuracy is high. Giving explanations to these users is often motivated by *increasing user trust* in the model to increase user acceptance [24, 30, 56].

The motivation for creating explanations gaining more interest in recent years is *assessing biases* [2, 26]. The data used to train models is gathered and created by humans, and human biases and prejudices can find its way into the models themselves [62, 30]. Explanations can give insight into the biases of the model.

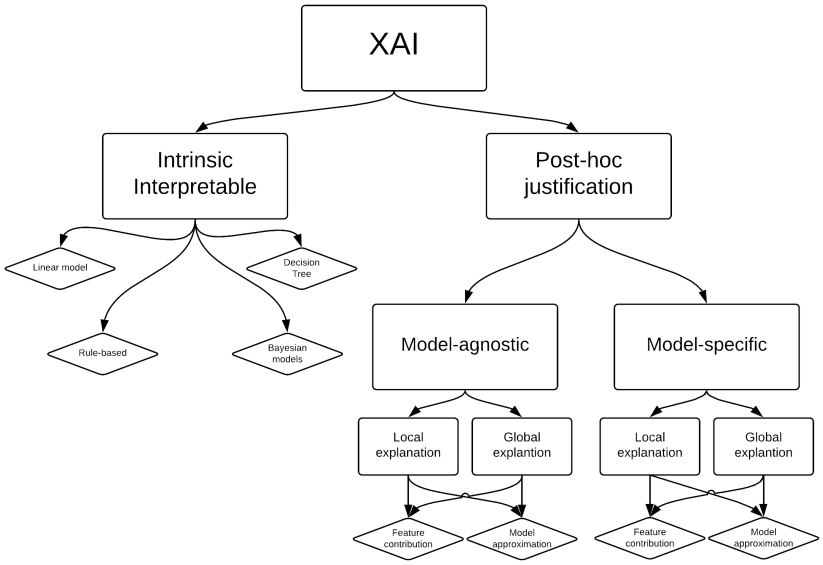


FIGURE 3.1: Overview of a selection of methods in the field of XAI

To explain complex models to people other than the machine learning experts, a survey by Biran et al. defined two main approaches; *prediction interpretation and justifications* or *interpretation of models* [10]. These definitions are the same in concept to transparency and post-hoc interpretability laid out in a survey by Lipton [53]. Only post-hoc techniques will be used in this thesis. However, as many of these techniques use interpretable models as an approximation, a brief summary of these techniques is given. Further distinctions between the approaches are model-specific or model-agnostic and local or global explanations. In the preliminaries of the research context a brief definition was given of these distinctions. In the following section, these distinctions are further explained and examples of these techniques are given. A partial overview of these different approaches is given in Figure 3.1.

3.3.1 Interpretable models

The concept of intrinsic interpretability of a model requires the design of the model to be easily interpretable [31, 53, 26]. The strictest definition laid out by Lipton [53] argues the model is interpretable if a person can contemplate the entire model at once. This requires the model to be sufficiently simple. A way a person can contemplate the entire model would be by taking all the input data and calculating the prediction by hand. This defines a model interpretable when it is simulatable. Another definition of an interpretable model is a model that "can be readily presented to the user with visual or textual artefacts" [73]. An example of such a model is the rigid rule-based decision systems discussed in the previous section. Other models generally assumed to be interpretable by design are linear models, decision trees, Bayesian Belief networks.

It should be noted that these models will not always result in intrinsically interpretable models [53]. The size of the model with high dimensional data may increase to not be reasonable to

perform inference (simulate). This definition of reasonable is subjective and may depend on the specific problem. High dimensional linear models or decision trees with a high number of leaves could for example be less interpretable than compact and shallow neural networks.

More complex, but still intrinsically interpretable, models were also proposed in recent years. An example is a feature selection and extraction approach using an easily interpretable logical formula for selection and grouping of features for the extraction [40].

3.3.2 Post-hoc justification

The concept of prediction justification or post-hoc interpretation tries to interpret the models in such a way it is understandable to the user. Here one tries to explain the complex model not by explaining how a model works; instead, we justify the predictions made. This justification requires a second model to provide the explanation of the existing model. A justification can for example be the determination of features which have the highest contribution to a certain prediction. There are model-specific and model-agnostic methods for the determination of these feature contributions proposed over the years [75, 7]. Other examples of these kinds of methods are *explanations by example*, *evidence as an explanation*, *text explanations* or *visual explanations*.

Explanation by example is one way to explain collaborative filtering techniques for recommendation systems [87]. This is most often in the form of "Because you like x, you also might like y" or "Similar users like this song".

Evidence as an explanation is for example used in text classification tasks. The proposed method by Lei et al. [49] is a good example of such an explanation technique. Here a small part of the overall text document classified is selected as being the most relevant in the prediction made and shown to the user as justification for the particular prediction.

A study by Krause et al. [44] proposed a visual explanation approach to explain to users how the system came to a certain decision. This was done, amongst other techniques, by showing the confidence (called uncertainty in the paper) of the system. One of the goals for showing this confidence is determining the weaknesses of the system for certain predictions. The user can then decide if the prediction is correct.

When looking at explanations for Random Forest models post-hoc approaches for the generation of explanations are most often used. Reason being that tree ensemble models are not easily interpretable due to the variance reduction through the aggregation of the intrinsic interpretable models. In the random forest model, this aggregation is the averaging of the different weak learners, removing the rule-based structure of the individual decision trees.

Model approximation

A popular post-hoc technique is approximating the complex model. To justify the predictions of a complex model, an intrinsically interpretable model is trained to approximate the results of the complex model. This means that the interpretable model is not trained on the ground truth of the training data. Instead, it tries to predict (approximate) the output of the complex model. The main assumption of this approach is that as long as the interpretable model is sufficiently close, the statistical properties of the complex model will be reflected in the interpretable model [26]. Evaluation of these techniques are therefore often the accuracy of the interpretable model in predicting the output of the complex model. However, a higher accuracy of the approximation often means a reduction in the interpretability of the simple model [6]. A distinction between model approximation techniques is global approaches and local approaches.

Global approximation For global approaches, the interpretable model tries to approximate the entire complex model. An approach for global approximation is for example a single decision

tree as the interpretable model approximation. The first use of a single decision tree as the interpretable model approximation was proposed by Craven [20], with other studies building on top of this approach [45, 12, 35]. These studies and approaches focus on the approximation of a neural network. However, it should be noted that these techniques only use the input and output of the neural network in order to approximate this network. This means these techniques can be used with any probabilistic model and can be seen as a model-agnostic approach. A recent example of a model-specific global approximation specifically for tree ensembles is inTrees by Deng [22]. The inTrees method works by extracting the rules that govern the splits in each of the weak learners. These rules are then processed by first removing the duplicates, measuring how long a rule is and pruning rules to their simplest form. The final step is combining these different rules into a simple set of if/then rules. These rules are easily interpretable and can be used as an approximate predictive model.

Local approximations Even when a global approximation focuses solely on achieving the highest accuracy in predicting the outcome of the complex model, this accuracy will never be 100%, and the complexity of the simple model will likely be high. Local approximation approaches were a response to this, with the assumptions that even complex models will show interpretable behaviour locally [73]. Locally in this sense means local in the feature space of a certain instance in the data. Instead of trying to capture the behaviour of the whole model, only the behaviour close by the instance is approximated.

One of the most popular implementations of this approach is called LIME by Ribeiro et al. [73]. The important part of the proposed method is the simple model does not try to approximate the complex model globally; instead, it approximates a model locally for a given instance. This makes the approximations locally more faithful to the complex model. The interpretable model used in the study is a linear classifier, which gives decision boundaries for features.

3.3.3 Feature contributions

With the approximation of complex models, we still do not have an explanation, only an approximate interpretable model. This does, however, make it possible to use explanation techniques of intrinsically interpretable models. One of the most popular approaches is showing feature contributions. In linear models, this is as simple as the weights assigned to a certain feature or the position in the tree in the case of a decision tree. Feature contribution determination for complex models does not require simple model approximation. There are global and local approaches for these determinations as well.

Global feature contribution determination for complex models are for example accuracy-based importance calculation for Random Forest models [13]. This approach keeps out-of-bag samples not used in the construction of the tree ensemble. The accuracy of these samples is calculated. This is followed by permuting the values of a single feature while keeping all other features values the same, after which the accuracy is determined again with this permutated feature. The main idea behind this permuting is that due to the randomness of the variable, the feature has no predictive power anymore. The feature permutation with the most impact on the accuracy gets the highest contribution score. This approach is model-agnostic, due to the fact that the determination is solely based on the accuracy of a complex model. There are also model-specific approaches for feature contribution determination. For example, importance determination based on the Gini impurity is specific to Random Forest models.

Local feature contribution approaches try to determine for a single prediction made, which feature contributed most for the specific prediction. Similarly to the local model approximation, the locality of instances is based on the location in the feature space. Based on this location, the behaviour of the complex model can differ, and cause the feature contribution to change. Similarly to the global permutation approaches, permutation or drop-out is used. However, now this is only

done on a single local instance or a couple of closest neighbours in the feature space. The drop-out approach instead removes a feature from these instances in order to determine the impact of this particular feature on the local prediction(s).

SHAP

A framework taking this approach even further is SHapley Additive exPlanations (SHAP) [56]. SHAP uses Shapley values for model feature influence scoring. Shapley values were originally proposed by Shapley in the field of game theory [78] and are a way for assigning payouts to players depending on their contribution to the total payout. The payout in this case is the prediction (probability), and the players are the different features of a given instance. The Shapley value is the average marginal contribution of a feature value across all possible coalitions. This means going through all possible features possibilities and look at the prediction made in order to determine the average feature influence. This approach is different from the basic drop-out or permutation approaches discussed before, as with this exhaustive approach all possible permutations or drop-out of features is calculated. In the basic approaches only a single feature is permuted or dropped out, with all other features remaining the same, losing possible interaction between features [85]. This exhausting approach guarantees consistency and local accuracy, something that is not the case for LIME [56]. However, a drawback is that SHAP is extremely slow for data with a large number of permutations. To improve efficiency, a number of approximations of the exhaustive approach were also proposed [56].

Random Forest specific feature contributions

With feature contributions being such a common way to explain predictions, a large number of methods to estimate these contributions have been proposed specifically for random forest models. A study by Palczewska et al. [67] detailed a three techniques to find patterns in the random forest's use of available features. The first technique is simply using the feature contribution median for each class. This is used as the standard level for each class. For a test sample, you look at the voting of the different trees and determine if the feature contribution is similar to the standard level of a certain class. The second clustering method looks at similar points and looks at the already known standard level feature contributions of these instances. The last method uses log-likelihood to cluster the point in a way to minimize the Euclidean distance.

Another study by Tolomei et al. [88] looked at determining the effort to change the prediction label for a certain instance in order to determine the feature contribution. This method goes through all the different paths of the ensemble and looks at how much a feature has to change in order to flip the decision made. While in the worst case computationally this is an NP-hard problem, in practice, it was found to be practical in the problem of determining low or high quality advertisements. This is achieved by limiting the search space through pre-computing a number of typical transformations likely to change the label of a given instance and by setting a limit on the number of permutations allowed for a given instance.

Prediction interpretation or justification by use of feature contributions is possible without any additional models when using a random forest model, as feature importance can be estimated with most implementations of the model in R [89] or Python [27]. These are global approaches towards the determination of which feature is most important for the complex model.

3.4 Evaluation of explanations

After defining explanations and interpretability, evaluation of these techniques is an important aspect. How do you evaluate an explanation's quality? Requirements of good explanations are often varied and differ across frameworks. Another difficulty is that these requirements often cannot be quantified directly. Together with the partially subjective nature of explanations make evaluating explanations difficult. Examples of requirements for good explanation are *truthful*, *transparent* and *understandable*.

A study by Doshi et al. [24] proposed a taxonomy for different approaches to the evaluation of explanations. In the taxonomy laid out three main approaches for evaluating interpretability: application-grounded, human-grounded, and functionally-grounded.

Application-grounded evaluation involves conducting human experiments within a real application [24]. This way of evaluating is particularly popular in the human-computer interaction and visualization communities. The main focus here is confirming that the system does indeed succeed in the task at hand. In other words, evaluation is done on the quality of an explanation in the context of its end-task. An example could be working with doctors on diagnosing patients with a particular disease. Here the whole system is built and tested with doctors. However, for the explanations of the complex models on real-world datasets, the evaluations are limited due to the difficulty of performing such an evaluation [1].

Human-grounded evaluation is about conducting simpler human-subject experiments that maintain the essence of the target application [24]. Here it is not necessary that the target community is participating in the evaluation. A larger pool of people can therefore be included in these kinds of evaluations. So instead of determining the quality of an explanation in a certain context, more concrete tasks are evaluated where the quality of the explanation can be inferred from this smaller task. An example would be a binary choice between two explanations where a user has to select the best explanation. A common approach is *forward simulation*, where the goal is to evaluate how well participants can prediction the output of a model [24, 46]. In such a simulation, the input and the explanation are given, while the participant is asked to predict the output. This is compared against the true output of the model to evaluate how interpretable the explanation makes the model. Finally, another common approach is the identification of (in)correct behaviour. In this case, the input, output and explanation are shown to a participant and is asked whether they agree with the prediction. An example of such an evaluation is given in the paper on LIME by Ribeiro [73].

Functionally-grounded evaluation requires no human experiments; instead, it uses some formal definition of interpretability as a proxy for explanation quality [24]. This is most appropriate for models already tested according to other evaluation techniques, such as the previous two in this taxonomy. The largest challenge is selecting the proxy which is most suitable. After this proxy is chosen, the problem can be best seen as an optimization problem. Examples laid out in the paper are improving the accuracy of the model. If the model is already found to be interpretable, improving the accuracy would improve the quality of explanations. Another example is evaluating the decision path length of a decision tree. A decision tree is deemed interpretable; therefore, evaluating an explanation of the decision tree can be determining the length of the decision path and finding a way to reduce this length.

3.4.1 Evaluation metrics as proxy

Given that most of the high-level requirements of explanation cannot be measured directly, simpler evaluation metrics are used as proxies. An improvement in the proxy is assumed to also improve on the high-level requirements of explanations. These proxy metrics can be measured quantitatively.

For this review, two groups of metrics are discussed; *objective metrics* and *subjective metrics*.

Objective metrics Objective metrics are all measures not dependent on perceived or subjective measures. Examples of these metrics are *task effectiveness* or *task efficiency*. *Task effectiveness* is a measure of accuracy in performing a particular task. Most often it measures if an explanation helps in increasing the accuracy of human-decision task. In the case of inland ship inspector, the task is deciding whether or not to inspect a ship based on the likelihood of a violation. Correctly deciding to inspect a ship where a violation is present can be measured objectively. An example of evaluating the accuracy of the human expert's decision was performed by McGuirl and Sarter [60].

Task efficiency determines whether an explanation helps with performing the task more efficient. Looking at the time it takes to perform the task with the explanation is a common measure of this metric.

An example of such an evaluation is by Schmidt & Beissmann [76]. Information transfer rate is used as an evaluation metric as a proxy for understandability of the model. The crowd-sourced experiment asks participants to replicate model predictions and measures the time it takes to complete a task. The specific task in this experiment was replicating the prediction made by a machine learning model. Empirical evidence determined that the proposed metric robustly differentiates between interpretability of the different models.

Subjective metrics Subjective metrics are based on the opinion of the user or participant of the experiment. A common evaluation technique, both for human-grounded evaluation as well as application-grounded evaluation, is reviewing the explanation with end-users subjective feedback. When looking at subjective markings such as understandability of an explanation, proxies have been proposed to measure the interpretability instead [47, 71]. Other studies looked at evaluating the quality of explanations with the proxy of user self-reported trust in the explanation [69].

3.4.2 Human-grounded evaluation of SHAP

With the popularity of SHAP and the newly defined taxonomy for the evaluation of explanation techniques, Weerts et al. defined a human-grounded evaluation for SHAP explanations for Alert Processing [95]. In the evaluation users were asked to perform simplified alert processing tasks, with and without the explanations generated by SHAP. The evaluation metrics defined were task effectiveness, task efficiency and mental efficiency. There were two experimental designs for the evaluation of explanations given.

For both experiments, there was no significant improvements found in the three metrics when comparing against giving no explanation. Based on a survey held after participating in the experiments, it was found that the explanation did change the reasoning applied by our participants. The leading source of evidence was the model's confidence score in both experiments.

3.5 Research gaps

Conformal Prediction for ranking Based on the literature discussed a number of research gaps are found, bridging the gaps of using the Conformal Prediction framework for traditional machine learning problems and the information retrieval problem of ranking.

The Conformal Prediction framework is most often evaluated in terms of *efficiency* while still achieving the guaranteed error-rate. This means reducing the size of the prediction sets as much as possible. However, evaluating the relative performance of the informative predictions sets for binary classification only is, to our knowledge, not researched. Increase in relative performance compared to the global performance of the system indicates how the model confidence can be used to select and therefore rank instances.

The notion of confidence expressed by conformity is not yet used to rank predictions of traditional machine learning methods. In general, the ranking of predictions of classification models is most often sorting on the probability.

The notion of confidence *is* used for a number of information retrieval ranking problems. However, these approaches, such as collaborative filtering, are not used for traditional classification problems. The structure of the datasets for these specific problems most often differ from those used in more traditional classification methods, such as the ones used in this thesis. A goal for this thesis is therefore a novel ranking approach incorporating confidence which is compatible with any probabilistic classification problem.

Confidence displays A number of studies laid out in this review evaluated the use of confidence displays. These confidence displays are either based on basic metrics to express confidence separately from probability or even uses probability as confidence. The use of conformity to express confidence in an explanation is not yet researched. Furthermore, the confidence displays in the studies discussed do not explain or justify this confidence. Instead, only the impact of showing the score or value itself is evaluated.

Explaining of confidence As discussed in Section 3.3, giving the probability as an explanation by itself is not enough. Therefore, a large number of explanation methods are proposed, with a few of those laid out in this chapter. Particularly, we looked at techniques justifying the probability by determining the features contributing most towards this probability. The most contributing features can be seen as evidence supporting the prediction. To our knowledge, these techniques are not yet used in the context of confidence. This is the case for both confidence expressed by the variation as well as confidence expressed by conformity. Simplified; the explanation methods currently are used to explain the mean of the probability distribution and are not looking at explaining the range or the variation of the prediction.

Chapter 4

Predicting model confidence

In this section we define model confidence and the methodology for determining this model confidence to answer the first research question:

How do you predict the confidence score separately from class probability?

The definition of model confidence is determined in Section 4.1, together with an example showing the intuition for this metric. In Section 4.2 the working of the Conformal Prediction framework is described, together with which features were used in this thesis. This includes the basics such as the non-conformity function and prediction sets, together with the desirable properties of the confidence determined.

Finally, we perform two experiments to determine the behaviour of confidence in relation to the accuracy of the conformal prediction. The main contribution of the first experiment is determining if using the confidence in a prediction can help in selecting instances with higher accuracy. If this is the case, it could help in the ranking of instances as well. By selecting based on the separation of the p-values by the significance level, only high confident instances are tested. Another contribution is evaluating if the model confidence determined is reliable for the real-world problem of violations on *inland* ships. This is achieved by checking if the guaranteed error-rate is met.

In a second experiment, the meta-conformity approach is evaluated. The contribution of this experiment is determining whether the predictions by the Conformal Prediction outperform the predictions of the base model.

4.1 Definition of model confidence

Before going into the methods used to define model confidence, it is useful first to describe the idea behind this metric and why it was chosen to look at in this research. As discussed in the literature review in Section 3.1, this concept of confidence is defined in numerous ways. In this research, model confidence tries to answer the following question:

How confident is the model in its output?

In the case of probabilistic models, this output would be the probability itself. The question we try to answer is *"How confident is the model in its probability?"*. This can, therefore, be seen as an additional quality measure for a single prediction by the model.

To illustrate the core idea behind this approach, let us imagine a simple Logistic Regression model trying to solve a binary classification problem. This simple model is shown in Figure 4.1. In this example each instance in the data consists of only two features to easily represent them in 2D space. A simple Logistic Regression model determines probabilities for each instance with the sigmoid function. The function maps any real value into another value between 0 and 1. In order

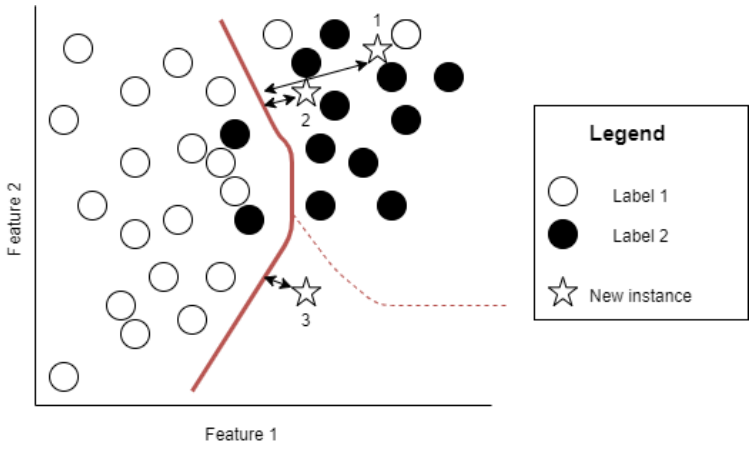


FIGURE 4.1: Simplified model

to classify the instances, the model creates a decision boundary D to maximize the accuracy on the training data consisting of only two labels. This results in the following; for a given instance p_i, the probability is determined by $d(D, p_i)$. This means that if an instance is further away from the decision boundary, the probability for the predicted label is higher.

In the example shown in Figure 4.1 there are also three new instances shown which we want to classify. If we look at the probability as a function of $d(D, p_i)$, we find that the probability of the new instance 1 is higher than the probability of instance 2. Even though in the training data an incorrectly labelled instance is similar to instance 1. The new instance 2 only has the same labelled training instances in its neighbourhood. However, it got a lower probability due to being close to the decision boundary. Another problem can be seen when looking at the third instance; this instance is assigned label 2. There are however no representative instances in the training data, so none of the nearest neighbours are of label 2. This means that the decision boundary in this space could be incorrect. To illustrate; the dotted line is also a valid decision boundary with the same accuracy as D. This alternative decision boundary is just as valid.

This indicates that only looking at $d(D, p_i)$, which is based on the probabilities of a given instance, does not mean that we can be confident that the classification is correct. In this research we not only look at the probability of a given instance, but also define the confidence of the model for that given instance.

For the determination of this confidence, most research summarized in Section 3.1 looks at the relationship between the instance which has to be classified and the instances in the training data. When looking at prediction intervals in this simplified example, this approach looks at the effect of permuting the training data. These permutations have obviously an effect on the decision boundary in this example, and therefore the probability of a given test instance. If this probability changes drastically when slightly changing the training data, the model confidence is low for that given instance. This approach therefore looks at solving the problem of the third instance in Figure 4.1, where a different decision boundary in a part of the feature space low on instances could cause incorrect determination of probability.

When looking at the confidence of the model for the given instances in this example, useful additional information is the similarity of test instances and the instances in the training data. In concept, this is how the Conformal Prediction framework determines the confidence in a certain prediction. However, this framework looks in the probability space of a given model, instead of the feature space shown in this simplified example. More details as to how this works are described in Section 4.2. To determine the confidence of a certain classification one can look at the three nearest neighbours for the three instances in the example of the simple Logistic Regression model, as shown in Figure 4.2

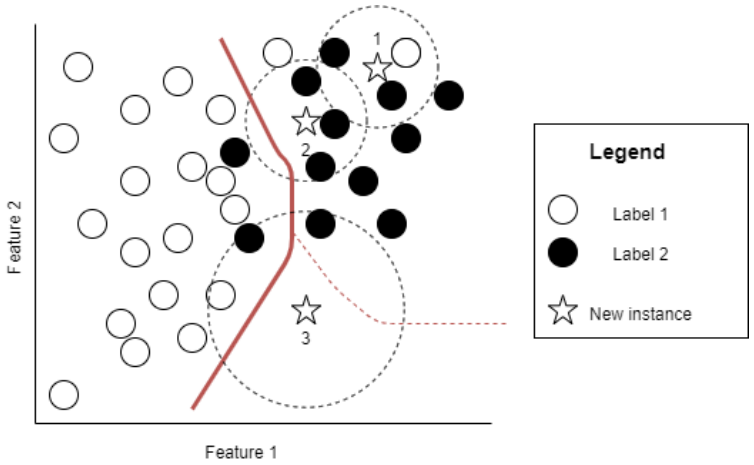


FIGURE 4.2: Simplified model with conformal prediction

For the new instance 1 in Figure 4.2, one of the nearest neighbours is classified differently than the instance itself, and the distance to the neighbours is not large. This could lead to a decrease in the confidence of the model for this given instance. For the second instance the neighbours are all classified the same, and the distance between the new instance and its neighbours is small. So while the probability is low for the instance, the model can be confident that this probability is determined correctly. The third new instance has the nearest neighbours the farthest away, and one neighbour is classified differently. Although the probability for instance 2 and 3 is close, as $d(D, p_2)$ & $d(D, p_3)$ are similar, the confidence in these predictions is low when looking at the nearest neighbors of an instance.

Probability is purely determined by the model's configuration and the features of an instance that has to be classified. To determine the model confidence of a prediction, additionally, the instances in the training data are used.

4.2 Conformal Prediction Framework

There are many machine learning algorithms proposed in order to deal with high-dimensional data problems. These algorithms perform well in a number of different situations when looking at measures such as accuracy, false positives, false negatives or the area under the receiver operating characteristic curve. Furthermore, these algorithms do not require any parametric statistical assumption about the training data. However, in traditional parametric statistics, a well-studied area is confidence estimation. The Conformal Prediction framework tries to bring this research and developments in the field of parametric statistics to the field of machine learning.

The central concept behind conformal predictors is as follows; for a new instance the "strangeness" to the training data is determined with the use of a (non-)conformity function. The goal of this measure is to determine how well this new instance conforms to the training data. This measure is then converted into p-values, which allows us to make not only predictions, but also estimate the confidence in a prediction.

A traditional example of a use case for this confidence is in the medical field. Here it is important to measure the risk of a misclassification (meaning misdiagnosis) and allow only a low risk of error.

Based on this short and simplified introduction one may think the conformal framework is just a k-nearest neighbour algorithm; however, there are significant differences. In this section, a summary is given of the workings of the conformal framework as well as what part of the framework is used in this research.

4.2.1 Assumptions

In this section the basics of the Conformal Prediction Framework are described. Let's assume a certain setting where there is a training dataset and we want to predict a new instance *in the same instance space I*. This results in a training set of i_1,...,$i_n \in I$ with a test instance of i_{n+1} or t. In this setting, the training set is not a set but a sequence. In the Conformal Prediction Framework there are only a few assumptions. The first assumption is the randomness assumption, meaning that it is assumed that all instances (i_1,...,$i_n + 1$) are from the same independent probability distribution P on I. With this first assumption, the second assumption follows naturally; as the instances are taken from the same probability distribution, any permutation of the training sequence is exchangeable with the original training sequence. These assumptions can mean the instances in the training data as well as the new instance are independent and identically distributed (i.i.d. assumption). These assumptions are not unique to the Conformal Prediction framework and are used in most machine learning algorithms.

4.2.2 Prediction sets

Another aspect of the Conformal Prediction Framework is the use of *prediction sets*. This means that the output for a single instance is not a single class, instead, it returns a set of classes as its prediction. To illustrate, lets assume a machine learning model trying to classify with 5 possible classes $c1, ..., c5 \in C$. The traditional model will return one of these classes based on the $arg\,max\, P(c_i)$, where $P(c_i)$ returns the probability for c_i. The prediction set in the Conformal Prediction Framework will return all classes above a threshold based on a certain significance level $\epsilon \in [0, 1]$ as a set. The determination of this ϵ is a compromise between validity and efficiency of the overall predictor. If you set ϵ to 0, all classes are returned in the set. This results in the prediction always being *valid*, as the correct class will always be contained in the set. This is however a trivial solutions and is non-informative. *Efficiency* determines if the returned set is as small as possible while still guaranteeing a given error-rate. This guarantee has to be imposed to not only return the trivial case of $\emptyset$. This trivial case is always returned when $\epsilon = 1$. The set of predictions is also called the prediction region where, similarly to confidence interval techniques discussed in Section 3.1.1, the region of possible outcomes is determined with a certain guaranteed probability.

4.2.3 Conformal predictions

As mentioned in the introduction of this section, this framework works with the notion of "strangeness" or (non)conformity. This (non)conformity measure is a function $A(I, t)$ mapping any sequence of instances $i_1, ..., i_n$ to another sequence of real numbers $\alpha_1^t, ..., \alpha_n^t$ that is the same for all possible permutations $\phi()$ of the original sequence. The instances $i_1, ..., i_n$ as well as t can be any object; in the case of classification it is often represented as $((X_1, y_1), ..., (X_n, y_n))$ and (X_{n+1}, y_{n+1}) respectively, where X represents the feature values and y the label of an instance.

Here a_1^t represents how well the test instance t conforms to i_1.

$$A((i_1, ..., i_n), t) = (\alpha_1^t, ..., \alpha_{n+1}^t) = A((i_{\phi(1)}, ..., i_{\phi(n)}), i_{\phi(n+1)}) = (\alpha_{\phi(1)}^t, ..., \alpha_{\phi(n+1)}^t)$$

With these conformity values p-values for t can be determined as follows:

$$p^t = \frac{|\{i = 1, ..., n+1 | \alpha_i^t \geq \alpha_{n+1}^t\}|}{n+1}$$

The p-values represent the proportion of a_i's which are at least as large as the α_{n+1} representing the test instance. This results in a value between $1/(n+1)$ and 1. A p-value of $1/(n+1)$ is achieved when α_{n+1} is the largest value. This means that the instance t is very non-conforming. While a p-value closer to 1 means conforming instances.

TABLE 4.1: Summary of the different notation throughout this section

Symbol	Definition	Description
$i_1 = (x_1, y_1) \in I$	Data instance	An individual instance of f features in instance space I. $I \setminus t$ are the training instances.
$P(i_1) = P(y_1 \| x_1)$	Probability	The output of the base model. Expression of probability of y_1 given X_1.
$A(I \setminus t, t)$	Conformity of individual instance	Mapping the conformity between $\forall i \in I \setminus t$ and t
α_r^t	Conformity between two instances	Expression of the conformity between the test instance t and train instance r
p^t	p-value for test instance t	Expressing the $A(I \setminus t, t)$ as a single value for t
$\Gamma^\epsilon(I)$	Prediction set	Output of the Conformal Predictor. Returns all X_t and $y \in Y$ combinations with p^t above a certain ϵ
$conf((x_t, y_t^0)) = 1 - p_t^1$	Confidence of conformal prediction	The confidence in the prediction $y = 0$. A set of predictions with 95% confidence will at least be 95% accurate.
$cred((x_t, y_t^0)) = p_t^0$	Credibility of conformal prediction	The credibility in the prediction $y = 0$ for t.

The conformal predictor with significance level ϵ be Γ^ϵ uses these p-values to make a prediction:

$$\Gamma^\epsilon(I) = \{t | p^t > \epsilon\}$$

In words, the conformal predictor returns at the significance level ϵ the set of possible predictions. With the definition of Γ^ϵ with $p^t > \epsilon$ it tries to determine conformity, but using $\leq$ would result in a non-conformity function. In the Conformal Prediction framework a measure of non-conformity is most often used instead of a conformity measure. This is because of A being easily expressed as a distance function, similarly to the nearest neighbour example in Section 4.1. Such a distance measures non-conformity, where the higher distance increases the strangeness or nonconformity.

Hypothesis testing

This conformal predictor performs a similar task as Hypothesis testing. Hypothesis testing is proof by contradiction and starts with the assumption that a Hypothesis H_0 is true. H_0 in conformal prediction is most often that for the data sequence $I \cup t$ the randomness assumption still holds. The alternative hypothesis H_a is that $I \cup t$ is not random. In other words, H_0 is the case that test instance t conforms to the training data, while H_a is the case that t does not conform.

In simple notation this is expressed as follows:

- The test example t is assigned a possible label y_{n+1}: (X_{n+1}, y_{n+1})
- Hypothesis Test:
    1. H_0: The data sequence $I \cup t$ is generated independently from the same distribution and is therefore random.

2. H_a: The data sequence $I \cup t$ is not random.

Where the decision is based on the p-values is done as follows:

- For a significance level ϵ:
  1. Reject H_0 if $p_t \leq \epsilon$
  2. Do not reject H_0 if $p_t > \epsilon$

The p-value is the probability, assuming that H_0 is true, of obtaining another p-value in the test statistic at least as contradictory to H_0 as the value calculated from the available training data. This is distinctly different than the probability of a certain label or H_0 being true. Furthermore, the p-values for all possible labels do not need to sum to 1. This hypothesis testing is done for all possible assignments of labels to t and results in p-values for each label for each instance.

With this principled approach to obtaining predictions the confidence and credibility of a certain prediction can be clearly defined with guarantees in accuracy. The confidence tries to express how often a prediction is correct, instead of predicting the individual probability of an instance. With a set of predictions with at least 95% confidence, this measure guarantees that 95% of the predictions are correct, even if we cannot assert a full-fledged 95% probability for each prediction when we make it. To avoid overconfidence for the instances that are unusual, *credibility* is defined as the largest ϵ for which the prediction set is empty ($\Gamma^{\epsilon}(I) = \emptyset$). This overconfidence occur when for a single instance all p-values are low; while having a high confidence, due to the low p-value of the class not predicted, the conformity to the class predicted is also low.

In the case of a binary classification problem these additional metrics are determined as follows:

- Determine p^0_{n+1} and p^1_{n+1} for the possible labels [0,1] for X_{n+1} of test instance t
- If $p^0_{n+1} < p^1_{n+1}$, predict label 1 with the confidence $1 - p^0_{n+1}$ and credibility p^1_{n+1}
- Else predict label 0 with the confidence $1 - p^1_{n+1}$ and credibility p^0_{n+1}

For classification with more than two classes the predicted label is the assignment of label and instance which results in the highest p-value. The credibility is this p-value and the confidence in the prediction is 1 minus the second largest p-value. The ideal case is $max(p^0, p^1) \approx 1$ and $min(p^0, p^1) \approx 0$. This results in both a high credibility and confidence for the given instance. An instance has a low credibility, meaning all p-values are low, implies that the training data is not random (biased) or the instance is not representative of the training set [90].

To conclude these last few sections; the Conformal Prediction framework looks at determining (non-)conformity between test instances and the training sequence. It returns prediction sets with a certain significance level. This set is determined with hypothesis testing based on p-values expressing the (non-)conformity of a certain instance and label assignment combination. These p-values can be used to predict in a classification problem and obtain valid prediction regions, confidence and credibility. To make this even more clear we provide an example of this overall approach:

- For test sample (X, y) with 5 possible labels the following p-values are determined: $p_{y=0} = 0.2, p_{y=1} = 0.1, p_{y=2} = 0.8, p_{y=3} = 0.6, p_{y=4} = 0.9$.
- $\Gamma^{0.85} = \{4\}$, with a confidence of 0.2
- $\Gamma^{0.75} = \{4, 2\}$, with a confidence of 0.4

- $\Gamma^{0.55} = \{4, 2, 3\}$, with a confidence of 0.8
- $\Gamma^{0.15} = \{4, 2, 3, 0\}$, with a confidence of 0.9

This example shows the compromise between *validity* and *efficiency*. The most *efficient* and *non-trivial* set is $\{4\}$. However, due to the large p-value of $y = 2$ the confidence, representing the *validity*, in the prediction is low. If you want to be 80% confident the correct label is contained in the prediction set it makes three outcomes possible: $\{4, 2, 3\}$. This larger *validity* comes at the cost of a larger prediction set (less *efficient*).

4.2.4 Non-conformity function

The most important aspect of the Conformal Prediction framework is the determination of the non-conformity function A defined in Section 4.2.3. This function determines the p-values, which are the essential parts for making a prediction as well as determine the prediction set, credibility and confidence.

This determination can be any function expressing the strangeness or nonconformity of a given sample to the training data. There are model-agnostic functions as well as model-specific functions possible. The model-agnostic approaches work on any machine learning or data mining algorithm working with probability. In the case of classification model-agnostic functions looks only at the probability outputs $P()$ of the model and the original data as input to determine α_i. Three examples of such functions are:

- Inverse probability: $\alpha_i = 1 - P(y_i|x_i)$
- Margin: $\alpha_i = 0.5 - \frac{P(y_i|x_i) - max_{y_j \neq y_i} P(y_j|x_i)}{2}$
- Nearest neighbours: $\alpha_i = \frac{min_{j \neq i \& y_i = y_j} d(x_i, x_j)}{min_{j \neq i \& y_i \neq y_j} d(x_i, x_j)}$, where d is the Euclidean distance

Here the first two functions are based on the output of the underlying model while the third example looks at the original input data. This third example is the one shown in Section 4.1.

The model-specific functions use additional specific information from the underlying algorithm to determine α_i. An example would be the non-conformity function of TCM-RF [92]. This function uses the notion of identical paths in the forest in order to determine non-conformity:

$$\alpha_i = \sum_{j=1}^{K} prox_{ij}^{\neg y_i} / \sum_{j=1}^{K} prox_{ij}^{y_i}$$

Here $prox(i, j)$ represents the percentage of trees having identical paths for sample i and j. K expresses the number of nearest neighbours to incorporate for determining the non-conformity score.

4.2.5 Transductive versus Inductive

There are a large number of approaches developed for this conformity framework–methods like bagging of conformal predictors or bootstrapping [68][52]. The largest divide between different approaches is however if the predictor is transductive or inductive. All predictors discussed up to this point have been of the transductive type, where there is no general hypothesis about unseen data. Instead, we generate a new hypothesis based on all training data for the new instance. This makes the algorithms extremely inefficient with large datasets as for each test instance and label combination the p-values for the whole dataset has to be determined. For this problem an

inductive conformal predictor has been proposed [90]. In this method the training data of size n is split into a *proper training set* of size t and a *calibration set* of size c, where $c = n - t$.

The classification rule is then determined with the proper training set using the underlying machine learning algorithm once. For each new instance only $\alpha_{t+1}, ..., \alpha_{t+c}$ are determined and used in determining the approximate p-values. If the dataset is truly random, this approximation is the exact p-value [77].

4.3 Experimental design

For the evaluation of the Conformal Prediction framework, specifically model confidence, the existing Python library *nonconformist* version 2.1.0 is used [51]. The advantage of this library is the compatibility with the popular *scikit-learn* library [70]. All the different machine learning models inside the *scikit-learn* library are therefore compatible. Version 0.23 of *scikit-learn* was used during these experiments.

The *nonconformist* library allows for both transductive and inductive predictors, as well as more complex conformal predictors like bootstrapped predictors. More importantly, the library allows defining any non-conformity function. Model-agnostic non-conformity functions can be defined with a standard format of parameters, allowing the conformal predictor to work on any classifier. Model-specific functions require custom parameters passing to provide additional model-specific data to the function. For all evaluations of the Conformal Prediction framework the model-agnostic margin error non-conformity function was chosen, due to the performance of this function compared to other model-agnostic functions [36].

For the different experiments in this chapter a number of the classifiers are used. The focus in this chapter will be on tree ensembles techniques, namely Random Forest and XGBoost. These classifiers are chosen due to their performance on the specific problem of violation on *inland* ships. Other simpler models used during evaluation are *Logistic Regression*, *Quadratic Discriminant Analysis*, *k-nearest neighbours* and *Naive Bayes*. Additional results with the other datasets and classifiers are presented in Appendix A.

First, the behaviour of the conformal predictor across significance levels is determined. Secondly, a short experiment is performed to determine the performance of the conformal prediction, which is compared to the base model's prediction.

4.3.1 Experiment 1: prediction set across significance levels

In this experiment the behaviour of the Conformal Prediction framework is evaluated when looking at different significance levels. Different types of prediction sets, with different confidence levels, are compared to determine the impact of the confidence measure on the accuracy across significance levels. The datasets in this experiment are used for binary classification problems.

The goal of this experiment is twofold; firstly, confirming the guaranteed error-rate of the Conformal Prediction Framework. Meeting the guarantee on the different datasets in this study confirms the i.i.d assumption of the different datasets. Secondly, a selection based on instances with higher confidence is made with the use of the significance level. Measuring the accuracy can indicate the usefulness of model confidence in selecting predictions. This experiment, in combination with Section 4.2, answers the first research question of this chapter:

How do you predict the confidence score separately from class probability?

The prediction is made with the Conformal Prediction framework and evaluated by selecting based on the significance level separating the p-values in the binary problems. To our knowledge, evaluating this subset of instances in the test set is not yet performed. It can indicate if selecting

based on confident predictions result in higher accuracy compared to the global performance of the model.

When looking at binary classification problems, only four possible prediction sets can be determined by the Conformal Prediction framework. The set either contains neither class, it contains only one class or it contains both classes, resulting in four possible predictions sets.

$$\emptyset, \{0\}, \{1\}, \{0, 1\}$$

The sets containing no class or both classes are *non-informative* as these are the trivial cases discussed in Section 4.2.2. Classification in these *non-informative* situations is still possible and is determined by the largest p-value, while one minus the other p-value determines the confidence. However, this removes the guarantee of *validity*. The other two cases of only containing a single class in the prediction set are informative. The difference between the trivial cases is the p-values for a given instance of the two classes are separated by the significance level and are likely further apart in general. Informative *efficiency* is achieved in the two cases of a set containing only a single class [34]. The Conformal Prediction framework has guarantees about validity. Specifically, at significance level ϵ, the probability of the true class label not being contained in the prediction set is 1-ϵ [77]. In other words, the error rate is bound. This is only guaranteed if the i.d.d. assumption described in Section 4.2.1 holds.

As the i.d.d. assumption is not easily tested for the inland ship dataset, the validity of the results are evaluated. This evaluation of the conformal prediction sets is inspired by the evaluation performed by Johansson et al. [34].

In a cross-validation setting the accuracy of the conformal prediction is evaluated. The accuracy is determined only for the instances where there is either a single class prediction set or a prediction set with both classes. In the case of a prediction set containing both classes it as always correct and the y_{true} is therefore the same as y_{pred} in this evaluation. For the single class instances the error rate is determined to get to an overall accuracy score of the conformal predictor at different significance levels. The evaluation is described in more detail in Algorithm 1

Algorithm 1 Evaluating prediction sets

$x \leftarrow$ dataset containing instances $x_1, ..., x_n$
$y \leftarrow$ labels for x containing $y_1, ..., y_n$
$cv \leftarrow k$-fold cross-validation
$S \leftarrow$ set of significance levels from 0 to 1 with step-size 0.01
for all $f_i \in cv$ **do**
  for all $\epsilon \in S$ **do**
    split (x, y) in training and test set.
    clf.fit(x_{train},y_{train})
    $res \leftarrow icp$.predict(x_{test},y_{test})
    icp.fit(clf)
    $set \leftarrow icp$.predict(x_{test},y_{test}) at significance ϵ
    drop all $\emptyset$ from set
    calculate accuracy for set with res

In this evaluation the margin probability function is used as a non-conformity measure. The reason for choosing this function is the model-agnostic property of this particular function. The performance can therefore be tested on any datasets and classifiers. Inductive Conformal Prediction is used with 20% of the training data being used as the calibration set. This is to speed up the evaluation steps across the different datasets.

Besides testing the validity of the resulting prediction sets, this experiment additionally evaluates the relative performance of the informative prediction set over the overall performance of the model. In binary classification problems this is particularly interesting, as there does not exist a prediction set larger than one which is still somewhat informative. By evaluating the accuracy of the subset, the usefulness of confidence in selecting instances is tested. This approach is unique, as traditionally confidence is only used with the inclusion of the non-informative prediction set to achieve the guaranteed error rate.

4.3.2 Experiment 2: Meta conformity approach

Before evaluating the additional confidence information in a ranking problem, an important step is determining which classification to ultimately use. There are two possibilities; the prediction of the original machine learning model or the prediction of the conformal predictor. This separation of classifying using the original model and confidence determination using the conformal framework was proposed by Smirnov [82][81].

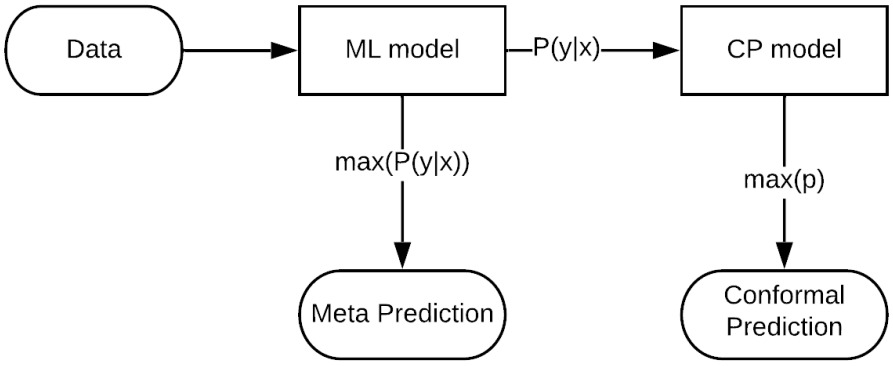


FIGURE 4.3: Two possible predictions compared in meta approach

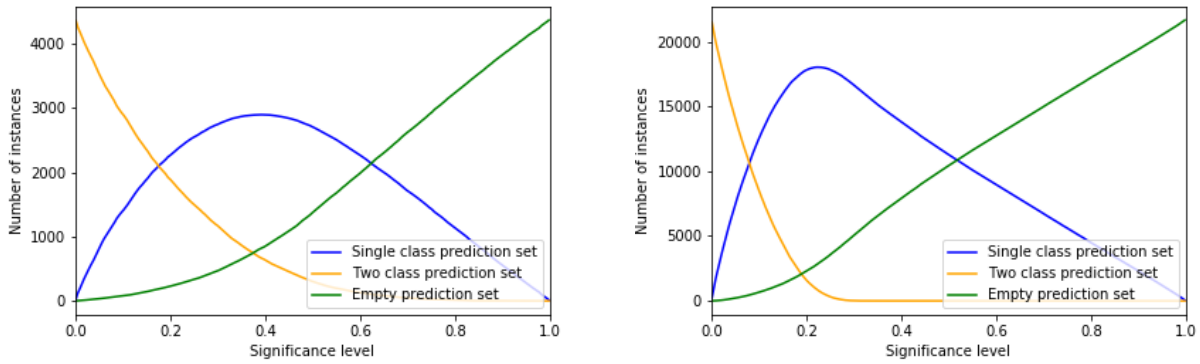
For the prediction of the original machine learning model, the most probable class is chosen as label for a given instance. The Conformal Prediction framework takes the probabilities and determines the p-values based on the inverse probability function. The highest p-value determines the label when predicting with the Conformal Prediction framework. In this experiment a short evaluation comparing the meta-approach and the conformal prediction (CP) for the particular datasets and classifiers is performed. The area under the Receiver Operating Characteristic curve is used as an evaluation metric. In order to determine if the difference between the two approaches is significant, an independent Student's t-test is used.

4.4 Results

In this section we describe the results of the two experiments in this chapter. The experiments determine the behaviour of the conformal prediction across significance levels and the performance of the two prediction approaches.

4.4.1 Prediction sets on binary classification

When looking at the type of prediction sets across significance levels, the behaviour is generally the same across all datasets and classifiers. At significance level 0, all prediction sets contain both classes. These trivial cases start decreasing when increasing the significance level. The same occurs for the trivial efficient case of the empty prediction set close to significance level 1. Here these instances increase when approaching this significance level. The number of informative prediction sets, the ones only containing a single class, are zero at significance level 0 and 1. The number of these informative prediction sets across significance levels represents a bell curve. These prediction sets across significance levels are expected behaviour; prediction sets with two classes mean that both p-values are higher than the significance level, which is more likely with lower significance levels. With the empty prediction set, both p-values are below the significance level, which is more likely at the higher significance levels. The informative prediction sets means a single p-value is above the significance level, and the other below it. This results in the bell curve with a certain maximum. In Figure 4.4 this behaviour is shown for two combinations of classifier and dataset.



(A) Random Forest and *inland ship* dataset

(B) XGBoost and *churn* dataset

FIGURE 4.4: Number of prediction sets of a certain type across significance levels

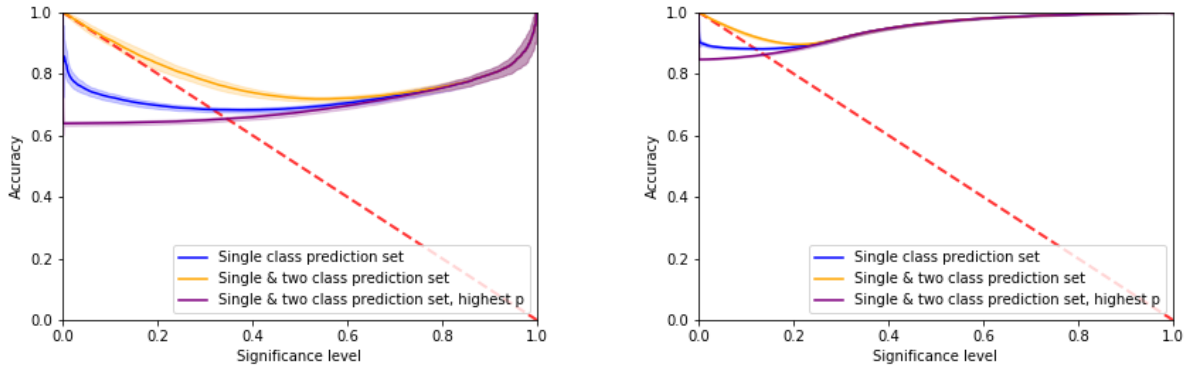
With these different prediction sets across significance levels, the accuracy is also plotted. As mentioned before, the error rate is bound with the Conformal Prediction framework. The guarantee of error-rate or accuracy can be evaluated by determining the accuracy of the instances containing only a single class in the prediction set together with all instances with a prediction set containing both classes. For these latter instances the prediction is always correct, as the correct class is always contained within the prediction set. This situation is plotted as the orange line in Figure 4.5, with the red dotted line representing the bound of the error rate. From this it can be seen that for these two combinations the guarantee of validity is met. This is the common evaluation of the Conformal Prediction framework to determine if the guarantees are met.

For this experiment, additionally, the accuracy of predicting based on the highest p-value is plotted. This evaluates the global accuracy of the conformal predictions.

The accuracy is no longer guaranteed in this case. In Figure 4.5, it can be seen that the accuracy drops significantly for all significance levels where two class prediction sets occur. These two class prediction sets mostly occur for lower significance levels. This is expected, as in the original situation these prediction sets were always deemed correct, resulting in 100% accuracy for these instances. By selecting the class with the highest p-value in these prediction sets, incorrect classifications are possible. This results in the accuracy close to significance level 0 becoming the accuracy of the base model. A selection of instances in the test set is made when moving towards a significance level of 1, excluding based on the credibility of the prediction.

Uniquely, the usefulness of confidence in selecting instances is tested by evaluating the performance of the informative prediction sets across significance levels.

As with the previous determination of accuracy based on the highest p-value, this only impacts the performance when looking at significance levels where two class prediction sets occur. The difference is that in this case the instances with two class sets are excluded. These single class instances (informative sets) are the predictions with higher confidence compared to these excluded instances. In the case of the two class prediction set both p-values are larger than the significance level. The confidence of a prediction is 1 minus the second largest p-value. Together with the single class instances having a p-value below the significance level, this means a higher confidence. The resulting accuracy for only these instances with higher confidence is improved when looking at the overall accuracy of the conformal prediction.



(A) Random Forest and ship dataset

(B) XGBoost and churn dataset

FIGURE 4.5: Accuracy across significance levels

An observation from the resulting accuracy is that the guaranteed performance of the Conformal Prediction framework is not always met when looking at the inland ship database. This guarantee is only met when the accuracy of both the single and two class predictions is above the red dotted line representing the error rate. This is most likely caused by the noise in the data, as well as the i.i.d. assumption not always being met. For example, a ship can be inspected several times, resulting in dependency between these instances. Another reason for the i.i.d. assumption not holding could be the fact that an owner can have multiple ships, also creating a dependency between the instances. For the other 'clean' machine learning datasets, the guarantee is met [77]. All resulting plots of the different combinations of classifier and dataset are contained in Appendix A.1.

To conclude, this section looked at the prediction sets behaviour across different significance levels. The guaranteed error-rate was largely met when looking at the inland ship dataset and the Random Forest base model. It also showed the increase in accuracy when looking at predictions with a high confidence compared to all predictions based on the highest p-value.

4.4.2 Meta-Conformal classification

As mentioned in the experimental design, a short evaluation comparing the meta-approach and the conformal prediction (CP) for the particular datasets and classifiers is performed. The area under the Receiver Operating Characteristic curve (AUC) is used as an evaluation metric. In order to determine if the difference between the two approaches is significant, an independent Student's t-test was performed. The resulting metrics are shown in Table 4.2.

TABLE 4.2: Comparing the performance of the conformal prediction and the base model

Dataset	Classifier	AUC CP	AUC Meta	t-value	p-value
ship	knn	0.5472	0.5585	4.1313	**0.000117339005012**
ship	rf	0.6605	0.7016	19.2769	**6.75088607247962E-27**
ship	xgb	0.6650	0.7133	20.2868	**5.08739016174619E-28**
ship	lr	0.5680	0.5531	-4.0563	**0.000150736137687**
ship	qda	0.5499	0.5500	0.0156	0.987605778544881
ship	nb	0.4918	0.5338	7.0393	**2.51763120616029E-09**
churn	knn	0.7537	0.7872	13.4582	**1.72102459645182E-19**
churn	rf	0.8294	0.8417	7.0687	**2.24745826975125E-09**
churn	xgb	0.8066	0.8113	2.2320	**0.029489385684369**
churn	lr	0.8260	0.8407	9.1993	**6.22957726897507E-13**
churn	qda	0.7051	0.6712	-0.8345	0.407445464352508
churn	nb	0.7992	0.8204	7.9656	**6.99565772531347E-11**
adult	knn	0.6366	0.6609	19.2726	**6.82553765394818E-27**
adult	rf	0.7591	0.9154	161.1019	**1.3130743179194E-78**
adult	xgb	0.7968	0.9013	100.5639	**8.81659705918814E-67**
adult	lr	0.6400	0.6096	-2.3671	**0.021287563843755**
adult	qda	0.8576	0.8698	14.6883	**3.34561745547866E-21**
adult	nb	0.6855	0.8340	108.5290	**1.0845974193001E-68**
spambase	knn	0.8157	0.8643	18.7403	**2.77757550049706E-26**
spambase	rf	0.9449	0.9847	28.7081	**5.63942398571053E-36**
spambase	xgb	0.9609	0.9845	18.2811	**9.54149056667351E-26**
spambase	lr	0.8860	0.9649	35.7735	**3.1786405682527E-41**
spambase	qda	0.8964	0.9110	1.4530	0.15161661414028
spambase	nb	0.9295	0.9441	7.7888	**1.38470645802578E-10**

The results indicate that the labelling of a given test instance can best be performed by the original classification algorithm based on the highest probability. For the large majority of the tested combinations of datasets and classifiers the meta-approach outperforms the conformal prediction based on p-values, for most of the combinations significantly ($p < 0.05$). With these observations it can be concluded that with a simple model-agnostic non-conformity function, the Conformal Prediction framework is in most cases not able to outperform the base model when looking at overall performance. Only in a select number of situations with basic machine learning models, like Logistic Regression or Quadratic Discriminant Analysis, the conformal prediction outperforms the base model. Based on these results it was decided to use the meta-conformity approach in further research, meaning that the prediction function of the original machine learning classifier is used to evaluate the performance of the overall model as well as the determination of correlation and ranking of the test instances.

4.5 Conclusions

In this chapter we have described an approach to answer the first research question:

How do you predict the confidence score separately from class probability?

Confidence expressed with conformity To determine the confidence in a prediction separately from the class probability, the notion of conformity to the training data is used. The workings of the Conformal Prediction framework is described, together with an evaluation of the behaviour of this framework across different significance levels. The goal is to determine if selecting an instance in the test set based on higher confidence can improve the accuracy compared to the accuracy of the overall test set. To our knowledge, this is not evaluated before and can indicate the usefulness of confidence in selecting instances. The highly confident instances are determined to be the ones where the significance level separates the two p-values in the binary classification problem. An increase in accuracy is found when looking at predictions with a high confidence compared to all predictions based on the highest p-value.

The guaranteed error-rate was largely met when looking at the *inland* ship dataset and the different classifiers. In the folds not achieving the guaranteed error-rate, violations of the i.i.d. assumption are the most likely culprits. The assumption could be violated because ships can be inspected multiple times, making these inspections not independent. This is similarly the case for ships of the same owner. However, the error-rate was met when looking at the average over the folds in the cross-validation, indicating that the confidence predictions are sufficiently reliable overall for the specific problem of prediction violations on *inland* ships.

Finally, a short evaluation of the meta-conformity approach shows that using the predictions of the base model results in the best performance across a number of datasets and classifiers. The Conformal Prediction framework is therefore only used to determine the confidence in the predictions of the base model.

Chapter 5

Ranking with model confidence

In the previous chapter model confidence is determined using a measure of conformity. This confidence is evaluated by selecting highly confident instances and determining the accuracy of these instances in relation to the overall performance. Based on the improvement found in the previous chapter, in this chapter we use the obtained confidence from the Conformal Prediction framework to rank instances for multiple traditional machine learning problems as well as the real-world problem of violations on *inland* ships, answering the second research question:

Can model confidence predictions improve the ranking of predictions?

First, correlation is determined between the error-rate and the measure of *probability*, *confidence*, *credibility* and different combinations of these measures. The goal is to determine how to include the additional confidence measure in the ranking of the instances. This is followed by a second experiment where we perform ranking with several classifiers and datasets. The ranking based on individual measures is performed to gain insight into the behaviour of these values. With the correlations found in the first experiment, the approach for incorporating *confidence* in the ranking based on *probability* is determined and performed.

5.1 Taxonomy Ranking algorithms

The topic of ranking is a fundamental problem in the field of Information Retrieval. Many Information Retrieval problems are by nature ranking problems, such as document retrieval, collaborative filtering, key term extraction and sentiment analysis [54]. For each of these specific problems, the ranking problem is optimized. In general, there are three main groups of approaches can be defined; *pointwise*, *pairwise* and *listwise*.

Pointwise. With an existing implementation of a machine learning model, like the inland ship model, the most straightforward approach is using the model directly for the ranking problem as well. In this case it is assumed the model can precisely predict the *true* probability of a certain event or class, which for ranking is most often the relevancy of documents. Ranking the instances based on the *true* probability would result in the optimal ranking. This is the basis of the Probability Ranking Principle (PRP) [74]:

"If a reference retrieval system's response to each request is a ranking of the documents in the collection in order of decreasing probability of relevance to the user who submitted the request, where the probabilities are estimated as accurately as possible on the basis of whatever data have been made available to the system for this purpose, the overall effectiveness of the system to its user will be the best that is obtainable on the basis of those data."

The request of the user in the case of a traditional classification is the single question the prediction tries to answer. In the situation of inland ships, the request would be returning violating

ships, and the optimal ordering would mean all *true* violating ships having a higher probability of violation than ships not in violation.

Pairwise. In the real world, it is however not often possible to predict the *true* probability of an event. Furthermore, to achieve a good ranking it is not necessary to know the exact probability, only that one instance is more probable than the other. This idea is expressed in the pairwise approach, where the focus is on the relative order between two instances. The ranking problem is reduced to a classification problem on two instances. The classification problem is deciding for each pair which instance is more probable than the other. The optimal ranking is achieved in this case when the classification accuracy is 100%.

Listwise. Listwise approaches take the idea of the pairwise approach one step further. Instead of determining the order of only a pair of instances, the listwise approach tries to find an optimal order of the whole list of instances. There are two main sub-techniques for this approach. The first is optimizing the ordering of instances based on direct measures such as Normalized Discounted Cumulative Gain (NDCG) or Mean Average Precision (MAP). The other sub-technique focuses on minimizing a loss function specific to the ranking problem we want to solve.

5.2 Experimental design

For the evaluation of ranking with confidence measures from the Conformal Prediction framework, the existing Python library *nonconformist* version 2.1.0 is used [51]. The advantage of this library is the compatibility with the popular *scikit-learn* library [70]. All the different machine learning models inside the *scikit-learn* library are therefore compatible. Version 0.23 of *scikit-learn* was used during these experiments. For all evaluations of the Conformal Prediction framework the model-agnostic margin error non-conformity function was chosen, due to the performance of this function compared to other model-agnostic functions [36].

For the different experiments in this chapter a number of the classifiers are used. The focus in this chapter will be on tree ensembles techniques, namely Random Forest and XGBoost. These classifiers are chosen due to their performance on the specific problem of violation on *inland* ships. Other simpler models used during evaluation are *Logistic Regression*, *k-nearest neighbours* and *Naive Bayes*. Additional results with the other datasets and classifiers are presented in Appendix A.

First, the correlation between different measures and the error of the classification is calculated. Secondly, a ranking experiment incorporating the additional measures from the Conformal Prediction framework is performed on both individual measures as well as a combination of these measures.

5.2.1 Experiment 1: Correlation between error and additional metrics

With the use of the Conformal Prediction framework two additional measures of quality are determined for a given prediction; *credibility* and *confidence*. This is in addition to the *probability* given by the base machine learning model. Before incorporating the additional information in the ranking of the different instances, determining which combination of these values correlates strongest with the accuracy of the predictions can hint at the optimal combination. If the correlation between a measure and the prediction error is strong, this could improve the selecting or sorting of instances. This experiment is therefore an initial step to answer the second research question of this chapter. The score used in ranking will be based on the strongest correlation between the score assigned to an instance and the prediction error, as we cannot manually inspect all different combinations of measures and their performance in the ranking.

To determine if these values correlate with the performance of the base model, this section will describe the correlation between the error-rate and different combinations of the three available values as well as several combinations of these values. Strong correlation, negative in the case of the error-rate and positive in the case of accuracy, would indicate the usefulness of these values when ranking a list of predictions.

A number of different combinations of *probability*, *confidence* and *credibility* are evaluated. The first three are the most basic combinations where the additional measures of the Conformal Prediction are multiplied with the probability of the base model. The reasoning behind these combinations is the assignment of extra weights to the instances with both high probability and model confidence. By multiplying the instances with both low probability and either confidence or credibility are assigned a lower score. Another combination tested is the difference between the two p-values obtained by the Conformal prediction framework. The lower p-value is retrieved by 1 minus the *confidence*.

The Pearson correlation is used to determine the correlation between the error of the predictions and the different measures and their combinations. Pearson correlation is suitable for quantitative variables, including dichotomous variables. The model either classifies correctly or incorrectly, making the performance on an individual prediction a dichotomous variable. This is compared against the quantitative variable of either *probability, credibility, confidence* and combinations of these values. This correlation is used instead of the Spearman's rank correlation or the Kendall tau correlation. The reason being the binary labels of the problems in this thesis. This results in many ties, meaning there are only two average ranks when looking at the Spearman's rank correlation. A similar problem occurs for the Kendall tau correlation, where the optimal ordering cannot be determined; all positive instances can be randomly sorted and the ranking would still be just as optimal.

5.2.2 Experiment 2: Pointwise ranking with confidence measures

The second experiment of this chapter is defined to answer the second research question of this study:

Can model confidence predictions improve the ranking of predictions?

Different ranking problems are evaluated to determine if there is an improvement when incorporating model confidence.

Pointwise ranking with confidence measures

With the notion of model confidence clearly defined in the Conformal Prediction framework, this experiment will look at the effect of using this additional information for the ranking of the instances in a test set.

The problem is modelled as a ranking problem where different orderings of the test instances are evaluated. The sorting is achieved by assigning a numerical score to each instance. For the baseline of this evaluation this score is simply the probability predicted by the base machine learning model. As discussed before, if the probabilities determined are as accurate as possible, the sorting with these probabilities will give the optimal solution. However, in the real world these estimations are never 100% accurate. The sorting will therefore also not be optimal. With the inclusion of additional quality measure evaluation on the ranking is performed on a number of classifiers and datasets.

The ranking approach in this evaluation is pointwise as the score used for sorting the instances is determined based on the individual instances [54]. The assumption in this evaluation is that the ground truth of a violation translates to the ground truth of relevancy, the term often used in

ranking problems. A ship is only relevant to the inspector if there is a violation taking place. Instead of looking at the accuracy, where both the classification of violating and non-violating ships are evaluated, the metric of precision@k is used. This metric looks at different top k of a list and determines the precision for these k instances.

The reason for selecting a pointwise approach is twofold. Firstly, these classification problems are solved with probabilistic models, making the adaption to ranking with probabilities straightforward. Secondly, the problems in this thesis have binary labels and the two classes are balanced in all the datasets, making this particular problem a bipartite ranking problem [4][3][43]. For this specific problem, it is found that a good binary class probability estimation results in a good performing bipartite ranking [66]. The classifiers used in this study all learn by estimating the probability distributions and classifying by thresholding at 0.5. Meanwhile, for pairwise and listwise approaches, optimizing is difficult due to the large number of ties between instances; all positive instances can be shuffled with each other without any impact on the ranking performance, and the same is the case for negative instances. This results in half of the pairs being ties, while for the listwise approach a large number of optimal rankings exist. However, to confirm this intuition, a quick evaluation of these approaches was performed and compared against the pointwise approach laid out in this thesis. Here it is found that the pointwise approach does significantly outperform the pairwise and listwise approach. The results can be found in Appendix A.3.

Firstly, the ranking with confidence or credibility is compared against the ranking of probability in order to understand what the measures mean when sorting the test instances. With this understanding, an experiment re-ranking based on the model confidence is performed. For this re-ranking the results of the first experiment in this chapter are used. Based on the correlations found in this first experiment the re-ranking will be based on a multiplication of the *probability* and *confidence*, as it contained the highest overall correlation when looking at the different classifier and dataset combinations. As discussed in section 4.2, the confidence measures the quality of a prediction. For this experiment it is proposed to incorporate this measure into the existing ranking based on probability. Evaluation is done by determining the precision@k for the ranked list. The goal of re-ranking is visualized in Figure 5.1. Instead of relying on a single score to rank the predictions, the confidence is used to re-rank the probability ranking.

When looking at confidence and probability of single predictions, four quadrants can be defined in the outcome space of the overall system. When ranking based on a single measure instances in two quadrants are moved to the top of the list. In Figure 5.1 this is illustrated with the colored boxes, which relate to the instances in Figure 5.3.

By combining the confidence in a prediction with the probability the goal is to move highly confident and probable predictions to the top of the list.

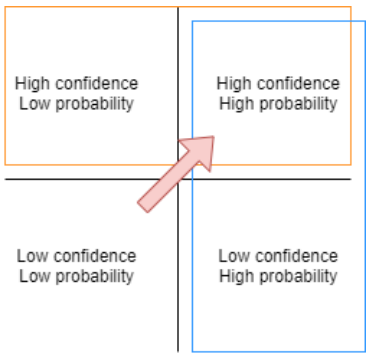


FIGURE 5.1: Outcome space of probability and confidence

To incorporating the confidence measure into the existing probability ranking, we will look at multiplying the confidence with existing probability:

$$P(y|x) * (1 - min(p_x)),$$

By multiplying, instances in the quadrant of high probability and high confidence are moved up the list and instances with only one of these two measure being substantial moved down compared to instances with a higher combination of probability and confidence.

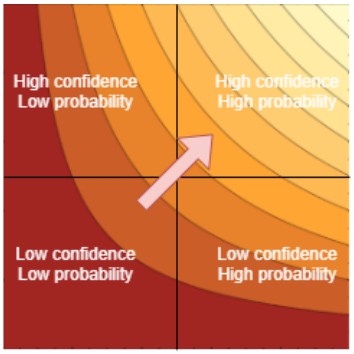


FIGURE 5.2: Increase weight highly confident and probable instances

5.3 Results

5.3.1 Correlation between error and additional metrics

In Table 5.1 these correlations are determined for the *inland ship* dataset with two tree ensemble classifiers; Random Forest and XGBoost. In Appendix A.2 the same results are given for the other machine learning datasets with the same classifiers.

The resulting correlation coefficients indicate that in certain situation the additional measures provided by the Conformal Prediction framework on individual predictions of a machine learning model can result in higher correlation with the error of the machine learning model. Between the basic metrics of *probability*, *confidence* and *credibility*, the probability is in most cases the strongest correlated metric with the error rate when using tree ensemble techniques. However, there are a number of instances of datasets and classifiers where the other metrics have a stronger correlation. When looking at the single metrics, the *probability* determined by a Random Forest model was always strongest correlated with the error, while for the XGBoost model *credibility* was strongest correlated with the error for 3 of the 4 datasets used in this chapter. XGBoost uses Gradient Boosting, which means the trees are built one tree at the time in an additive manner; the shortcomings of previous weak learners are used to create a better tree. With noisy data and no parameter tuning, this can result in slight overfitting. Recall that the *credibility* expresses overconfidence.

For the simpler machine learning models, the confidence measure is in a number of instances stronger correlated with the error. Recall that confidence tries to determine the quality of predictions, and in the case of the Conformal Prediction framework this is expressed with the probability of **a correct prediction**. With an accurate prediction of the *probability* of **an event** (class label in classification), this says more than *confidence* [77]. However, as these simple models can be less accurate in determining the *probability* of an event, the confidence correlates more strongly with error in certain cases. Combining the additional measures with the *probability* of the base model does give the strongest correlation in most dataset and classifier combination for at least one combination of these measures. Even when *probability* correlates more with error than *credibility* and *confidence*, combining these additional measures with the *probability* results in even higher correlation with the error of the predictions.

TABLE 5.1: The Pearson correlation between different measures and the error on the *inland ships* dataset

Metric	Random Forest		XGBoost	
	Correlation Coeff	p-value	Correlation Coeff	p-value
Conf	-0.104	2.98E-2	-0.048	0.21
Cred	-0.133	3.03E-2	-0.200	2.81E-6
Prob	-0.196	1.64E-5	-0.168	6.42E-4
Prob*Conf	-0.165	3.33E-4	-0.140	2.17E-2
Prob*Cred	-0.163	2.47E-4	-0.203	3.06E-6
Prob*Conf*Cred	-0.186	1.37E-4	-0.205	3.75E-6
Prob*(Conf+Cred)	-0.196	3.07E-5	-0.197	3.07E-6
Prob*(Conf^2+Cred)	-0.196	2.25E-5	-0.191	9.79E-5
Cred-(1-Conf)	-0.189	6.38E-5	-0.204	3.06E-6
Prob*(Cred-(1-Conf))	-0.192	5.30E-5	-0.205	2.36E-6

With the correlations found for the different classifier and dataset combinations, the multiplication of *probability* and *confidence* is found to be the strongest correlated with the error-rate of the base model. While not being the highest correlated in all situation, the correlation is robust.

5.3.2 Pointwise ranking with confidence measures

In this section the results of using model confidence to rank test instances of a classification problem are given. Firstly, it is described how individual measures perform when using these to rank the predictions. This is followed by using the *confidence* as an additional measure to re-rank the baseline of ranking based on the *probability*.

Probability versus confidence sorting In Figure 5.3 the ranking performance of the random forest classifier on the *inland ship* dataset is given for sorting based on probability versus confidence. The ranking performance was determined with 5-fold cross-validation repeated 4 times. This approach was chosen due to the large variation in performance when using smaller test sets, as is the case with 10-fold cross-validation.

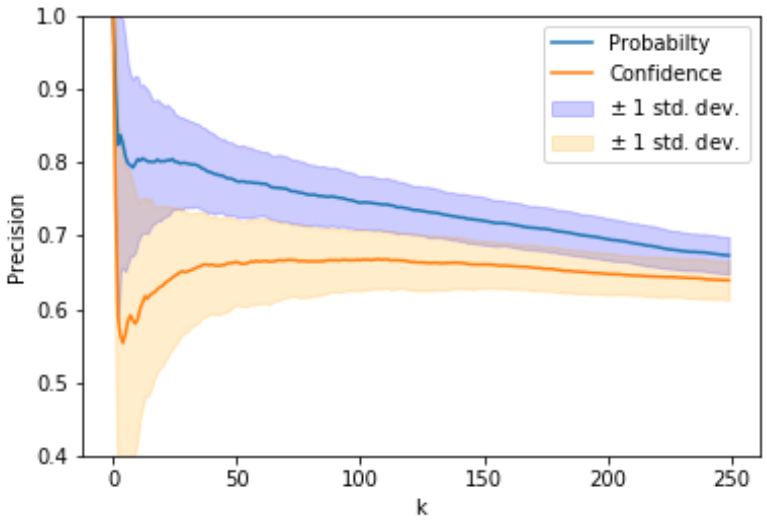


FIGURE 5.3: Precision@k on the *inland* ship dataset and RF classifier

The ranking performance when ranking based on confidence is significantly ($p < 0.05$) lower than with the baseline of probability. This is not unexpected, as the confidence score only looks at the quality of the prediction. For example, a prediction with only 0.51 probability can be non-conformal to the class not predicted, giving a high confidence to the prediction and resulting in the model being confident that the probability of 0.51 is correct. However, This still means that the other class is almost just as probable. When sorting based on the confidence measure alone, precision@k metric already approaches the precision performance on the whole test set at $k = 50$, with some fluctuation and large variation when $k < 30$ due to the limited number of instances. This indicates an even distribution of confident predictions over the whole probability space.

Probability versus credibility The same evaluation is performed with the credibility determined by the Conformal Prediction framework. Recall that credibility measures the conformity of the instance to the training instances with the same predicted class [77]. In Figure 5.4 the ranking performance of the random forest classifier on the *inland ship* dataset is given for sorting based on probability versus credibility. Using credibility to rank the test instances resulted in an improvement in precision@k with k between 1 and 15, significantly so ($p < 0.1$). With $k > 20$ the precision@k is lower when sorting by credibility versus probability. This confirms the results of the meta-approach evaluation, where the conformal prediction based on the highest p-value has a lower performance than the prediction of the base model.

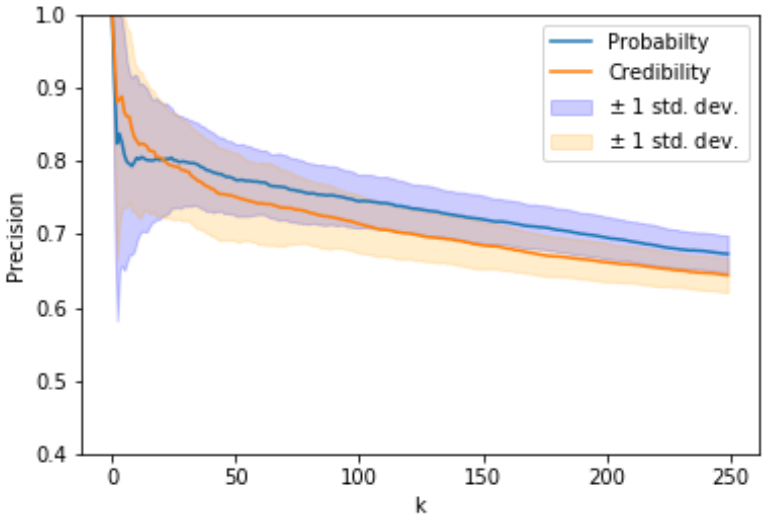
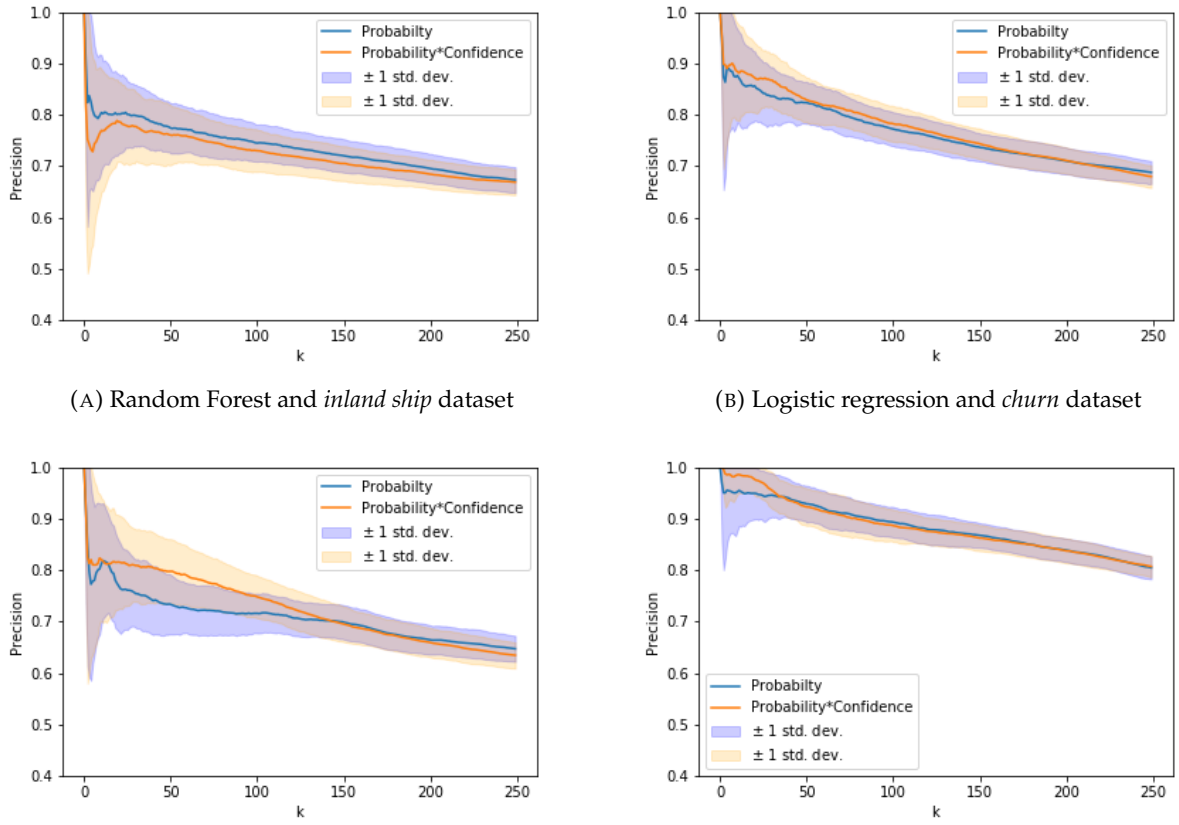


FIGURE 5.4: Precision@k on the *inland* ship dataset and RF classifier

Using confidence to re-rank

The final results describe using confidence in combination with the existing ranking based on probability to re-rank the baseline. Based on the correlations found in the first experiment in this chapter, multiplying the probability with the confidence was used to re-rank the instances. All the resulting precision@k with $k = \{5, 10, 25, 50\}$ are described in Appendix A.4. In Figure 5.5 a selection of combinations of classifier and datasets are plotted in their entirety.



(A) Random Forest and *inland ship* dataset

(B) Logistic regression and *churn* dataset

(C) Naive Bayes and *churn* dataset

(D) 11 nearest neighbours and *adult* dataset

FIGURE 5.5: Precision@k for different combinations of dataset and classifiers

The resulting precision@k indicates that, for complex models and datasets used in this evaluation, the addition of confidence in the ranking of the predictions in most situations does not improve ranking. However, for the simpler models included in this study, the inclusion of confidence in the ranking did give a significant improvement ($p < 0.05$) in most situations.

5.4 Conclusion

In this chapter we have introduced a novel pointwise approach of incorporating confidence measures based on conformity into existing probability based rankings. Confidence estimation tries to answer a different question, which can most easily be expressed as the quality of the prediction.

In order to determine the usefulness of this additional information, the following research question is answered:

Can model confidence predictions improve the ranking of predictions?

Incorporating confidence in pointwise ranking The baseline is pointwise ranking with probability, and this is compared with ranking by both confidence and credibility. These measures on their own do not improve ranking over ranking with probability.

Therefore, using the additional measures to re-rank the instances ranked by probability is proposed. Based on the correlation found between different combinations of the three measures and the error rate for a number of classifiers and datasets, the multiplication of *probability* and *confidence* is used to rank the instances.

Using the confidence measure to re-rank the list sorted by probability does not improve the ranking when looking at complex tree ensemble methods and the datasets used during this experiment. However, for simpler *interpretable* models, the addition of confidence does improve the ranking of predictions.

The reason behind this difference in performance between the re-ranking and the original ranking can be explained with the Probability Ranking Principle. This principle says that if the probabilities are as accurate as can be with the available data, ranking based on these probabilities is optimal. The complex tree ensemble models in this thesis perform better when looking at traditional evaluation metrics of classification problems. This indicates that these complex models are able to more accurately determine the *true* probabilities contained in the data used in this thesis. Therefore, the ranking with these probabilities is closer to the optimal ranking. The additional confidence measures do not improve the ranking in these cases. If the tree ensemble models were not able to accurately determine the probabilities from another dataset, the addition of confidence could improve the ranking of instances.

For the more interpretable models in this experiment the approximations of the probabilities are more rudimentary. Therefore the additional measure of confidence did significantly improve the ranking for most combinations of classifier and dataset, with a higher precision at the top of the sorted list.

Chapter 6

Explanations based on confidence

The overall goal for the problem of violations on *inland* ships in the Netherlands is supporting inspectors in deciding which ship to inspect. The real-world data for this problem is noisy and imperfect, resulting in a model with a relatively modest predictive performance. The goal of the thesis is using confidence to improve the human-decision support of the system in two ways. The first way is described in the previous two chapters; determining reliable confidence metrics and ranking based on this confidence of the model. This allows for a selection of predictions with a higher precision. The second way is discussed in the next two chapters; explaining from the context of *confidence* instead of the traditional context of *probability*.

There are a large number of explanation frameworks developed over recent years. These frameworks try to explain the output of a machine learning model, which in most cases for classification is probability. In this chapter not only the probability of the base model is explained, but also the confidence in the prediction, as determined in Chapter 4. For this study it was chosen to use SHapley Additive exPlanations (SHAP), an additive feature attribution method.

The reason for choosing this framework is the model-agnostic approach with desirable properties; *local accuracy* and *consistency* (6.1.2). SHAP approximates the Shapley value, a concept from cooperative game theory, in order to determine feature contributions.

The goal of this chapter is answering the following research question:

How can confidence prediction be used to generate model-agnostic local explanations?

First, a summary of the workings of the SHAP framework is described. This is followed by adapting the SHAP framework to determine the feature contributions of model confidence alongside the probability of the prediction.

Finally, an exploratory data analysis is performed comparing the feature contributions between the context of *probability* and the context of *confidence*. The goal is determining if the feature contributions, and therefore the explanations, differ enough between the contexts to be separately evaluated in a user study and determine how they differ between the contexts.

6.1 SHapley Additive exPlanations

SHAP uses Shapley values for model feature influence scoring. Shapley values were initially proposed by Shapley in the field of game theory [78]. Shapley values are a way for assigning payouts to players depending on their contribution to the total payout. The payout in this case is the prediction (probability) and the players are the different features of a given instance. The Shapley value is the average marginal contribution of a feature value across all possible combination of features. This means going through all possible features possibilities and look at the prediction made to determine the average feature influence. This exhaustive approach guarantees consistency and local accuracy, something that is not the case for LIME [56]. However, a drawback is

that calculating these Shapley values based on all possible permutations is extremely computationally expensive.

6.1.1 The Shapley Value

Shapley value was proposed decades ago in the field of game theory [78]. The Shapley value is the average marginal contribution of a player across all possible coalitions of players in a cooperative game. With these marginal contributions, the goal is to determine how each player contributes to the outcome of the game, commonly called the payout. The Shapley value is determined as follows:

$$\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n - |S| - 1)!}{n!} (v(S \cup \{i\}) - v(S))$$

where N is the total set of n players with S being a subset of players in N. The function $v(S)$ determines the expected payout of the players in S can obtain by working together. $v(S \cup \{i\}) - v(S))$ represents the influence of including player i in the game. This formula can be explained as follows; the players enter a room in random order. All players in the room participate in the game. The Shapley value of a player is the average change in the payout that the coalition already in the room receives when the player joins the other players for all possible combinations.

The Shapley value is a unique solution and has many desirable properties, like *efficiency*, *symmetry*, *additivity* and *null player*.

Efficiency The sum of all Shapley values of individual players equals the total payout of the cooperative game:

$$\sum_{i \in N} \phi_i(v) = v(N)$$

Symmetry The contributions of two players j and k should be the same if they contribute equally to all possible coalitions.

$$\forall S \in N : v(S \cup \{j\}) = v(S \cup \{k\}) \wedge j, k \notin S \Rightarrow \phi_j = \phi_k$$

Additivity When combining payouts the respective Shapley values can also be combined to determine the overall Shapley values.

$$v(S_1) + v(S_2) = v(S_3) \Rightarrow \phi_{S_1} + \phi_{S_2} = \phi_{S_3}$$

Null player When a player does not influence the payout, the Shapley value is zero.

$$\forall S \in N : v(S \cup \{i\}) = v(S) \Rightarrow \phi_i = 0$$

The Shapley value is the *only* solution with these desirable properties, as was demonstrated by Young [96].

6.1.2 SHAP value

The concept behind the determination of the Shapley value can be translated to a more general machine learning problem as a unified measure of feature importance. Instead of determining the payoff for all different coalitions of players, the influence of a feature is determined by looking at the outcome of the model for all different combinations of features. The function $v()$ for determining the SHAP values expresses the influence of a feature on the difference between the

expected prediction and the prediction of an individual prediction when conditioning on that feature. While not the first using Shapely values in the determination of feature contributions, SHAP by Lundberg & Lee is a popular study and implementation of the concept [56]. In this study the approach of determining Shapley values is compared with other model-agnostic local explanation techniques which all approximate a complex model with a linear model, with binary values representing if a feature is included in the output. The binary values therefore represent the coalition of features. The study describes these specific approaches as additive feature attribution methods. The linear model in all these techniques is used to determine feature contributions:

$$f(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n$$

For a linear model shown above, the determination of feature contribution is simply

$$\phi_i(f) = \beta_i x_i - E(\beta_i x_i)$$

where $E(\beta_i x_i)$ is the average (expected) effect of feature i over the whole dataset.

Compared against other additive feature attribution methods

A total of four additive feature attribution methods are compared against the Shapley values, including LIME [73]. The definition of the Shapley value shows that there is only one possible additive feature attribution method with *local accuracy* and *consistency*. This result implies that methods not approximating Shapley values violate local accuracy or consistency.

The *local accuracy* looks at how close the linear model approximates the base model locally, meaning that the approximation could be less accurate when looking globally. Local accuracy is defined as follows:

$$f(x) = g(x') = \phi_0 + \sum_{j \in N} \phi_j x'_j$$

In this equation $x = h_x(x')$, with h_x converts binary values into the original inputs. The binary values represent if the local approximation uses the feature value in the original input space.

In this definition of local accuracy, if ϕ_0 is set to the expected average output of $f(x)$ and all x'_j to 1, the definition is the same as the *efficiency* property of the Shapley value.

Consistency means that if a model changes so that the marginal contribution (β) of a feature value increases or stays the same, the Shapley value also increases or stays the same. This consistency property can be defined as follows; for any two models f and f' that satisfy:

$$f'(z') - f'(z' \backslash i) \leq f(z') - f(z' \backslash i)$$

for all inputs $z' \in \{0,1\}^N$:

$$\phi_i(f', x) \leq \phi_i(f, x)$$

Consistency is a more generalized property, from which the properties *symmetry*, *additivity* and *null player* of the Shapley value can be determined [56].

The proof of a unique solution indicates that both LIME and other generalized additive feature attribution models do not have the desirable properties when not approximating Shapley values. This does not mean these approaches cannot have these properties, as can be seen in the KernelSHAP definition discussed in Section 6.1.3.

6.1.3 Approximation method

To calculate the SHAP values exactly is challenging. For all of the possible coalitions, which are factorial in the number of features, the expected prediction with and without the coalition of features has to be calculated, which is exponential in the number of features, resulting in $O(N! * 2^N)$. This makes this exhaustive exact approach untenable for large datasets.

Therefore, approximations of the Shapley values are proposed. In the original study on SHAP the model-agnostic KernelSHAP technique is proposed [56], together with a number of model-specific approximations. In a later follow-up study a model-specific approach to tree ensembles is proposed [55].

KernelSHAP

The proposed KernelSHAP approach is an adaptation of the LIME method by Ribeiro [73]. Recall from the literature survey that LIME is a technique for locally approximating a complex model with an interpretable model. The explanations are generated by the following:

$$\xi = \arg\min_{g \in G} \mathcal{L}(f, g, \Pi_x) + \Omega(g)$$

where g is a model in the class of interpretable models G. f is the complex model with $f(x)$ given the probability of a certain event. $\Pi_x(z)$ is a distance measure between x and z in order to define when another instance is local. Finally $\Omega(g)$ determines the complexity of the interpretable model g. For a linear model this can be the number of non-zero weights or the depth of the tree in the case of a decision tree. Finally, let $\mathcal{L}(f, g, \Pi_x)$ be a measure of how unfaithful g is in approximating f in the locality defined by Π_x. This represents the *local accuracy* or *local fidelity* of the interpretable model.

In order to ensure both interpretability and local accuracy, the function $\mathcal{L}$ has to be minimized, while not increasing the complexity $\Omega(g)$ too much. Making the decision on which interpretable linear model to use is a trade-off between faithfulness to the original model (*local accuracy*) and *interpretability* of the approximate linear model.

Based on the ability to set the parameters of the loss function, this approach is not guaranteed to produce Shapley values, the unique solution with the desirable properties. However, with certain forms, the Shapley values can be approximated:

$$\Omega(g) = 0,$$

$$\Pi_x(z) = \frac{M - 1}{\binom{M}{|z|} |z|(M - |z|)},$$

$$\mathcal{L}(f, g, \Pi_x) = \sum_{z \in Z} (f_x(z) - g(z))^2 \Pi_x(z)$$

where $|z|$ is the number of non-zero elements in z and M the number of input features.

For the KernelSHAP approach a linear approximation model is used for $g(z)$. Together with the loss function $\mathcal{L}$ being a squared loss function, it is possible to approximate the SHAP values with linear regression. Together with the properties of LIME, this allows regression-based, model-agnostic estimation of SHAP values with the properties of *local accuracy* and *consistency*.

6.1.4 SHAP Interaction values

As discussed in section 3.3.3, a benefit of SHAP values is that the contribution of a feature is determined not only based on the individual feature value, but by coalitions of features. This means that features can *interact*, where globally SHAP values of a feature is not an exact linear relation. In other words, two instances with the same feature value for a single feature can have different SHAP values for this single feature. To determine the interaction between features, the Shapley interaction index determines which features interact most with a single feature:

$$\phi_{i,j} = \sum_{S \subseteq \setminus \{i,j\}} \frac{|S|!(n - |S| - 2)!}{2(n-1)!} \delta_{i,j}(S),$$

where $i \neq j$ and:

$$\delta_{i,j}(S) = f_x(S \cup \{i,j\}) - f_x(S \cup \{i\}) - f_x(S \cup \{j\}) + f_x(S)$$

By removing the effect of the two single features, the influence of the features interacting is determined. This is done, as with the calculation of the original Shapley values, for all possible coalitions of features.

6.1.5 Types of explanations

The Shapley value of a feature is not the difference between the predicted value after removing the feature during training. Instead, given the current set of feature values, the contribution of a feature value to the difference between the actual prediction and the mean prediction is the estimated Shapley value. This definition allows for contrastive explanations that compare the prediction with the average prediction.

Furthermore, features contributing to a decrease of the probability or confidence can be defined, not only features contributing positively to the prediction. This can be beneficial in a human decision support system, like the one for the inspectors of inland ships. As the model is not 100% accurate, a number of predictions are incorrect. Determining features lowering the outcome of the model could be useful information when the user disagrees with the prediction made.

6.2 Implementation of SHAP

The techniques proposed for SHAP are incorporated in a Python library by Lundberg [56][55]. This implementation has the option to determine for a single instance the Shapley value as "forces". Here the values can either be positive or negative. The value represents how much influence (force) a given feature has on increasing or decreasing the output of the model from the baseline score. This baseline score is the average output over all the different instances. As discussed before, the output for probabilistic classification models is the probability of the instance belonging the to predicted class. The SHAP library version 0.35.0 is used during this study [80].

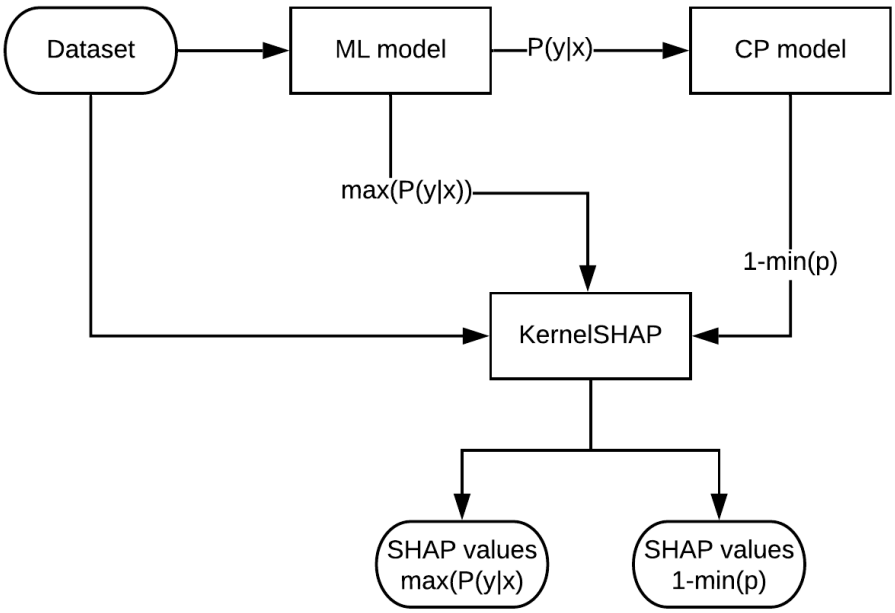


FIGURE 6.1: Overview of the determination of two feature contributions

The SHAP framework was adapted to not only look at this probability output, but also determine the Shapely values for the confidence score determined by the Conformal Prediction framework. This requires to define a custom function in the *nonconformist* library returning the confidence of an individual prediction for the Inductive Conformal Predictor, which was wrapped to passed on to the SHAP framework. An overview of the pipeline of this approach is given in Figure 6.1.

As the SHAP framework has to work on both the base machine learning model and the Conformal Prediction model, the model agnostic approach KernelSHAP is used to determine the SHAP values for both the probability and the confidence of an individual prediction. The base model used is a random forest model, and the Conformal Prediction framework uses the model-agnostic margin non-conformity function. These contributions are used to explain the answer of two different questions: *"What is the probability of a violation?"* and *"How confident are we in the prediction of violation being correct?"*. The answer is simply the output of either the base model or the Conformal Prediction framework, a single value in both cases. The approach in explaining the answer for both questions is the same; the contributions explain which information contributes to the probability or confidence of a prediction.

Before comparing the explanations of the two unique contexts concerning user trust with a user study, an exploratory review of these contributions is done. The goal is determining the difference between the two contexts of explanations. For a total of 250 ships in the *inland* ship dataset the Shapley values are determined for the 252 features, resulting in 63000 SHAP values for each context. The 250 instances were selected with k-means clustering, which additionally assigns a weight representing how many instances a single instance represents in the test set.

Basic metrics such as average SHAP value and distribution is determined to get a general idea about the difference of SHAP values between contexts. The features with the greatest difference between context are defined. Global feature importance using the SHAP framework is also calculated to see if there is a difference in the global influence of a single feature. The SHAP values for individual predictions are also compared with two examples, as well as with a combination plot of all the SHAP values. Finally, a number of dependency plots are made. These show the relation

between the feature value and SHAP value and give insight into the behaviour of these values in both contexts. In the following section the resulting plots and metrics are given for the *inland* ship dataset. In Appendix B the same plots and metrics are given for the *churn* dataset.

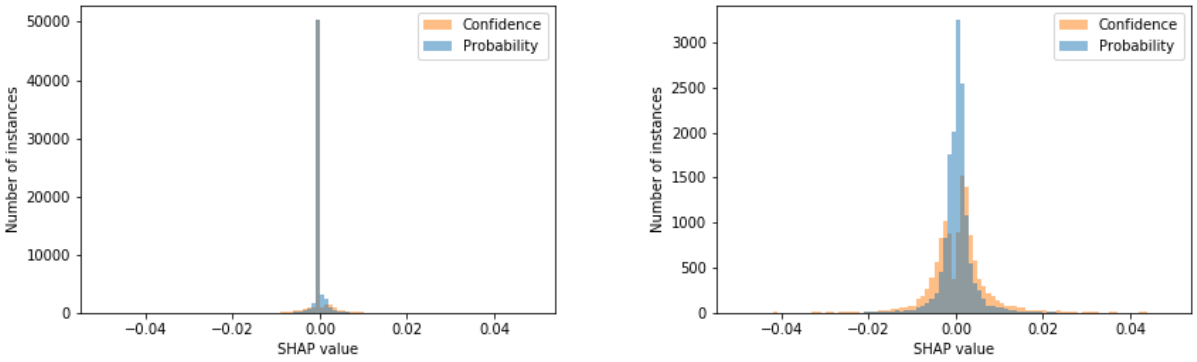
6.3 Exploration SHAP values between two contexts

To compare the difference in SHAP values, a number of visualizations and metrics are used. First, the absolute SHAP values are averaged between the two contexts of probability and confidence. This is also done when removing all SHAP values of zero (*null players*). The SHAP values of 0 mean that the feature is not contributing to the given prediction. The number of non-contributing features between the two contexts is also given. The results are given in Table 6.1.

TABLE 6.1: Difference in SHAP values between the context of *probability* and *confidence*. This is done for all values and all non-zero values. The number of zero instances is also given.

Context	Average absolute SHAP value	Without 0	# instances of 0
Probability	6.94E-4	2.97E-3	48291
Confidence	1.40E-3	6.81E-3	50004

The absolute SHAP values explaining the confidence of the Conformal Prediction framework are, on average, around two times as large compared to those explaining the probability of the base model. This means that features are contributing a larger amount to the overall output of confidence. This happens while a larger number of SHAP values are zero, meaning the feature has no impact on the individual prediction. To see whether this is the cause for both positive and negative SHAP values, the SHAP values are plotted in a histogram in Figure 6.2 for both cases in Table 6.1. From the figure it can be concluded that when explaining the confidence for the *inland* ship dataset, the SHAP values are larger for both negative and positive features globally.



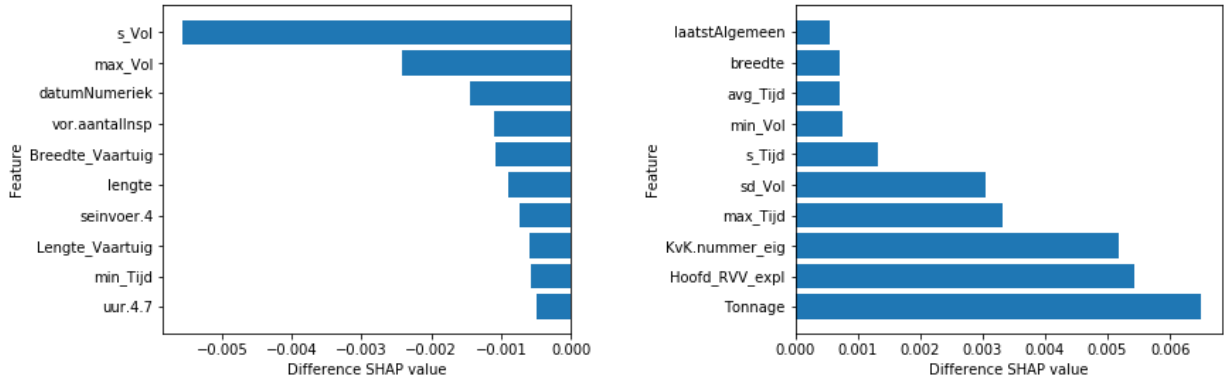
(A) SHAP values between two contexts

(B) SHAP values minus zero instances

FIGURE 6.2: Difference in SHAP values between two contexts

Difference between SHAP values

Besides the difference in the size of the SHAP value, it is interesting to determine the most significant difference between SHAP values for a given feature. In Figure 6.3 the average difference between the two contexts is given. The negative difference means that the SHAP values are lower in the context of *confidence* compared to the context of *probability* and a positive difference means an increase in the SHAP values.



(A) Decreased SHAP values

(B) Increased SHAP values

FIGURE 6.3: Difference of SHAP values between the two contexts

The overall difference in SHAP values strongly relates to the difference in global feature importance, however, the correlation is not necessarily positive for an individual feature. An overall positive difference does mean the feature contributes more to an increase in the model output. When looking at the *inland* ship dataset, the unique identifiers of the ship's owner contribute to a larger output in the context of *confidence*. At the same time, information about fuelling does have a more considerable negative impact on the confidence. In the following section the global feature importances between contexts is given.

Global feature importance

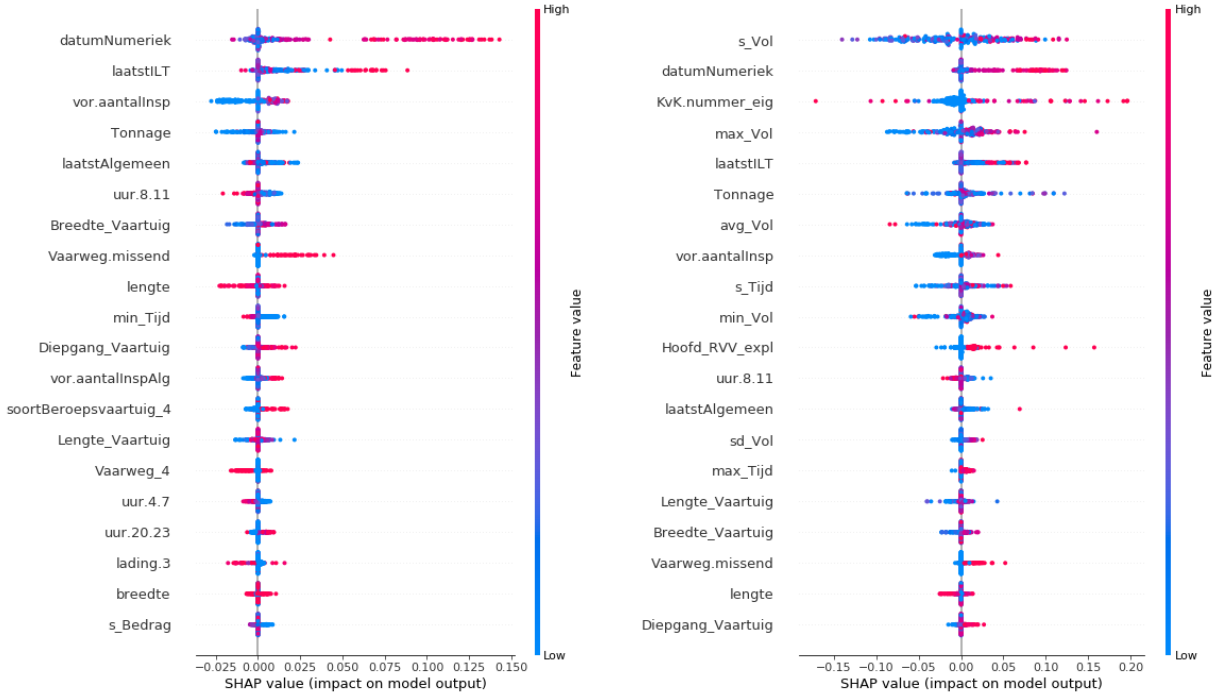
To determine the global feature importance with the SHAP framework, a summary plot of the two contexts is given in Figure 6.4. The summary plot combines feature importance with feature effects. The points represent a Shapley value for a feature and instance combination, with the colour representing the value of the feature from low (blue) to high (red). The features are ordered according to their importance.

The features importances between the two approaches differ when averaging over the 250 instances. The reason for the reduction in the number of instances is the large computational time as the number of combinations increases exponentially with the number of instances. The behaviour of individual features and their SHAP values between the contexts is mostly the same. Meaning that a positive correlation between a feature value and its SHAP value in the context of *probability* also means a positive correlation in the context of *confidence*. This indicates that the importance of the feature globally differs between the two approaches, but the relation between the feature value and SHAP value does not.

When looking at the feature importances between contexts on the *inland* ship dataset, the features on fuelling (*s_Vol*, *max_Vol* and *avg_Vol*) and identifiers of the owner of the ship (*KvK.nummer_eig* and *Hoofd_RVV_expl*) are contributing more to the determination of confidence compared to the probability. These were also the features which differ most between the two contexts. The differences were both negative and positive, however, this does not mean the contribution of the feature is not increased. Recall that confidence is determined by looking at the conformity in the probability space of a test instance to the training data. Fuel information and the identifiers of the owners are therefore useful features to determine similarity or conformity between instances.

Individual predictions

As mentioned before, the SHAP values are determined for explaining individual (local) predictions. An example of these SHAP values for an individual prediction is given in Figure 6.5. In this



(A) Probability feature contribution (B) Confidence feature contribution

FIGURE 6.4: Summary plot of the most important features

figure force plots are given. Each block represents a SHAP value in this plot, where the width of the block represents how large the SHAP value is.

The two plots represent different contexts to explain. The top plot shows how individual features contribute to the decrease or increase in the base model output compared to the average prediction output. The lower plot does the same for the output of the Conformal Prediction framework. These individual plots are represented in Figure 6.4 with a point for each feature.

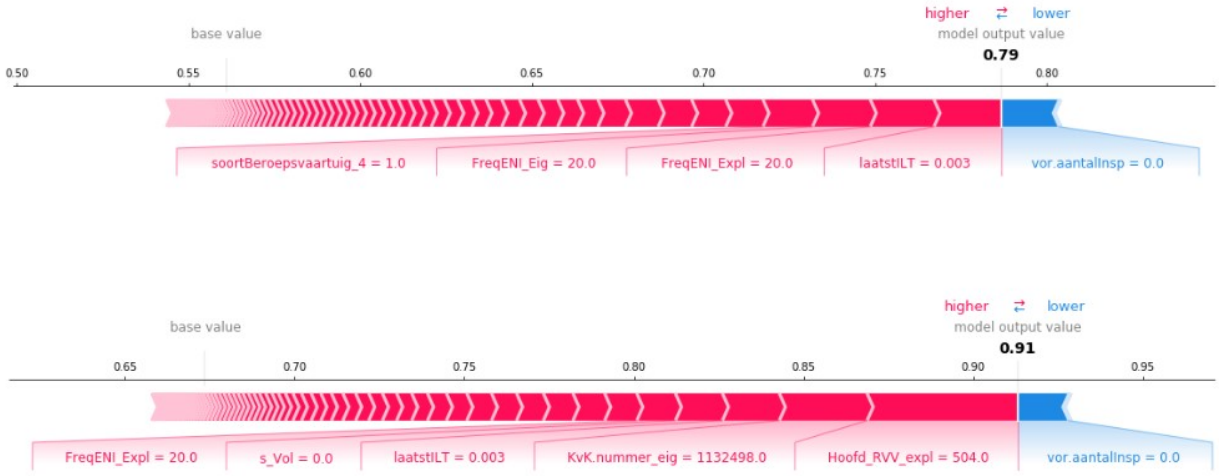


FIGURE 6.5: SHAP values of a single prediction of the RF on the *inland* ship dataset. The plot *above* is from the context of *probability*, the *lower one* from the context of *confidence*

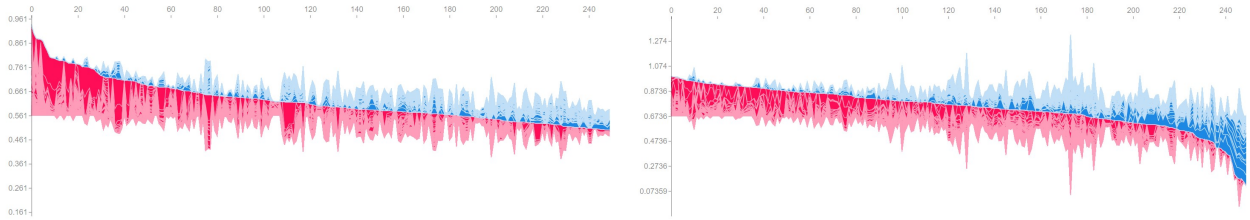
The two plots also indicate the difference in feature influence between the two contexts. While

a positive contributing feature in one context will not negatively contribute in the other context, how much a feature influences the increase or decrease from the average prediction does differ between the two contexts. In both contexts many are contributing, however small, to the overall prediction. This is caused by the definition of ξ, where all features are tested.

Combining individual predictions

These individual plots can also be combined into a single plot to get a more general overview of the difference between the two contexts. The values shown in the individual plots are rotated 90 degrees and put beside each other in Figure 6.6, sorted by the output of the model.

The resulting plots are harder to distinguish than the individual ones, however, several observations can be made. Firstly, the features negatively contributing to the prediction are more dominant when looking at model confidence compared to the probability. This could help in deciding to disagree with the prediction of the model, with information making it clear why the confidence is low. The overall size of the SHAP values in the context of *confidence* is larger, as found before. In this case, this results in a higher average prediction, as well as more "variation". This kind of variation is caused by sizeable opposing SHAP values, meaning that a larger number of features have large SHAP values.



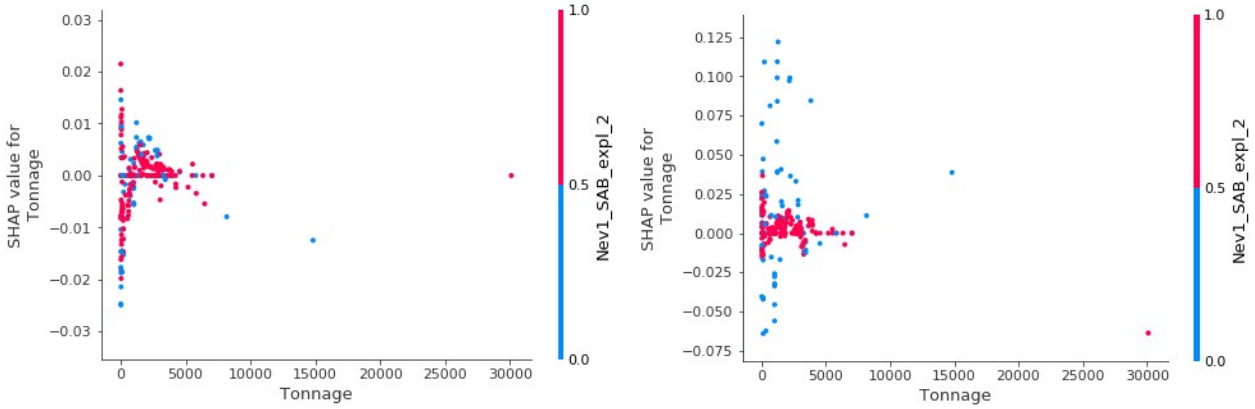
(A) Probability feature contribution of all individual prediction

(B) Confidence feature contribution of all individual prediction

FIGURE 6.6: Force plot of all individual predictions

Interaction between features

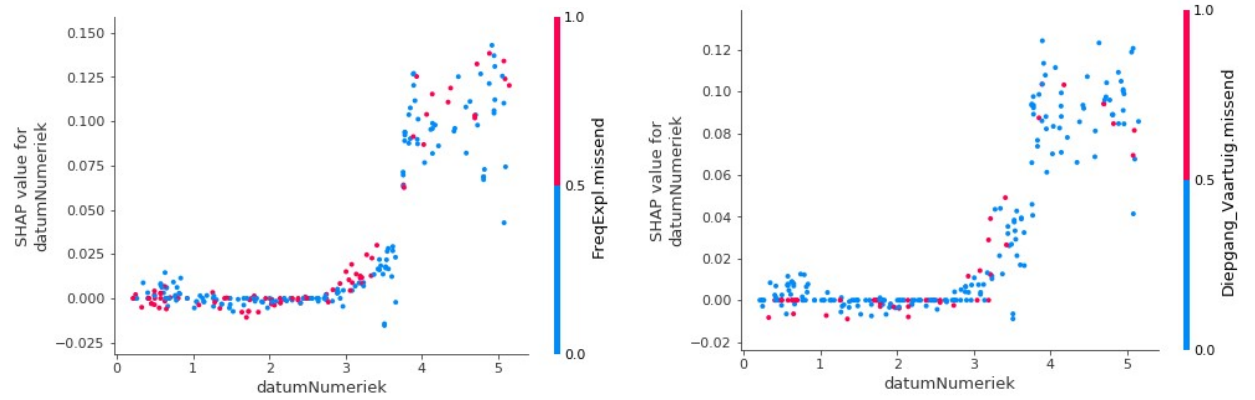
Finally, we look at a number of feature dependency plots comparing the two contexts. Dependency plots plot the feature value against the SHAP value for all instances to determine the behaviour of a given feature. If there is a perfect line in such a plot, there is no interaction with other features. When there are feature values with different SHAP values, there is an interaction with other features. This means that a combination or coalition of feature values influence the overall contribution of the individual features. In Figure 6.7, the dependency plot of the *Tonnage* feature is given for both contexts. This feature was selected due to the relative similar global importance in both contexts as well as being a large continuous feature.



(A) Probability context (B) Confidence context

FIGURE 6.7: Relationship of the *Tonnage* feature value and SHAP value. The most interacting feature *Nev1_SAB_expl_2* is represented with colour

When ignoring the colour of the plot initially, we see there is almost no correlation between the feature values and SHAP values. This means the interaction with other features largely determines the SHAP value. The SHAP framework has the build-in function to determine interaction values. The most interacting feature in both contexts, *Nev1_SAB_expl_2*, is plotted with colours. This feature represents if the second activity of the owner of the ship is the transport on water. To determine whether the interaction between the two features contributes to the difference in SHAP values, separation of instances based on the additional colour should be possible. Therefore, the interaction is not that strong when looking at the context of *probability*. However, in the context of *confidence*, these two features do interact; if *Nev1_SAB_expl_2* is 1, the SHAP value of *Tonnage* is pushed towards 0 more than when it is 0.



(A) Probability context (B) Confidence context

FIGURE 6.8: Relationship of the *datumNumeriek* feature value and SHAP value

In Figure 6.8 the same plot is given for the feature *datumNumeriek*. Here a stronger relation between the feature value and SHAP value is found. There is still interaction between features, however, between the two contexts, the most interacting feature is different. The interactions are in both contexts not strong enough to separate the instances and explain the difference in SHAP values for similar feature values. The difference is likely caused by a large number of smaller interactions, making the exact determination of the difference between context difficult. However, we do see that the interaction between features differs between context, allowing in some cases to increase the interaction. Also, notice that the relation between feature values and

SHAP values remains mostly the same between contexts, as was found when looking at global feature importances.

6.4 Conclusion

The goal of this chapter is answering the third research question:

How can confidence prediction be used to generate model-agnostic local explanations?

In this chapter we proposed to use the SHAP framework to explain two different contexts; *probability* and *confidence*. The SHAP framework allows for determining feature contribution for any output of a model, with the desirable proprieties of *efficiency*, *symmetry*, *additivity* and *null player* by approximating Shapley values. The definition of a Shapley value proves that only when approximating Shapley values, an additive feature attribution method has these desirable properties. The SHAP values allow for model-agnostic explanations for individual instances. The SHAP values for an individual instance show how much a particular feature "forces" the output of a model from the average output of the model, which can be in both a positive and negative direction.

For this research a novel approach for explaining model confidence is proposed. The SHAP framework is adapted to determine not only the feature contributions towards probability, but also the confidence in the classification.

These contributions are used to explain the answer of two different questions: *"What is the probability of a violation?"* and *"How confident are we in the prediction of violation being correct?"*. The answer is the output of either the base model in the case of probability or the Conformal Prediction framework in the case of confidence, a single value in both cases. The approach for explaining the answer for both questions is the same otherwise; the contributions explain which information contributes to the probability or confidence of a prediction.

Exploratory data analysis is performed to determine the difference in SHAP values between the contexts of probability and confidence. The SHAP values between contexts differ both locally and globally. For the *inland* ship dataset, the SHAP values are larger when explaining confidence and a larger number of features contribute to the confidence prediction. There is a difference in global feature importance between the context of *probability* and *confidence*, however, the behaviour between feature value and SHAP value remains largely the same. This means that if globally a feature value is positively correlated with the SHAP value of this feature in the context of *probability*, the correlation is also positive in the context of *confidence*. The slope and the linear relationship of the feature and SHAP value do slightly differ between the contexts. Locally, this causes the other features having the highest absolute SHAP value between the contexts. This is caused in part by the difference in the interaction between features, meaning that certain features have a different impact on the SHAP value of another feature in one context compared the other. An example would be that while the contribution of the weight of a ship being 3000 kilograms in one context does not depend on the fact that the inspection takes place in the weekend, in the other context this does increase or decrease the contribution of the weight of the ship.

With these results, it is found that the feature contributions of individual instances, and therefore the explanations, differ significantly between the contexts. This difference is mostly in the magnitude of a contribution due in part to different interactions between features.

Chapter 7

A human-grounded evaluation of confidence based explanations

In the previous chapter a novel approach of explaining from confidence was proposed; adapting the SHAP framework such that explanations from the context of *confidence* are determined with feature contributions. In the data analysis a significant difference between the context of *confidence* and the context of probability was found when looking at feature contributions. In this chapter we evaluate how this difference is received and perceived by users. The goal of the evaluation is answering the final research question:

How are explanation based on confidence received by users?

This is achieved with a human-grounded evaluation, which takes the form of a user study where the two contexts of explanations are compared for the real-world problem of inspecting *inland* ships in the Netherlands. The explanations from the context of probability serve as a baseline upon which to compare the confidence based explanations. *Task effectiveness*, *perceived usefulness* and *user trust* are used as evaluation metrics, which are tested with three corresponding hypotheses. Additional analysis is performed by determining the usefulness of individual features, feedback by the participants, the time taken to complete the tasks as well as separating the results based on the prediction and the correctness of this prediction.

7.1 Evaluating explanations

The topic of explanation methods making complex machine learning predictions more interpretable has become increasingly popular in recent years. However, evaluating the usefulness of these kinds of explanations has not been done thoroughly [24]. The few studies that do look more closely at evaluating these kinds of explanations use confidence and probability often interchangeably. The focus for this evaluation is, therefore, looking at the impact of separating these terms and explaining them separately, answering the question: *What is the impact of explaining model confidence instead of probability?*

For this study, a model-agnostic approach has been chosen because the focus of the experiment is on evaluating explanations for the confidence values determined by the Conformal Prediction framework against explanations for the probabilities of the base model. This means that explanations will try to increase the interpretability of two different machine learning models. In this work, a human-grounded evaluation to determine the usefulness of explaining a complex machine learning model from the perspective of confidence instead of probability is described.

7.2 A human-grounded evaluation

As discussed in the literature review in Chapter 3, human-grounded evaluation is about conducting simpler human-subject experiments that maintain the essence of the target application [24]. Here the target community does not need to be participating in the evaluation. A larger pool of people can therefore be included in these kinds of evaluations. So instead of determining the quality of an explanation in a certain context, more concrete tasks are evaluated where the quality of the explanation can be inferred from this smaller task.

7.2.1 Research hypotheses

To evaluate the impact of explaining model confidence, a number of hypotheses are formulated. These look at the impact the two different explanations have on the task performance for inspection decisions. The task performance is evaluated by the metrics of *explanation effectiveness, explanation usefulness* and *user trust*.

Explanation effectiveness

Explanation effectiveness examines the impact of the explanation on the effectiveness of the human decision task. When looking at the decision on whether to inspect a certain ship, explanation effectiveness can be defined simply as the accuracy of the decisions made. If an explanation is effective, it will help the user make better decisions, as an effective explanation helps the user evaluate the quality of the prediction. By assessing this quality with the explanation the user can disregard incorrect predictions. The two different explanations have different model outputs as well as different feature contributions. Explanations based on confidence of the model in a certain prediction could more closely resemble human intuition and increase the explanation effectiveness. This could be the case as the explanation based on confidence presents how confident the model is in a certain prediction, instead of giving the probability of the most likely outcome. When the model indicates to be less confident in its prediction, it could lead the user to assess the model's prediction more closely. Additionally, the fact that the feature contributions between the two different explanation methods are not the same could help the user toward more essential features for determining if they trust this particular prediction.

Hypothesis 1. *Explanations based on confidence of a prediction increase explanation effectiveness of a ship inspection decision compared to the explanation based on the probability of the prediction.*

Explanation usefulness

Explanation usefulness examines how the explanation is used in the overall decision made by the human decision-maker. Both explanation approaches reveal features that are important for the decision or confidence of that decision. This allows the experts to determine if the connection between these features and the overall decision makes sense. By determining which part(s) of the explanation contributed towards the overall decision on whether to inspect the ship, it can be determined if the explanation is used in the decision-making process. Furthermore, this information could also help determine the features the user finds important in the decision-making process. This could help in follow-up work in feature selection in the overall system.

Hypothesis 2. *Explanations based on confidence of a prediction increase explanation usefulness compared to the explanation based on the probability of the prediction.*

User trust

User trust examines how confident the user is in using the model or its predictions. This is an important metric for evaluating the explanations: a system which looks to aid a human decision process should have the trust of the user. Otherwise, the user could simply ignore the system altogether.

Transparency of the explanation is sometimes linked with user trust, with some prior studies finding that transparency of the system increased the user trust in the system [53]. The reasoning is that if a user can understand why the model made a certain prediction, the user can be more confident in the prediction being correct (or not).

Hypothesis 3. *Explanations based on confidence of a prediction increase user trust compared to the explanation based on the probability of the prediction.*

These hypotheses are tested with a user experiment where users have to decide if they want to inspect a certain ship based on either an explanation based on confidence or probability. The procedure of this experiment is laid out in the next section.

7.3 User study

The main goal of the experiment is to compare explanations based on the same explanation framework and interface, but with different outcomes which need to be explained. The participants of the study are experts working for the inspectorate, and therefore only a limited number of participants are available. This resulted in choosing a within-subject design for this experiment. A within subject design means that every participant is subjected to all different conditions of the independent variables. For this experiment that means showing every participant explanations based on the probability as well as explanations based on confidence. The different value that has to be explained is the only independent variable in this experiment consisting of two levels. The dependent variables are measured in the form of answered questions by the participants of the study.

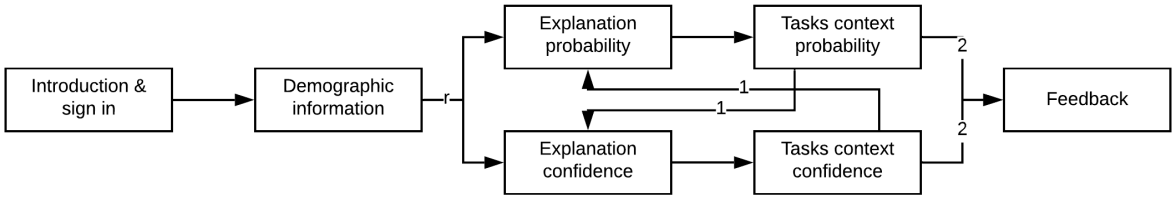


FIGURE 7.1: Overview of the structure of the user study

The experiment is taken in an online setting. Participants are asked to go to a website and sign in using their mail address and provide some demographic information: age, sex and profession. This is followed by a number of tasks to perform, with a final feedback page. As not all inspectors are proficient in English, the experiment is performed in Dutch. It was predicted that at least 25 people are necessary to participate in the study to determine significant results. This is determined by a two-sample one-sided test in power analysis. The analysis is described in Appendix C.5, both ad-hoc and post-hoc.

7.3.1 Inspection Task

The task the participant is asked to perform is deciding whether to inspect a certain inland ship based on the information provided with the explanation. This explanation looks the same but is from the two contexts discussed before. Besides the decision as to whether to inspect a ship, the participant can select which information is useful and if they trust in having made the correct decision. This task keeps the essence of the target application, however is simplified towards a binary choice whether to inspect a certain ship.

For each explanation based on confidence, the prediction of the base model is provided, followed by the confidence score of the Conformal Prediction framework. A plot showing the contribution of different features towards this confidence score is shown. This plot shows all features contributing to the prediction by SHAP values. With this specific problem a large number of features result in a large number of small contributions towards the prediction. It is not feasible to show all these contributions, as the amount of information that can be represented in an explanation is limited [24]. Therefore the ten features with the highest absolute SHAP value are selected with their feature value shown below the force plot.

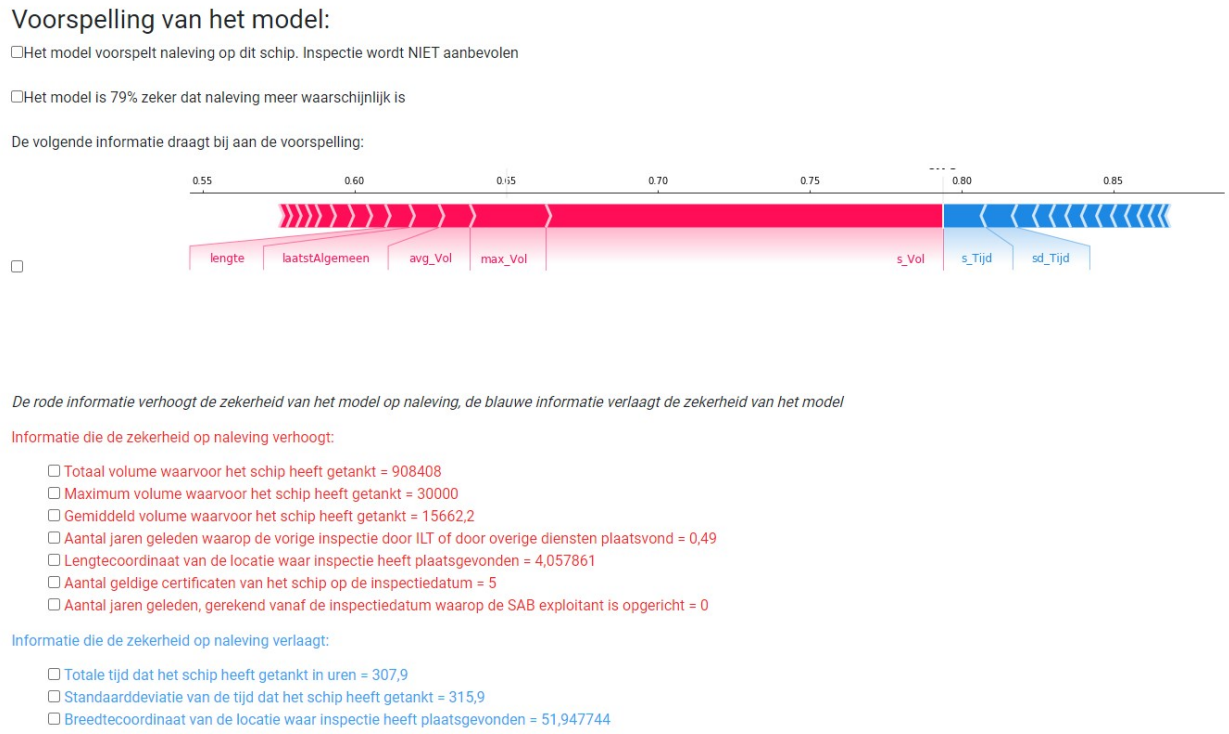


FIGURE 7.2: Interface of an explanation from the perspective of confidence

The explanations based on probability also show the prediction of the base model, followed by the probability of this base model. A plot showing the contribution of different features towards this probability score is shown, as well as the values of these features. Below the explanation four questions are asked to test the three hypotheses. These questions are shown in Figure 7.3.

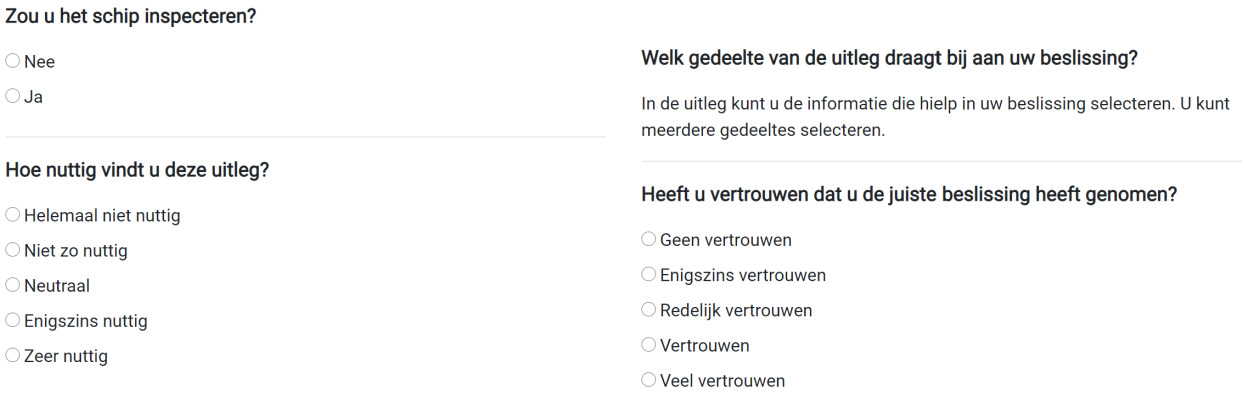


FIGURE 7.3: Questions asked in each task

7.3.2 Task ordering

A participant is shown a total of 20 ships which were previously inspected. One context is randomly selected for the first 10 tasks, with the 10 other tasks being presented from the other context. The ordering of the ships is based on the ordering in the test set, with the first ship assigned to a given participant being random. By not assigning a random ship each of the 20 instances we increase the coverage over both levels of the independent variable and the different ships. This is necessary due to the limited number of participants in the experiment. The reason for not alternating between the contexts is to avoid confusions as to which context is used, as the interface is almost identical.

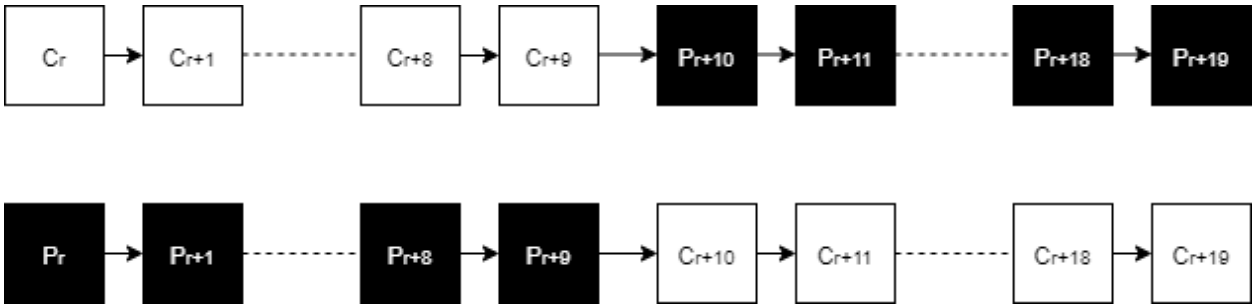


FIGURE 7.4: Procedure for a participant. P is an explanation based on probability, C is based on confidence and r is the random id of the first ship

The ordering makes it not possible for a participant to see a ship explained from both probability and confidence. By assigning the first ship r randomly to a participant, there are no groups of participants in this experiment. Grouping a limited number of participants could result in noise caused by the difference between individual participants.

7.3.3 Experiment details

Model used

For this experiment a random forest model was trained on the *inland* ship dataset provided by IDlab, explained in more detail in Section 2.5. The Conformal Prediction framework is fitted on this base model, with the non-conformity function based on the margin probability with a k-nearest neighbour normalization. The non-conformity function looks at the number of votes in the random forest as its distance measure. More details on these models are given in Section 6.2.

The performance of this classifier on this dataset is shown in Table 7.1

TABLE 7.1: Performance of the base model on the test set. Performed with 10-fold cross-validation

Accuracy	AUC	Precision
0.66	0.71	0.67

Instances used

With the relatively low performance of the classifier on the inland dataset, the ships included in this experiment are decided by ranking the instances in the test set. On the test set sorting based on both the probability and confidence are performed. The top 30 ships from the test set are taken, for which the accuracy is 66% and the precision is 70%. Instances are classified either way, not only the positive predicted classes were taken. Half of the ships are predicted to be in violation. This is therefore slightly different than the ranking performed in Chapter 5, where the focus was on positive instances (*violations*) only.

KernelSHAP was performed on both the probability and confidence values of these instances. Two explanations were generated for each ship; one explaining the probability of the random forest model, one explaining the confidence from the Conformal Prediction framework on top of the random forest model.

Participants

As discussed before, this experiment is modelled as a human-grounded evaluation. This allows the inclusion of other people besides the target user to participate in the user study. The larger pool of people in the context of evaluating the model predicting inland ships are people in the inspectorate who are somewhat knowledgeable on the issue of the inspections, but are not inspectors of inland ships. Examples are maritime inspectors or people from the IDlab who work on this specific problem. From the inspectorate 31 people participated in this experiment. These were all employees of the Inspectorate. A total of 12 inspectors participated who are specialized in the inspection of inland ships. A total of 7 inspectors from other fields, like maritime, participated as well. The other 12 participants are employees of the IDLab, which all have some knowledge about the problem setting of inspecting inland ships. The participants were not required to provide demographic information, however, most did (26). The average age of the participants is 45 years, with 81.8% males and 18.2% females.

Quality assurances

To get qualitative results of this user study, two checks are put in place. The first is a short explanation of probability and confidence before the respective tasks. Most of the participants have no knowledge of machine learning, without a basic understanding of the difference between the two contexts. A short introduction is therefore given with an example, along with two questions to test if the participant understands the explanation from a particular context. The second check is an additional random question in 10% of all tasks. This is a simple test of having to select the right box (*Please select "boat" from the list*). With this question we test if people are participating seriously and not just click through. Answering the question incorrectly is logged. Luckily, none of the participants in this small study answered this question incorrectly.

7.3.4 Results

In this section the methods and results of the experiment are discussed. This is achieved by determining if the hypotheses in Section 7.2.1 hold.

For all the hypothesis tests for Hypothesis 2 & 3 both a two-sample independent t-test and a Mann Whitney U Test is used. The t-test is chosen due to the large sample size and independent nature of the self-reported perceived usefulness and user trust. Likert scale values are ordinal values, not continuous values, and can therefore not be normally distributed. However, the t-test can still be used in hypothesis testing for Likert scale values [21]. When manually inspecting the histogram of the resulting perceived usefulness and user trust, a bell shape does indicate the distribution to be somewhat normally distributed. Parametric tests, such as t-test, are robust against non-normal distributions when the sample size is large enough due to the Central limit theorem, provided the distribution is truly normal [21]. With participants providing each 10 samples for each group, the sample size of a single group is 310. To be sure potential non-normality of the results does not influence the hypothesis testing, the nonparametric Mann Whitney U Test is also performed. The rejection of the null hypothesis for both hypothesis tests is the same for all situations, with similar p-values in most situations.

Hypothesis 1: Explanation effectiveness

To evaluate the explanation effectiveness, the decision of the participants of the experiment is compared with the true labels of the instances. The decision of the participants can be seen as a classifier, and metrics such as accuracy and precision can be determined. First, the performance for each completed task was compared between the context of *confidence* and *probability*. However, as the tasks for individual ships were not completed the same amount of times, difficult instances completed more than the average amount of completions could give biased results. Therefore the accuracy in the user decisions was also determined by first averaging the accuracy for each ship. The resulting accuracy with these two approaches is given in Table 7.2.

TABLE 7.2: The accuracy of human decisions comparing confidence and probability explanations

	Probability	Confidence	p-value
Global explanation	58.3%	59.1%	0.43
Average per ship	57.9%	61.9%	0.54

When averaging over all instances, there is a small difference in task accuracy between the context of *probability* and the context of *confidence*. The improvement in accuracy is greater when first averaging the instances per ship. The difference between the two contexts in both cases is not significant: $p = 0.43$ & $p = 0.54$. The Mann Whitney U test is performed for the accuracy without first averaging per ship. This is due to the discrete nature of the accuracy distribution, as an individual sample is a binary value expressing whether the participant made the correct decision. A Student's t-Test for related samples is performed when first averaging over all ships. The Student's t-Test was used as the accuracy was deemed to be normally distributed based on the Quantile-Quantile plots in Figure C.1.

We fail to reject the null hypothesis at a significance level of 0.05**.**

It should be noted that the human task accuracy is lower than the accuracy of the model itself. This means that the accuracy of the ultimate decision of the participant to inspect would be higher if

they simply always agreed with the prediction made by the model. This same result was found in several user studies by Poursabzi-Sangdeh et al. [71].

Comparing correctly and incorrectly labelled instances Another distinction can be made in the results, looking at the tasks with an incorrectly predicted class and tasks with a correctly predicted class. These results can determine if participants are able to determine if the model is correct in the case of correctly predicted classes. The results for these instances are shown in Table 7.3. In the case of incorrectly labelled instances, the accuracy can indicate if the participants are able to determine that the prediction is incorrect. The resulting accuracy for these instances is shown in Table 7.4.

TABLE 7.3: The accuracy of human decisions for tasks with correctly labelled instances

	Probability	Confidence	p-value
Global explanation	74.0%	76.1%	0.32
Average per ship	71.1%	76.5%	0.43

TABLE 7.4: The accuracy of human decisions for tasks with incorrectly labelled instances

	Probability	Confidence	p-value
Global explanation	28.0%	26.5%	0.42
Average per ship	29.1%	27.9%	0.89

The resulting accuracy is higher in the context of *confidence* for correct predictions, meaning that the participants agree more with the prediction made in this context. The improvement in accuracy is found for both the average over all the tasks and the average per ship. When the model made an incorrect prediction, the task accuracy is lower in the context of *confidence*. This is caused by the same reason as for the correct instances; a higher agreement with the prediction.

The difference in both cases is not significant for all four situations. The Mann Whitney U test is performed in the case of averaging over all samples, while a Student's t-Test for related samples is performed when first averaging over all ships.

The increase in agreement (75.6% vs 73.3%) results in the small improvement in accuracy in the context of *confidence*. In Appendix C the agreement in multiple situations is described.

The *task effectiveness* was not included in the ad-hoc power analysis in Appendix C.5, due to the nature of the distributions expressing the accuracy. However, with the results averaged per ship a post-hoc analysis was performed to determine the number of participants it would require to determine the improvement in *task effectiveness* found when using explanations from the context of *confidence*. From the results in Appendix C.5 it is found 196 participants are the minimum number required with the mean and standard deviation found.

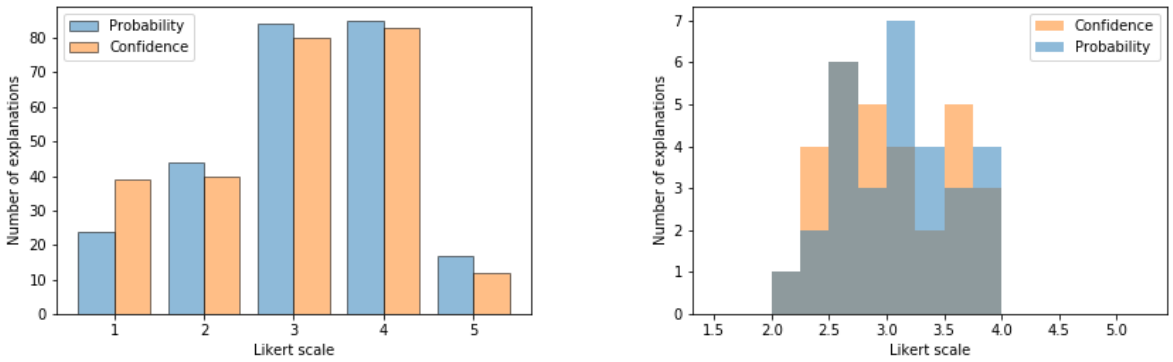
Hypothesis 2: Explanation usefulness

Hypothesis 2 in this experiment is evaluated by asking the participant their perceived usefulness of the explanation. This is done for each task according to a 5 point Likert scale. In Figure 7.5 the distribution of perceived usefulness of all the explanations is plotted. We want to determine whether explaining from the context of *confidence* results in a more perceived useful explanation.

As mentioned before, both a two-sample independent t-test and a Mann Whitney U test is used. The nonparametric hypothesis test results are given in Appendix C.

TABLE 7.5: Results for Hypothesis 2

	Probability		Confidence		p-value
	Mean	Std	Mean	Std	
All explanations	3.11	1.05	3.06	1.15	0.37
Positive explanations	3.11	1.12	2.86	1.16	0.08
Negative explanations	3.11	0.98	3.18	1.11	0.62
Correct explanations	**3.19**	1.04	**2.90**	1.15	**0.01**
Incorrect explanations	**2.96**	1.07	**3.26**	1.11	**0.04**



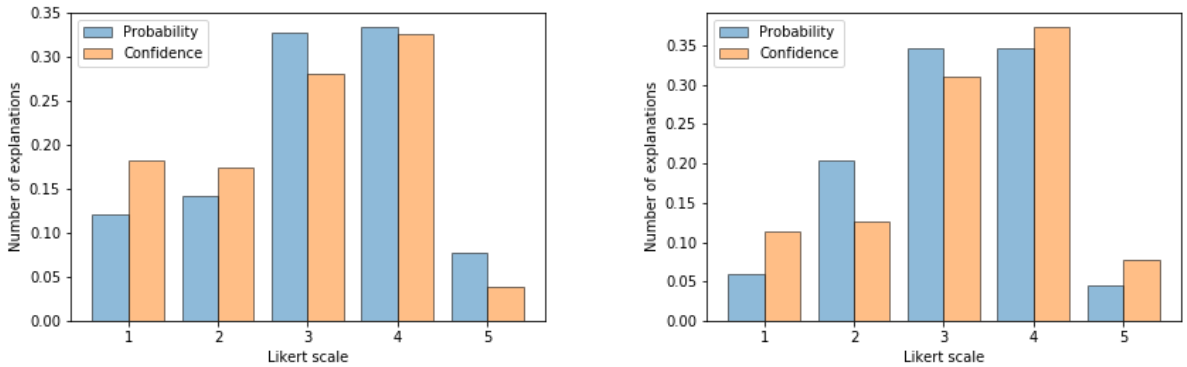
(A) Perceived usefulness of all tasks performed

(B) Average usefulness per ship

FIGURE 7.5: Perceived usefulness comparison between the two contexts

Figure 7.5a shows the perceived usefulness of all individual tasks performed. The tasks are separated by context. The difference in perceived usefulness differs only slightly; with a slight reduction in the perceived usefulness in the context of *confidence*, however, not significantly so ($p = 0.37$). In the context of *confidence*, there is a large percentage of *no perceived usefulness* when compared against the context of probability. This results in a higher standard deviation in the context of *confidence*.

In Figure 7.5b the perceived usefulness when first averaging by ship is shown. By doing so, the effect of certain ship tasks being performed more often is removed. The same decrease in perceived usefulness in the context of *confidence* was found.

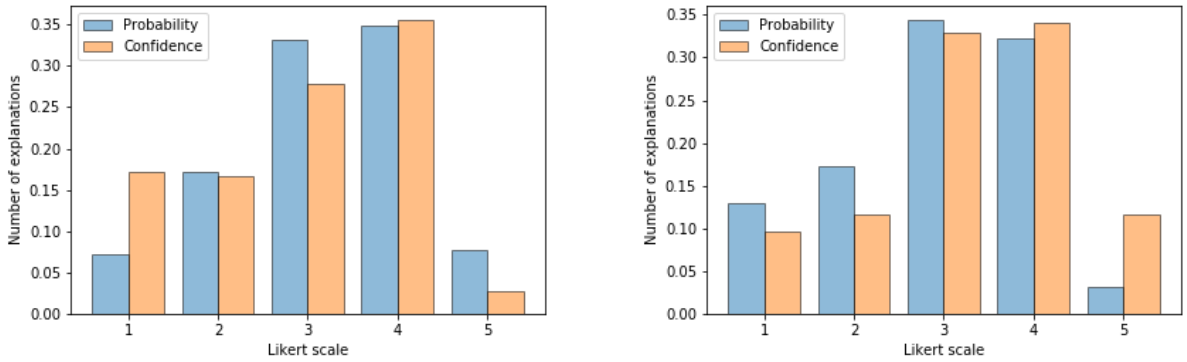


(A) Prediction of *violation*

(B) Prediction of *no violation*

FIGURE 7.6: Perceived usefulness of explanations separated by the prediction outcome

To determine if there is a difference in the perceived usefulness when looking at prediction, the results are separated by the binary prediction of *violation* or *no violation* (Figure 7.6). The perceived usefulness of the explanations of *violations* is improved, not significantly ($p = 0.08$), when explaining from probability. For explanations of *no violations* the average perceived usefulness is slightly improved in the context of *confidence*, also not significant ($p = 0.62$), with higher variability.



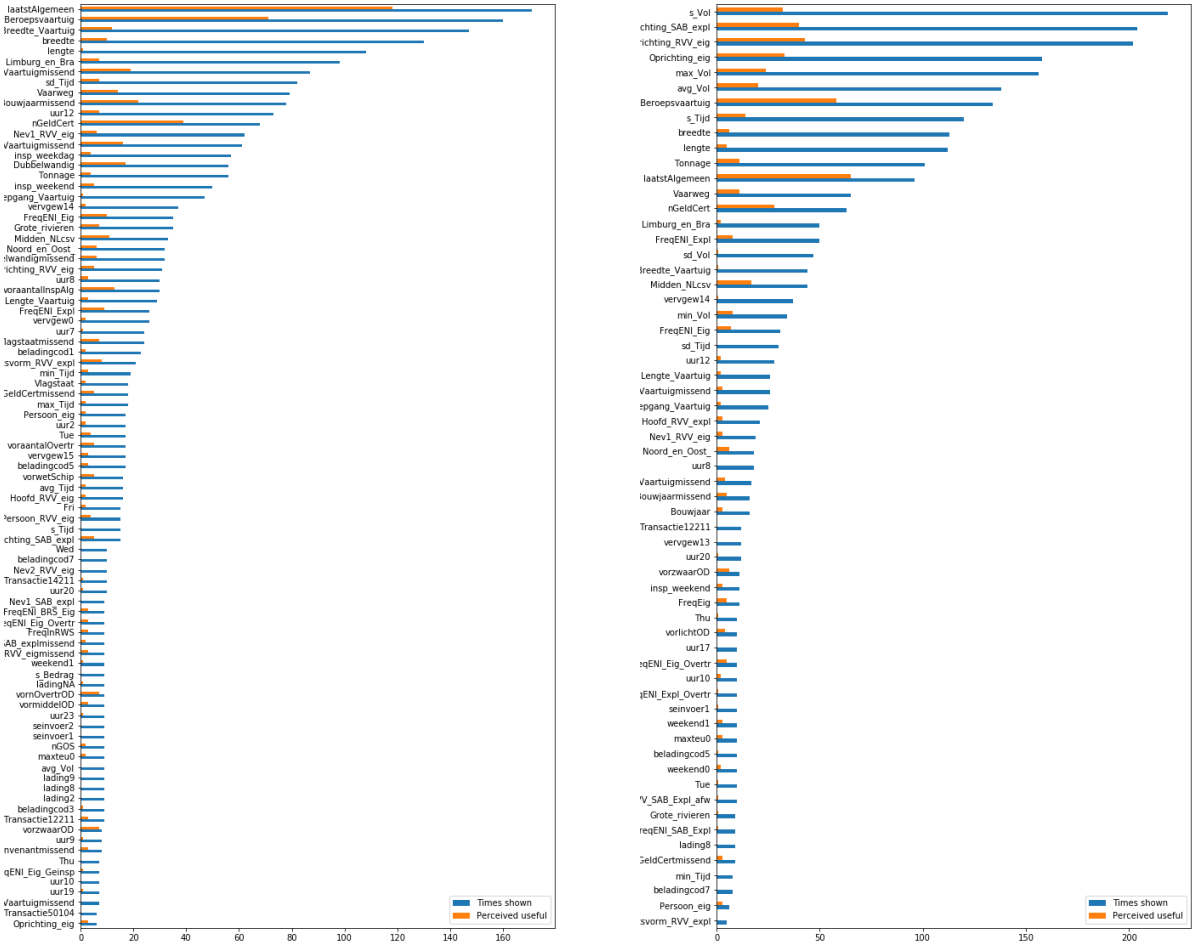
(A) Explanations of correct predictions

(B) Incorrect predictions

FIGURE 7.7: Perceived usefulness of explanations separated by the correctness of a prediction

The same evaluations are performed by separating based on the correctness of the predictions. Recall that the accuracy for the 30 ships is only 67%, so for a number of explanations the predictions are incorrect. The resulting perceived usefulness is significantly improved when explaining the context of *probability* when the prediction is *correct* ($p < 0.05$). The opposite results are found when the prediction is *incorrect*; the perceived usefulness for explanations from the context of *confidence* is significantly higher ($p < 0.05$).

Usefulness of features For each task a participant can select which information is useful for them in making their decision on whether or not to inspect a ship. The resulting self-reported useful features are given in Figure 7.8, along with the number of times the feature was shown in the experiment. In Appendix C this figure is represented as the percentage of times shown in Figure C.2. Each participant completes 10 tasks of either context, each showing 10 features most influencing the prediction. The number of features shown in either context is therefore the same. Comparing the total amount of selected features in the experiments is 579 (22.4%) in the context of *probability* versus 516 (20.9%) in the context of *confidence*. Two features are most useful in both contexts; The amount of time since the last inspection (*laatstAlgemeen*) and the type of ship (*soortBeroepsvaartuig*). These features were shown most often in the context of *probability*.



(A) Explanations of probability

(B) Explanations of confidence

FIGURE 7.8: Features reported to be useful

In Section 6.3, it is determined that for the problem of violations on inland ships, the context of *confidence* is influenced more by the fuel and owner information compared to the context of *probability*. This is also found when looking at the number of times these features were shown in this user study. The usefulness, as reported by the participants, for the fuel information is low compared to the number of times used to explain the prediction. The information about the owner is more useful to the participants of the experiment. These features are rarely shown in the context of probability, while being reported to be useful. Looking at the relationship between the perceived usefulness of features and the number of times the feature was shown, the features most often shown in the context of probability are also most useful to the participants for the two most useful features. The results suggest that the context of *probability* could more closely conform to human intuition of this problem. To further this conclusion, in Figure 7.9 a histogram is plotted showing how often each feature in the top 10 of most influencing features is useful to the participant. In the context of *probability* the two most influencing features are more often reported

to be useful, in the context of *confidence*, less influencing features in the top 10 are reported to be useful more often compared to the context of *probability*. For both contexts it is still the case that the features important for the model prediction, are in a lot of instances not important in the decision of the human expert. In other words, the importance of features does not align with the human intuition on the problem.

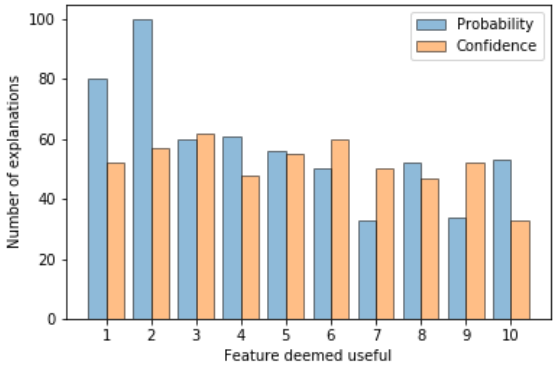


FIGURE 7.9: Most influencing features reported useful by participants

To conclude, we fail to reject the null hypothesis when looking at all 30 ships.

This means there is no significant difference in perceived usefulness between the context of *probability* and *confidence*. Also, when separating based on the label of the prediction, the null hypothesis is not rejected in both situations.

However, when separating based on the correctness of the prediction, the null hypothesis is rejected in both situations.

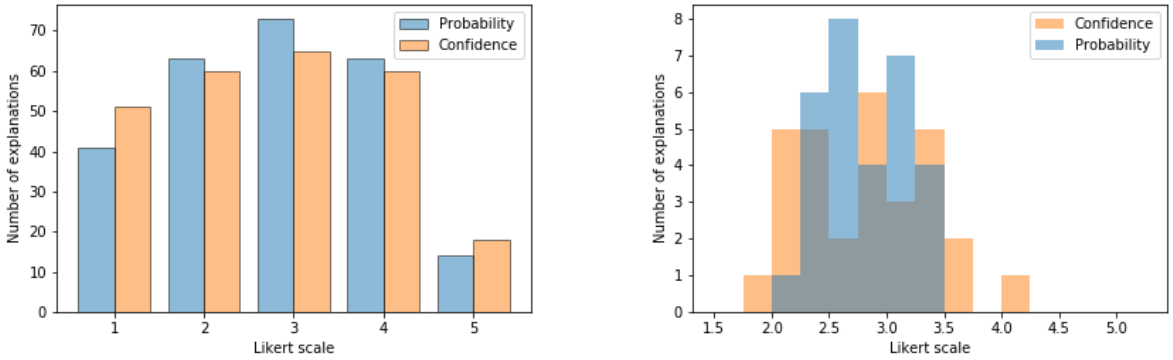
The perceived usefulness in the case of correct predictions is significantly higher when explaining *probability*, while the explanations of *confidence* result in significantly higher perceived usefulness when the prediction is incorrect. A possible explanation is in the determination of the confidence; the Conformal Prediction framework predicts the confidence by looking at the smaller p-value. This p-value belongs to the label not predicted. However, in the case of incorrect predictions, this is the correct label of the instance. When looking at the usefulness of individual features, it is found that two features are most useful in both contexts. The confidence context justifies predictions more often with fuel and owner information, with the owner information being useful to the participants.

Hypothesis 3: User trust

Hypothesis 3 in this experiment is evaluated by asking the participant the trust they have in making the correct decision. This is done for each task according to a 5 point Likert scale. We want to determine whether explaining from the context of *confidence* results the user trusting the explanation more in a certain context. The null hypothesis is tested with a two-sample independent t-test and a Mann Whitney U Test. There is no difference in the rejection of the null hypothesis between the two tests. The nonparametric hypothesis results are given in Appendix C. In Figure 7.5 a histogram of all self-reported user trust in each of the tasks is given.

TABLE 7.6: Results for Hypothesis 3

	Probability		Confidence		p-value
	Mean	Std	Mean	Std	
All explanations	2.80	1.13	2.81	1.23	0.91
Positive explanations	2.80	1.21	2.69	1.19	0.44
Negative explanations	**2.80**	1.06	**2.92**	1.26	0.37
Correct explanations	2.81	1.13	2.78	1.21	0.85
Incorrect explanations	**2.78**	1.14	**2.86**	1.27	0.66



(A) User trust of all tasks performed

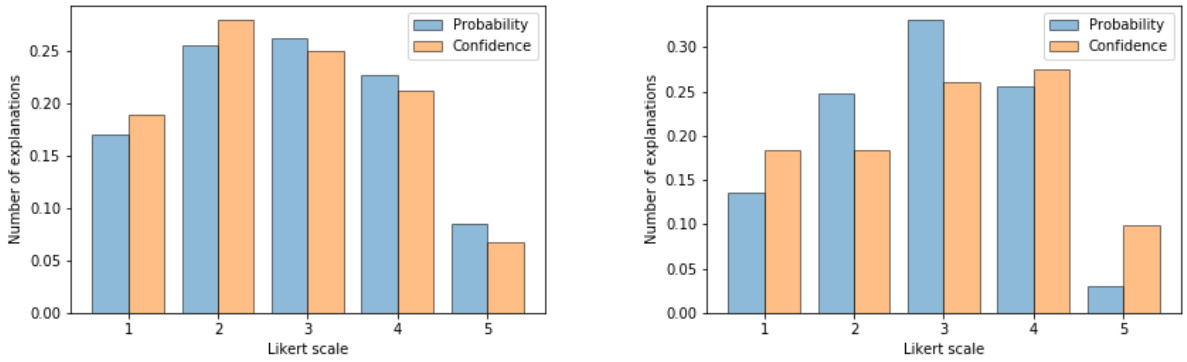
(B) Average user trust per ship

FIGURE 7.10: Perceived usefulness comparison between the two contexts

We fail to reject the null hypothesis at a significance level of 0.05**.**

Figure 7.10a shows user trust of all individual tasks performed. The tasks are separated by context. There is no significant difference in average user trust between contexts when looking at all explanations ($p = 0.91$). In the context of *confidence*, a larger number of users have either *no trust* or *total trust* when compared against the context of probability. This results in a higher standard deviation in the context of *confidence*. In Figure 7.10b the user trust is first averaged by ship. By doing so, the effect of certain ship tasks being performed more is removed. The same increase in standard deviation is found and no significant difference between the two contexts.

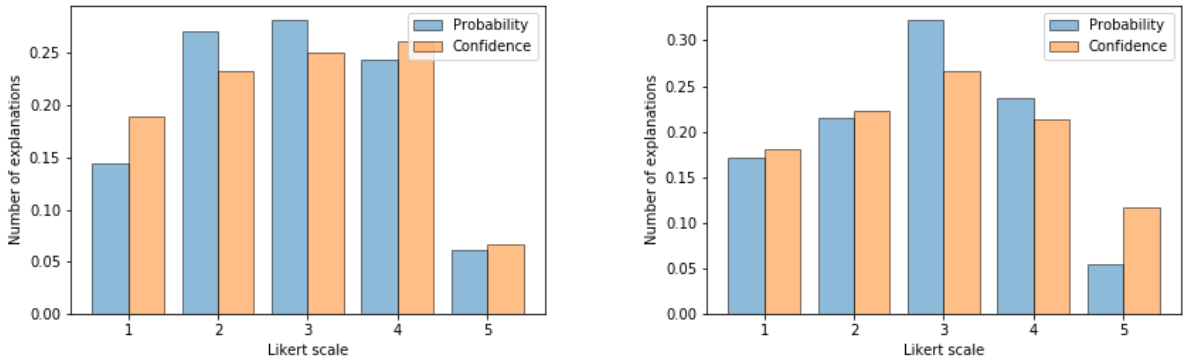
To determine if there is a difference in user trust between prediction outcome, the results are separated by the binary prediction of *violation* or *no violation* (Figure 7.11). User trust in the explanations of *violation* predictions is higher, not significantly ($p = 0.44$), when looking at the probability. For explanations of *no violation* predictions the average user trust is slightly improved, though not significantly ($p = 0.37$).



(A) Prediction of *violation* (B) Prediction of *no violation*

FIGURE 7.11: User trust of explanations separated by the prediction outcome

User trust is also evaluated when the prediction is correct, and when it is incorrect. The results are normalized, as the number of instances differs between the two contexts of *probability* and *confidence*. For both correct and incorrect predictions no significant difference in the user trust is found between the contexts ($p = 0.85$ & $p = 0.66$). The slight difference does mirror the results found in the perceived usefulness, where for explanations of incorrect predictions do have a higher user trust in the context of model confidence.



(A) Explanations of correct predictions (B) Incorrect predictions

FIGURE 7.12: User trust in explanations separated by the correctness of a prediction

With the resulting *perceived usefulness*, the context of *confidence* is *perceived* significantly less useful when the model is correct while significantly more useful when the model is incorrect. However, the resulting *user trust* indicates the participants were not more trusting of the decision made, even with an increase in *perceived usefulness*.

Results on individual inspection tasks

In the previous sections the resulting *task effectiveness*, *perceived usefulness* and *user trust* are shown. Next, we will look at how the difference in context impacts the individual inspection tasks. To achieve this, the results of the individual tasks are presented in Table 7.7. Here the tasks are sorted first by the *true* label, meaning whether or not the ship is actually in violation and then by the predicted label. This is followed by an average of all the different evaluation metrics per context.

From these results, the Pearson correlation between the different evaluation metrics is determined. Specifically, the difference of an evaluation metric between the two contexts is compared

TABLE 7.7: Results for individual inspection tasks

#	y_true	y_pred	Effectiveness		Usefulness		Trust	
			Probability	Confidence	Prob	Conf	Prob	Conf
1	0	0	60,00%	63,00%	2.9	2.25	2.8	1.75
4	0	0	50,00%	60,00%	3.5	3.4	3.25	3.1
6	0	0	71,00%	90,00%	2.14	2.9	2.29	3.0
10	0	0	100,00%	90,00%	3.78	3.3	3.2	3.4
13	0	0	100,00%	88,00%	3.67	3.75	3.1	4.25
14	0	0	91,00%	100,00%	3.36	3.83	2.6	3.5
24	0	0	75,00%	60,00%	2.88	3.0	3.13	2.7
29	0	0	56,00%	83,00%	2.67	2.5	2.33	2.25
30	0	0	33,00%	80,00%	3.22	2.7	3.0	2.2
2	0	1	22,00%	11,00%	3.33	3.67	3.0	3.33
17	0	1	21,00%	60,00%	2.43	2.8	2.79	2.6
23	0	1	33,00%	30,00%	3.22	2.9	2.67	2.2
27	0	1	50,00%	17,00%	2.5	2.83	3.25	2.42
3	1	1	89,00%	67,00%	3.22	2.11	3.0	2.22
5	1	1	56,00%	50,00%	2.89	2.3	2.6	2.5
7	1	1	0,00%	80,00%	2.33	2.8	2.0	2.6
8	1	1	29,00%	70,00%	3.29	3.0	2.86	3.0
9	1	1	86,00%	60,00%	2.86	2.8	2.57	2.8
11	1	1	100,00%	82,00%	3.75	2.45	3.25	2.82
15	1	1	100,00%	70,00%	3.3	3.7	2.7	3.2
16	1	1	100,00%	100,00%	3.92	3.83	3.17	3.67
18	1	1	92,00%	67,00%	3.67	3.17	2.92	3.0
19	1	1	92,00%	84,00%	3.08	2.67	2.67	2.67
28	1	1	44,00%	92,00%	2.44	2.75	2.33	2.42
12	1	0	20,00%	33,00%	3.2	3.78	3.1	3.7
20	1	0	20,00%	25,00%	3.1	4.0	2.7	3.5
21	1	0	11,00%	11,00%	2.56	3.89	2.22	3.33
22	1	0	25,00%	10,00%	2.75	2.7	2.25	2.1
25	1	0	12,00%	45,00%	2.88	2.73	2.75	2.91
26	1	0	75,00%	37,00%	3.88	3.55	3.13	2.82

with the difference of another evaluation metrics. The goal is to determine whether a difference in a metric between contexts translates into the same difference for another metric between the contexts. The correlation between the subjective metrics of *perceived usefulness* and *user trust* is found to be strong ($t = 0.68$ & $p = 4.0e - 5$). This indicates that if the explanation is perceived useful, the user has more trust in having made the right decision. Together with results of hypothesis 2 and 3, this suggests that a significant increase in the *perceived usefulness* of a certain context found for a specific situation is not enough to increase the user trust significantly.

The correlation is also determined between these subjective metrics and the objective metric of *task effectiveness*. There is no correlation between the *perceived usefulness* ($t = 0.23$ & $p = 0.21$) or *user trust* ($t = 0.15$ & $p = 0.43$) and *task effectiveness*. This indicates that an explanation deemed useful or increasing the user trust does not correlate with higher *task effectiveness*.

Written reflections of the participants

After the performing of the 20 tasks participants were able to provide feedback about the experiment. Participants were given some pointers about the kind of feedback we wish to receive via a presentation before the experiment took place. This was about the understanding of the explanation and the features shown in relation to the current working of the inspectors.

Based on manual inspection of the feedback, it is clear that the features are often not directly translatable to the decision of whether to inspect a ship. A selected of the feedback is described next. The original feedback is all in Dutch and is translated into English.

"Fuel variables are difficult to interpret, even more so when for example the total amount of times fuelled up a certain hour are shown, without the total times of other hours."

A participant noted that the fuel information in the context of *confidence* does not help him in making the decision, even more so if only a single feature is shown.

"I do not see a relation between the prediction and the factors influencing this prediction."

Another participant noted that he could not see any relation between the features shown and the decision on whether to inspect a ship. This further confirms that the feature importance of the model does not align with the human-intuition of the problem.

"Some of the ships I recognized from the features shown, this influenced my decision."

An inspector of inland ships noted that sometimes a feature was useful, insofar the feature made it possible to determine the ship in question. Based on previous knowledge he could make an informed decision. Other feedback is more on possible improvements in the presentation of features. For example, the location coordinates should be presented on a map, features encoded are sometimes shown multiple times and only the feature value is not enough; instead, if the feature value is high or low is more useful information. The difference in confidence and probability is largely understood, however, it in itself did not influence the decision. Only the difference in features shown was determinant of the decision, according to the written feedback. With all the written feedback, we conclude that in general there is still much improvement to be had in explaining the predictions of violations on inland ships. Participants mentioned that the features shown did not adequately explain the reasoning for a certain prediction.

7.4 Conclusion

The goal of this chapter was answering the last research question:

How are explanations based on confidence received by users?

This is achieved with a human grounded evaluation of the explanations in two different contexts. A user study is performed on the real-world problem of predicting violations on inland ships. A within-subject design is used, allowing participants to test both contexts.

The user experiment yielded several interesting results. Firstly, the results suggest an increase in *task effectiveness* when explaining confidence compared to explaining probability, however not significantly. The cause of this increase is a larger agreement between the participants' decision and the prediction of the base model. In a post-hoc power analysis the number of participants required to determine significance is estimated.

No significant difference in user trust is found between the explanations based on probability and the explanations based on confidence. This was the case for all situations evaluated in this study.

There also is no significant difference between the two contexts when looking at the overall *perceived usefulness* across all tasks. However, significant improvements can be found when separating the explanations by the correctness of the prediction; explanations of probability result in higher perceived usefulness for correct explanations while explanations of confidence improve the perceived usefulness for incorrect explanations. The features selected to be useful in the context of probability is slightly higher, with the most selected features also being higher in the top 10 of features. However, for both contexts the features important for the prediction often does not align with human-intuition of the problem.

The self-reported usefulness of specific features is also determined. The features reported to be useful is slightly higher in the context of probability and the most useful features, as reported by the participants, is more often most influencing the prediction of the base model. Explanations of confidence show more often fuel information and owner information. These features are also perceived to be useful information by the participants, however, these are not present in the explanations based on probability.

Based on the resulting evaluation metrics of the individual task, a strong correlation is determined between the *perceived usefulness* and *user trust*. So, while a number of situations lead to significant difference in the *perceived usefulness*, this is not enough to change the *user trust* significantly also. Both the subjective metrics are not correlated with the objective metric of *task effectiveness*, suggesting that participants' perceived usefulness and trust in their decision does not translate to a difference in *task effectiveness*.

Chapter 8

Conclusions

8.1 Summary

In this thesis we have looked at an approach using model confidence to select and interpret the predictions made by complex machine learning models. The goal was answering the following research question:

How can we predict confidence of a complex model to select predictions and provide inspectors with local model-agnostic explanations of this confidence?

In the first part of this thesis, the confidence in a prediction of any machine learning model is determined using the Conformal Prediction framework. The base machine learning model is trained in such a way that only the features of the test instance are needed to determine the probability of a class. The Conformal Prediction framework, additionally, looks at how much the test instance *conforms* to the training data in the probability space, to determine the quality measure of confidence. The behaviour of this confidence across different significance levels is shown. It is found that selecting the instances with high confidence improves performance relative to the global performance of the model. Furthermore, the class prediction of the Conformal Prediction framework was evaluated against the classification of the base model. The predictions by the base machine learning model were found to outperform the classification by the Conformal Prediction framework.

After the initial experiments with confidence, the problem of selecting predictions was modelled as a ranking problem. By determining the precision at the top of the list of predictions, the usefulness of including model confidence is evaluated. The baseline in the experiment is the sorting by the probability determined by the base machine learning model. Using the confidence measure to re-rank the ranking based on probability does not improve ranking when looking at complex tree ensemble methods. However, for simpler machine learning models the addition of confidence does improve the ranking of predictions. As the complex models are able to determine the *true* probabilities more accurately, the ranking with these probabilities is closer to the optimal ranking. The additional confidence measures do not improve the ranking in these cases. In the simpler models the approximations of the probabilities are more rudimentary, therefore an additional measure of confidence can improve the ranking.

In the second part of this thesis, an approach to explaining the context of confidence is described and evaluated. The explanation approach chosen is the SHAP framework, which determines feature contributions to justify the confidence estimation of an individual prediction. The SHAP framework is the only additive feature attribution method with the desirable property of *local accuracy* and *consistency*.

A user study is performed to determine whether explaining from the context of confidence results in higher quality explanations. These explanations are compared against the traditional

approach of explaining the probability of a prediction. We have compared the *task effectiveness*, *perceived usefulness* and *user trust*. In a few situations significant differences between the two contexts are found.

The *task effectiveness* in the context of confidence is slightly higher than the context of probability. The reason for this increase is the increased agreement in the prediction by the user. This results in a higher accuracy in the instances where the model is correct, and a lower accuracy for the instances where the model is incorrect. The overall accuracy in both contexts was lower than the accuracy of the model itself, meaning that always agreeing with the model would result in a higher accuracy. A post-hoc analysis is performed to determine the additional number of participants necessary to confirm that the improvement is significant. The perceived usefulness in the context of confidence is significantly higher when the model is incorrect, while when correct the perceived usefulness is higher in the context of probability. The same results were found when looking at users' trust, however, for this metric not significantly so.

8.2 Contributions and recommendations

Two main contributions are made in this thesis, from which a number of general recommendations are made:

- A novel approach for bipartite ranking on any machine learning dataset by incorporating confidence expressed by conformity. Improvements in the ranking are found for simpler machine learning models. Therefore, when choosing simpler machine learning models due to the increased transparency, the inclusion of confidence in the ranking of the instances could be worthwhile.
- A novel explanation approach from the context of confidence. An increase in agreement with the explanations is found compared against the traditional context of probability. Currently, inland ship inspectors find violations less than half of the time, while the accuracy of the model used is higher than 65%. A higher agreement in this particular case therefore improved the *task effectiveness*, though not significantly. More generally, explanations from the context of confidence can be useful when the machine learning model outperforms the user in making a prediction. *Perceived usefulness* of explanations from the context of confidence is also significantly improved for the instances the model is incorrect. The context of confidence can therefore be particularly useful when the accuracy of the base model is not high.

8.3 Conclusion

With the availability of vast amounts of data nowadays, using machine learning in support of human-decision problems is increasingly popular. An enormous amount of models are available, where the performance is most often measured with the accuracy in the prediction. However, not for every problem this is the most important metric of evaluation. For some problems, the global accuracy or precision is not as important. Instead, the relative performance of a selection of instances is most important. This is problem is a fundamental one in the field of Information Retrieval. However, for other problem settings, such as the inspections of inland ships, ranking based on more than just sorting by probability is not commonly done. By determining the quality of a prediction as the model confidence, an improvement in precision is found in a number of situations over the basic sorting based on probability. The improvement is most often found for the simpler *interpretable* models. Using these models is preferred over complex models when overall

accuracy is similar, due to the decrease in computational complexity and increase in interpretability.

For noisy and sparse real-world data, complex models often have the highest overall performance. These complex machine learning models are being used for more and more real-world applications, with people using these models having little understanding of the inner workings or reasoning of the model. This makes explaining either the models themselves or the output of these models a topic with increased interest. Only simpler *interpretable* models can themselves be easily explained. Therefore, justifying the output of a complex model is a more common approach for complex models, where the output is a probability. By determining the confidence in the correctness of the predictions, another context or output for explaining the prediction is possible. Evaluating explanations is challenging and not researched extensively. One may assume that showing features contributing to an output is fundamentally useful. However, as shown in this thesis, this is certainly not always the case, with features important for a complex model not aligning with the human intuition of the problem. Explaining from the context of confidence did result in certain benefits over explaining the standard probability.

By explaining from the context of confidence, the results indicate a potential higher agreement with the prediction results in an improvement in the *task effectiveness* for the human decision on inspecting inland ships in the Netherlands. When the model made an incorrect prediction, the explanation from the context of confidence was perceived to be more useful. Using confidence to explain prediction can therefore be useful when the overall accuracy of the model is not high. The increase perceived usefulness for these situations did not lead to an increase in user trust.

8.4 Limitations and Future Work

In the next section several limitations of this work are described, together with possible future directions for research.

Conformal Prediction

Due to only looking at binary classification problems, the behaviour of confidence in the Conformal Prediction framework for multi-class situations is not tested. Prediction sets of size 1 are less likely to occur for problems with many labels, resulting in either selecting only a small number of instances (having a high rejection rate [81]) or having more prediction of multiple labels. Regression problems are also compatible with the Conformal Prediction framework, and researching if the conclusions found in this thesis holds for these problems could be interesting.

For the evaluation of the confidence and prediction of the Conformal Prediction framework for a large number of classifiers and datasets, a model-agnostic non-conformity function is used. It would be interesting to evaluate the behaviour of the additional metrics with model-specific non-conformity functions.

Ranking with confidence

To limit the size of the thesis, not all combinations of probability, credibility and confidence was evaluated as a ranking problem. When looking at the correlation between these combinations and the error rate in a number of situations, different combinations are strongest correlated. In this thesis it is found that the inclusion of the quality measure of confidence helps most for simpler models. However, between combinations the best performing differs between classifiers and between datasets.

Furthermore, the absence of an improvement in ranking when looking at the ensemble methods and datasets used during this research does not mean the inclusion of confidence cannot

improve ranking when using these ensemble models. Future research could look into datasets where these models are not able to determine the *true* probabilities as accurately. Re-ranking with confidence could improve ranking in these instances.

Evaluation explanation

The results found in the user study suggest that while the explanation based on different contexts are different, in so far that the features shown are different, the user perception is in most situations not significantly different. A possible reason for this is specific to the problem of violations on inland ships; the features, in general, do not conform to the human intuition of the inspectors. Based on the written and oral feedback, inspectors decide based on more subjective markers or specific situations not captured in the features. Another possible reason for the small difference between the two contexts would be the similarity in the user interface. With the same user interface, it could cause the difference in the contexts to be unclear and result in the perceived usefulness or user trust to be the same. In follow up research, it would be interesting to see if more differentiating interfaces specific to the confidence determined by the Conformal Prediction framework would improve the results more significantly.

A specific use case for the Conformal Prediction framework in relation to explaining complex models is using the non-conformity function in prototyping. As discussed in the literature review, this is an approach in the field of XAI where similar instances in the training data are used to justify the prediction of the test instance. In current research, this is most often based on some measure of similarity based on the features of the instances. However, with the Conformal Prediction framework it would be possible to determine similarity in the probability space of the model. An example would be showing instances in the training data which results in the highest number of identical paths in the individual decision trees in a random forest model. Similarly, it would be interesting to research if the conformity measure can be used in conjunction with case-based reasoning explanation approaches. Case-based reasoning means using old experiences to understand and solve new problems. This approach determines similar instances already seen by the model to adapt to the new test instance [42]. This technique is used in the field of XAI to justify complex models, an example being [48]. The retrieval of similar instance can in this method also be tested with the non-conformity measures.

A limitation of the evaluation of the explanation method is the determining of useful features and how this relates to human intuition. In the user study participants were asked to select the features they found useful. However, in general, the features were not selected often. To better determine if the features most important to the model are also most useful for the user, a follow-up study could look at determining more generally the features deemed useful by the users. Instead of asking users whether the features important for the model are useful, users can select from the whole list features one time, which features are important for them in making their decision. Determining how this perceived usefulness correlates with the feature importances of the model could give valuable insights into the perceived quality of explanation. These insights into the human intuition of the problem could be used in a feedback loop to the model. An example of such a feedback loop based on the human in the loop approach specifically for explanations of complex models is presented in [28].

Bibliography

[1] Ashraf Abdul et al. "Trends and trajectories for explainable, accountable and intelligible systems: An hci research agenda". In: *Proceedings of the 2018 CHI conference on human factors in computing systems*. 2018, pp. 1–18.

[2] Amina Adadi and Mohammed Berrada. "Peeking inside the black-box: A survey on Explainable Artificial Intelligence (XAI)". In: *IEEE Access* 6 (2018), pp. 52138–52160.

[3] Shivani Agarwal. *A study of the bipartite ranking problem in machine learning*. Tech. rep. 2005.

[4] Shivani Agarwal and Partha Niyogi. "Stability and generalization of bipartite ranking algorithms". In: *International Conference on Computational Learning Theory*. Springer. 2005, pp. 32–47.

[5] Kellie J Archer and Ryan V Kimes. "Empirical characterization of random forest variable importance measures". In: *Computational Statistics & Data Analysis* 52.4 (2008), pp. 2249–2260.

[6] Alejandro Barredo Arrieta et al. "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI". In: *Information Fusion* 58 (2020), pp. 82–115.

[7] David Baehrens et al. "How to explain individual classification decisions". In: *Journal of Machine Learning Research* 11.Jun (2010), pp. 1803–1831.

[8] Ram B Basnet and Andrew H Sung. "Classifying phishing emails using confidence-weighted linear classifiers". In: *International Conference on Information Security and Artificial Intelligence (ISAI)*. 2010, pp. 108–112.

[9] Siddhartha Bhattacharyya. "Confidence in predictions from random tree ensembles". In: *Knowledge and information systems* 35.2 (2013), pp. 391–410.

[10] Or Biran and Courtenay Cotton. "Explanation and justification in machine learning: A survey". In: *IJCAI-17 workshop on explainable AI (XAI)*. Vol. 8. 2017, p. 1.

[11] BlastChar. *Telco Customer Churn*. 2018. URL: https://www.kaggle.com/blastchar/telco-customer-churn.

[12] Olcay Boz. "Extracting decision trees from trained neural networks". In: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2002, pp. 456–461.

[13] Leo Breiman. "Random forests". In: *Machine learning* 45.1 (2001), pp. 5–32.

[14] Christopher JC Burges. "From ranknet to lambdarank to lambdamart: An overview". In: *Learning* 11.23-581 (2010), p. 81.

[15] Tianqi Chen and Carlos Guestrin. "Xgboost: A scalable tree boosting system". In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016, pp. 785–794.

[16] Sergio Cleger-Tamayo et al. "Being confident about the quality of the predictions in recommender systems". In: *European Conference on Information Retrieval*. Springer. 2013, pp. 411–422.

[17] John W Coulston et al. "Approximating prediction uncertainty for random forest regression models". In: *Photogrammetric Engineering & Remote Sensing* 82.3 (2016), pp. 189–197.

[18] Koby Crammer, Mark Dredze, and Fernando Pereira. "Confidence-weighted linear classification for text categorization". In: *Journal of Machine Learning Research* 13.Jun (2012), pp. 1891–1926.

[19] Koby Crammer, Alex Kulesza, and Mark Dredze. "Adaptive regularization of weight vectors". In: *Advances in neural information processing systems*. 2009, pp. 414–422.

[20] Mark Craven and Jude W Shavlik. "Extracting tree-structured representations of trained networks". In: *Advances in neural information processing systems*. 1996, pp. 24–30.

[21] JFC De Winter and Dimitra Dodou. "Five-Point Likert Items: t test versus Mann-Whitney-Wilcoxon (Addendum added October 2012)". In: *Practical Assessment, Research, and Evaluation* 15.1 (2010), p. 11.

[22] Houtao Deng. "Interpreting tree ensembles with intrees". In: *International Journal of Data Science and Analytics* 7.4 (2019), pp. 277–287.

[23] Dmitry Devetyarov and Ilia Nouretdinov. "Prediction with confidence based on a random forest classifier". In: *IFIP International Conference on Artificial Intelligence Applications and Innovations*. Springer. 2010, pp. 37–44.

[24] Finale Doshi-Velez and Been Kim. "Towards a rigorous science of interpretable machine learning". In: *arXiv preprint arXiv:1702.08608* (2017).

[25] Mark Dredze, Koby Crammer, and Fernando Pereira. "Confidence-weighted linear classification". In: *Proceedings of the 25th international conference on Machine learning*. ACM. 2008, pp. 264–271.

[26] Mengnan Du, Ninghao Liu, and Xia Hu. "Techniques for interpretable machine learning". In: *arXiv preprint arXiv:1808.00033* (2018).

[27] *Feature importances with forests of trees*¶. URL: https://scikit-learn.org/stable/auto_examples/ensemble/plot_forest_importances.html.

[28] Jean-Marc Fellous et al. "Explainable artificial intelligence for neuroscience: Behavioral neurostimulation". In: *Frontiers in Neuroscience* 13 (2019), p. 1346.

[29] Charles R Gallistel et al. "The perception of probability." In: *Psychological Review* 121.1 (2014), p. 96.

[30] Leilani H Gilpin et al. "Explaining explanations: An overview of interpretability of machine learning". In: *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*. IEEE. 2018, pp. 80–89.

[31] Riccardo Guidotti et al. "A survey of methods for explaining black box models". In: *ACM computing surveys (CSUR)* 51.5 (2018), pp. 1–42.

[32] Sigrid Møyner Hohle and Karl Halvor Teigen. "More than 50% or Less than 70% Chance: Pragmatic Implications of Single-Bound Probability Estimates". In: *Journal of behavioral decision making* 31.1 (2018), pp. 138–150.

[33] Mark Hopkins et al. "Spambase data set". In: *Hewlett-Packard Labs* 1.7 (1999).

[34] Ulf Johansson, Henrik Boström, and Tuve Löfström. "Conformal prediction using decision trees". In: *2013 IEEE 13th international conference on data mining*. IEEE. 2013, pp. 330–339.

[35] Ulf Johansson and Lars Niklasson. "Evolving decision trees using oracle guides". In: *2009 IEEE Symposium on Computational Intelligence and Data Mining*. IEEE. 2009, pp. 238–244.

[36] Ulf Johansson et al. "Model-agnostic nonconformity functions for conformal classification". In: *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2017, pp. 2072–2079.

[37] Andrej Karpathy, Justin Johnson, and Li Fei-Fei. "Visualizing and understanding recurrent networks". In: *arXiv preprint arXiv:1506.02078* (2015).

[38] Guolin Ke et al. "Lightgbm: A highly efficient gradient boosting decision tree". In: *Advances in neural information processing systems*. 2017, pp. 3146–3154.

[39] Majdi Khalid, Indrakshi Ray, and Hamidreza Chitsaz. "Confidence-weighted bipartite ranking". In: *International Conference on Advanced Data Mining and Applications*. Springer. 2016, pp. 35–49.

[40] Been Kim, Julie A Shah, and Finale Doshi-Velez. "Mind the gap: A generative approach to interpretable feature selection and extraction". In: *Advances in Neural Information Processing Systems*. 2015, pp. 2260–2268.

[41] Ron Kohavi. "Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid." In: *Kdd*. Vol. 96. 1996, pp. 202–207.

[42] Janet L Kolodner. "An introduction to case-based reasoning". In: *Artificial intelligence review* 6.1 (1992), pp. 3–34.

[43] Wojciech Kotlowski, Krzysztof Dembczynski, and Eyke Huellermeier. "Bipartite ranking through minimization of univariate loss". In: *ICML*. 2011.

[44] Josua Krause et al. "A workflow for visual diagnostics of binary classifiers using instance-level explanations". In: *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*. IEEE. 2017, pp. 162–172.

[45] R Krishnan, G Sivakumar, and P Bhattacharya. "Extracting decision trees from trained neural networks". In: *Pattern recognition* 32.12 (1999).

[46] Isaac Lage et al. "An evaluation of the human-interpretability of explanation". In: *arXiv preprint arXiv:1902.00006* (2019).

[47] Isaac Lage et al. "Human evaluation of models built for interpretability". In: *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*. Vol. 7. 1. 2019, pp. 59–67.

[48] Jean-Baptiste Lamy et al. "Explainable artificial intelligence for breast cancer: A visual case-based reasoning approach". In: *Artificial intelligence in medicine* 94 (2019), pp. 42–53.

[49] Tao Lei, Regina Barzilay, and Tommi Jaakkola. "Rationalizing neural predictions". In: *arXiv preprint arXiv:1606.04155* (2016).

[50] Jundong Li et al. "Feature selection: A data perspective". In: *ACM Computing Surveys (CSUR)* 50.6 (2018), p. 94.

[51] Henrik Linusson. *nonconformist*. 2017. URL: https://github.com/donlnz/nonconformist.

[52] Henrik Linusson et al. "On the calibration of aggregated conformal predictors". In: *The 6th Symposium on Conformal and Probabilistic Prediction with Applications,(COPA 2017), 13-16 June, 2017, Stockholm, Sweden*. 2017, pp. 154–173.

[53] Zachary C Lipton. "The mythos of model interpretability". In: *arXiv preprint arXiv:1606.03490* (2016).

[54] Tie-Yan Liu. *Learning to rank for information retrieval*. Springer Science & Business Media, 2011.

[55] Scott M Lundberg, Gabriel G Erion, and Su-In Lee. "Consistent individualized feature attribution for tree ensembles". In: *arXiv preprint arXiv:1802.03888* (2018).

[56] Scott M Lundberg and Su-In Lee. "A unified approach to interpreting model predictions". In: *Advances in Neural Information Processing Systems*. 2017, pp. 4765–4774.

[57] Justin Ma et al. "Identifying suspicious URLs: an application of large-scale online learning". In: *Proceedings of the 26th annual international conference on machine learning*. ACM. 2009, pp. 681–688.

[58] Sam J Maglio and Evan Polman. "Revising probability estimates: Why increasing likelihood means increasing impact." In: *Journal of personality and social psychology* 111.2 (2016), p. 141.

[59] Maciej A Mazurowski. "Estimating confidence of individual rating predictions in collaborative filtering recommender systems". In: *Expert Systems with Applications* 40.10 (2013), pp. 3847–3857.

[60] John M McGuirl and Nadine B Sarter. "Supporting trust calibration and the effective use of decision aids by presenting dynamic system confidence information". In: *Human factors* 48.4 (2006), pp. 656–665.

[61] Sean M McNee et al. "Confidence displays and training in recommender systems". In: *Proc. INTERACT*. Vol. 3. 2003, pp. 176–183.

[62] Ninareh Mehrabi et al. "A survey on bias and fairness in machine learning". In: *arXiv preprint arXiv:1908.09635* (2019).

[63] Nicolai Meinshausen. "Quantile regression forests". In: *Journal of Machine Learning Research* 7.Jun (2006), pp. 983–999.

[64] Thomas Melluish et al. "Comparing the Bayes and typicalness frameworks". In: *European Conference on Machine Learning*. Springer. 2001, pp. 360–371.

[65] Jianyu Miao and Lingfeng Niu. "A survey on feature selection". In: *Procedia Computer Science* 91 (2016), pp. 919–926.

[66] Harikrishna Narasimhan and Shivani Agarwal. "On the relationship between binary classification, bipartite ranking, and binary class probability estimation". In: *Advances in neural information processing systems*. 2013, pp. 2913–2921.

[67] Anna Palczewska et al. "Interpreting random forest classification models using a feature contribution method". In: *Integration of reusable systems*. Springer, 2014, pp. 193–218.

[68] Harris Papadopoulos. "A cross-conformal predictor for multi-label classification". In: *IFIP International Conference on Artificial Intelligence Applications and Innovations*. Springer. 2014, pp. 241–250.

[69] Andrea Papenmeier, Gwenn Englebienne, and Christin Seifert. "How model accuracy and explanation fidelity influence user trust". In: *arXiv preprint arXiv:1907.12652* (2019).

[70] Fabian Pedregosa et al. "Scikit-learn: Machine learning in Python". In: *the Journal of machine Learning research* 12 (2011), pp. 2825–2830.

[71] Forough Poursabzi-Sangdeh et al. "Manipulating and measuring model interpretability". In: *arXiv preprint arXiv:1802.07810* (2018).

[72] *President-Forecasting the US 2020 elections*. URL: https://projects.economist.com/us-2020-forecast/president.

[73] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should i trust you?: Explaining the predictions of any classifier". In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM. 2016, pp. 1135–1144.

[74] C. J. Van Rijsbergen. *Information Retrieval*. 2nd. USA: Butterworth-Heinemann, 1979. ISBN: 0408709294.

[75] Marko Robnik-Šikonja and Igor Kononenko. "Explaining classifications for individual instances". In: *IEEE Transactions on Knowledge and Data Engineering* 20.5 (2008), pp. 589–600.

[76] Philipp Schmidt and Felix Biessmann. "Quantifying interpretability and trust in machine learning systems". In: *arXiv preprint arXiv:1901.08558* (2019).

[77] Glenn Shafer and Vladimir Vovk. "A tutorial on conformal prediction". In: *Journal of Machine Learning Research* 9.Mar (2008), pp. 371–421.

[78] Lloyd S Shapley. "A value for n-person games". In: *Contributions to the Theory of Games* 2.28 (1953), pp. 307–317.

[79] Durga L Shrestha and Dimitri P Solomatine. "Machine learning approaches for estimation of prediction interval for the model output". In: *Neural Networks* 19.2 (2006), pp. 225–235.

[80] Slundberg. *slundberg/shap*. URL: https://github.com/slundberg/shap.

[81] Evgueni Smirnov, Nikolay Nikolaev, and Georgi Nalbantov. "Single-stacking conformity approach to reliable classification". In: *International Conference on Artificial Intelligence: Methodology, Systems, and Applications*. Springer. 2010, pp. 161–170.

[82] Evgueni N Smirnov, Georgi I Nalbantov, and AM Kaptein. "Meta-conformity approach to reliable classification". In: *Intelligent Data Analysis* 13.6 (2009), pp. 901–915.

[83] Carolin Strobl et al. "Bias in random forest variable importance measures: Illustrations, sources and a solution". In: *BMC bioinformatics* 8.1 (2007), p. 25.

[84] Carolin Strobl et al. "Conditional variable importance for random forests". In: *BMC bioinformatics* 9.1 (2008), p. 307.

[85] Erik Štrumbelj, Igor Kononenko, and M Robnik Šikonja. "Explaining instance classifications with interactions of subsets of feature values". In: *Data & Knowledge Engineering* 68.10 (2009), pp. 886–904.

[86] William R Swartout. "XPLAIN: A system for creating and explaining expert consulting programs". In: *Artificial intelligence* 21.3 (1983), pp. 285–325.

[87] Nava Tintarev and Judith Masthoff. "A survey of explanations in recommender systems". In: *2007 IEEE 23rd international conference on data engineering workshop*. IEEE. 2007, pp. 801–810.

[88] Gabriele Tolomei et al. "Interpretable predictions of tree-based ensembles via actionable feature tweaking". In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM. 2017, pp. 465–474.

[89] *Variable importance with random forests*. URL: https://www.rdocumentation.org/packages/randomForest/versions/4.6-14/topics/importance.

[90] Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.

[91] Stefan Wager, Trevor Hastie, and Bradley Efron. "Confidence intervals for random forests: The jackknife and the infinitesimal jackknife". In: *The Journal of Machine Learning Research* 15.1 (2014), pp. 1625–1651.

[92] Huazhen Wang et al. "Hedged predictions for traditional Chinese chronic gastritis diagnosis with confidence machine". In: *Computers in biology and medicine* 39.5 (2009), pp. 425–432.

[93] Jialei Wang, Peilin Zhao, and Steven CH Hoi. "Exact soft confidence-weighted learning". In: *arXiv preprint arXiv:1206.4612* (2012).

[94] Jun Wang. "Mean-variance analysis: A new document ranking theory in information retrieval". In: *European Conference on Information Retrieval*. Springer. 2009, pp. 4–16.

[95] Hilde JP Weerts, Werner van Ipenburg, and Mykola Pechenizkiy. "A Human-Grounded Evaluation of SHAP for Alert Processing". In: *arXiv preprint arXiv:1907.03324* (2019).

[96] H Peyton Young. "Monotonic solutions of cooperative games". In: *International Journal of Game Theory* 14.2 (1985), pp. 65–72.

[97] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks". In: *European conference on computer vision*. Springer. 2014, pp. 818–833.

[98] Guido Zuccon, Leif Azzopardi, and Keith van Rijsbergen. "Back to the roots: mean-variance analysis of relevance estimations". In: *European Conference on Information Retrieval*. Springer. 2011, pp. 716–720.

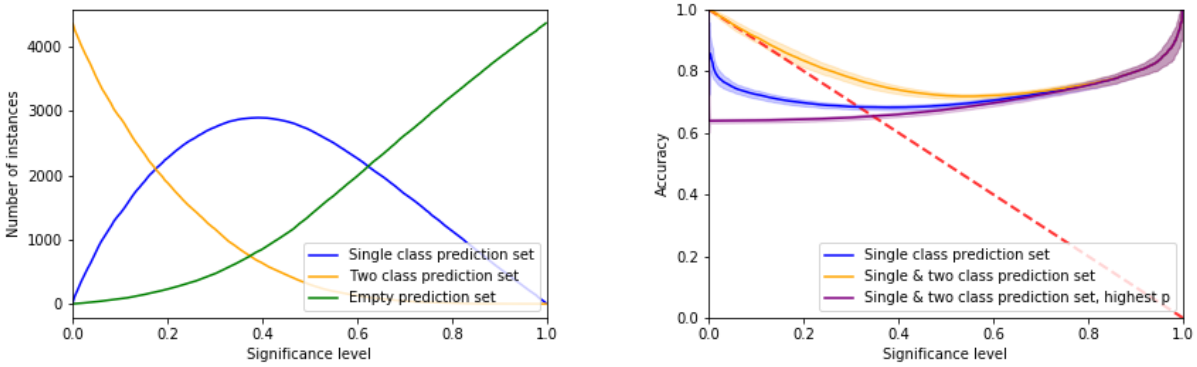
Appendix A

Determining confidence

A.1 Prediction set behaviour

In this section additional plots of the behaviour of the Conformal Prediction framework on different combinations of datasets and classifiers is given.

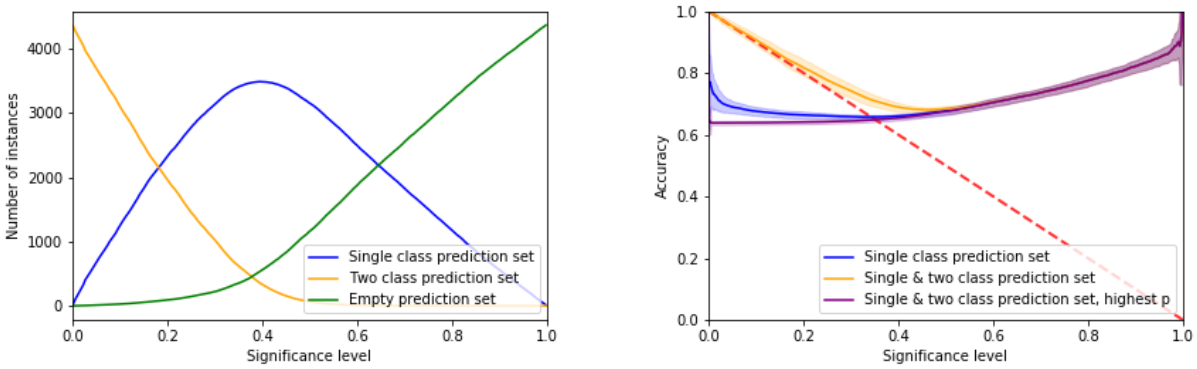
A.1.1 Inland dataset



(A) Size of different types of prediction sets

(B) Accuracy of different types of prediction sets

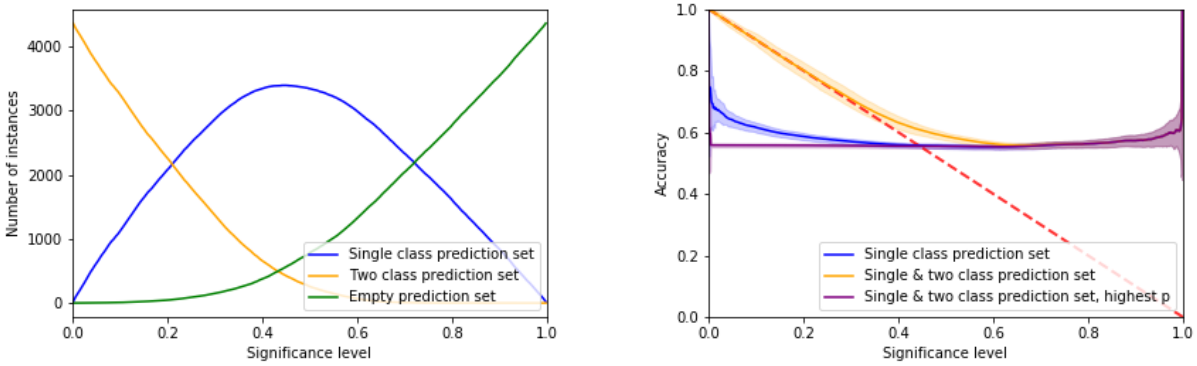
FIGURE A.1: Random Forest model



(A) Size of different types of prediction sets

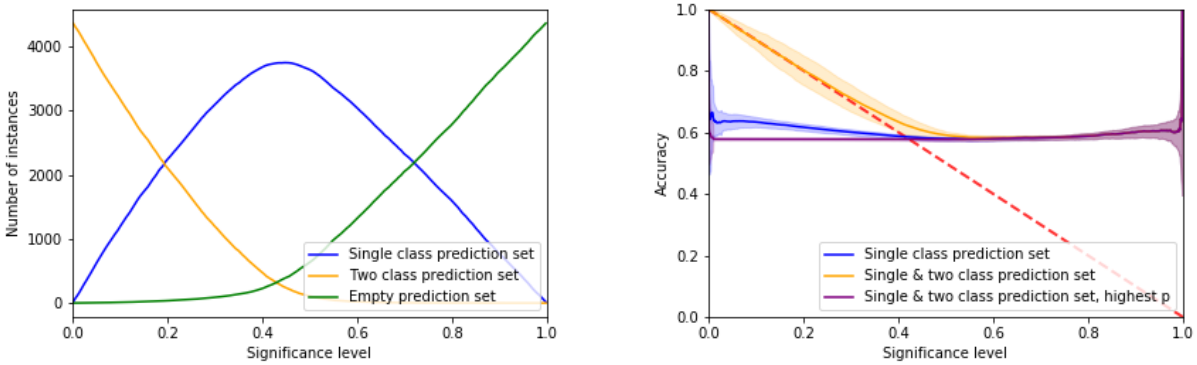
(B) Accuracy of different types of prediction sets

FIGURE A.2: XGBoost model



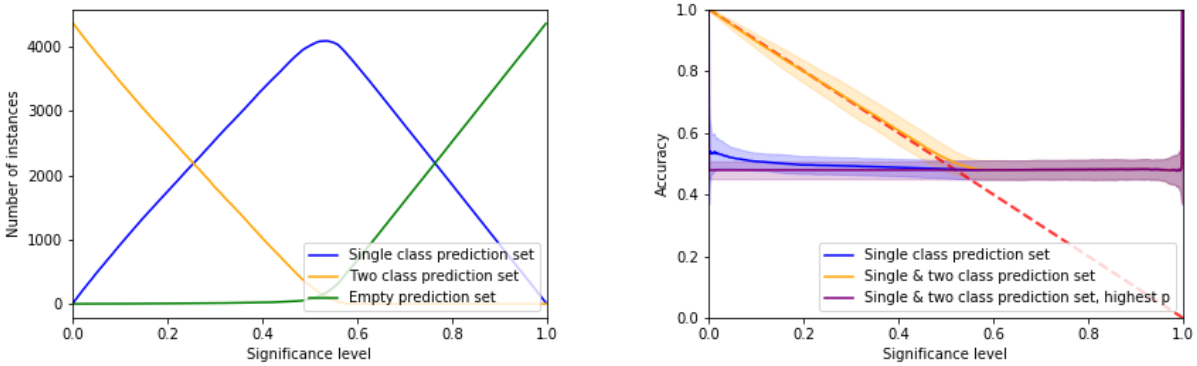
(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

FIGURE A.3: *k*-nearest neighbor model



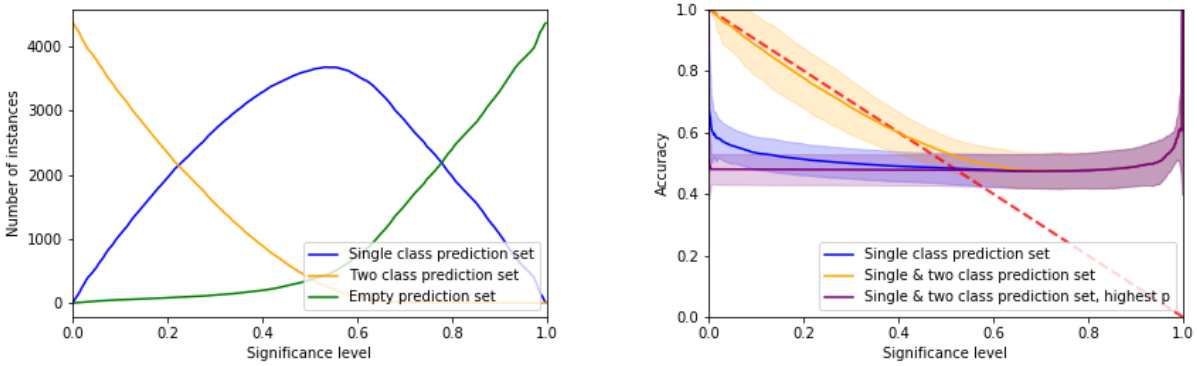
(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

FIGURE A.4: Logistic regression model



(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

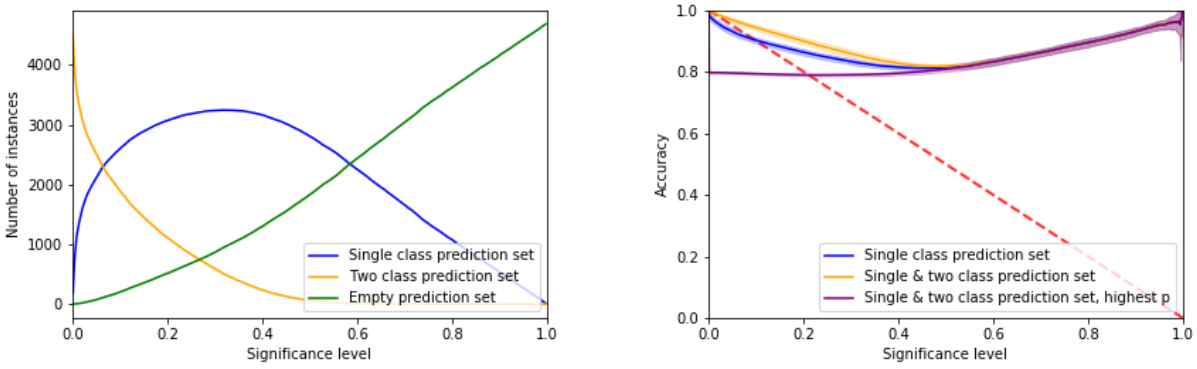
FIGURE A.5: Quadratic discriminant analysis model



(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

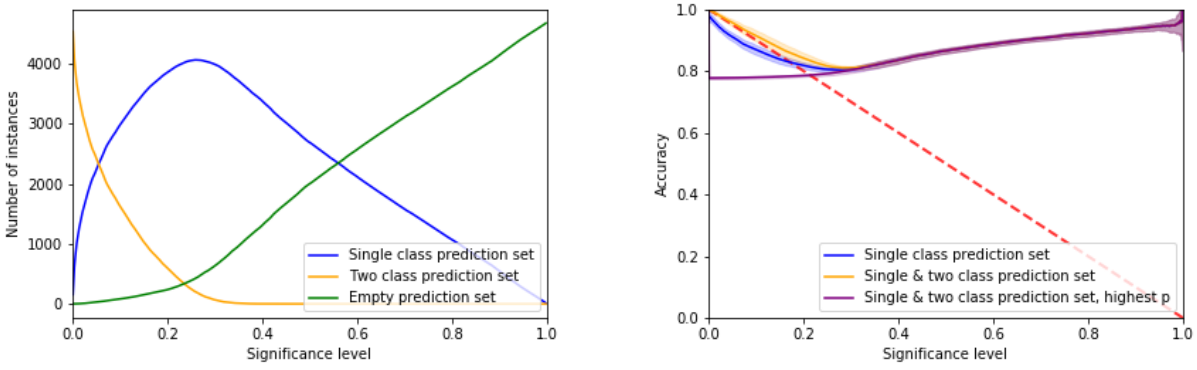
FIGURE A.6: Naive Bayes model

A.1.2 Churn dataset



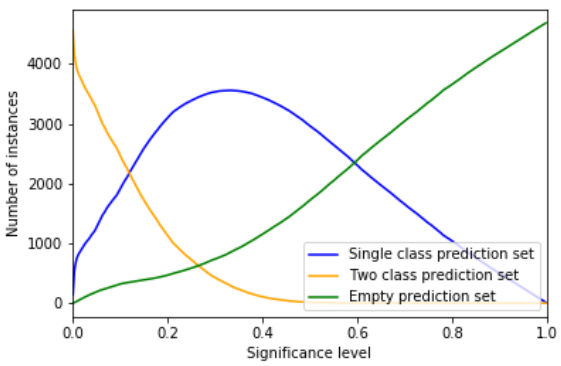
(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

FIGURE A.7: Random Forest model

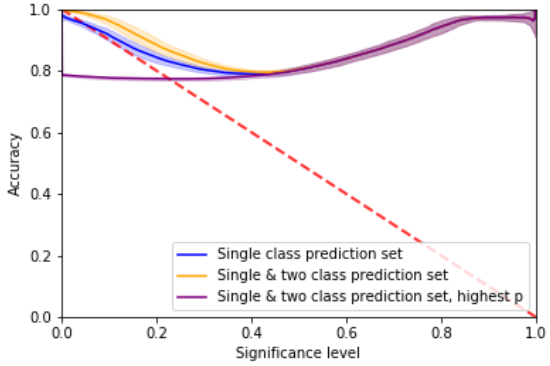


(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

FIGURE A.8: XGBoost model

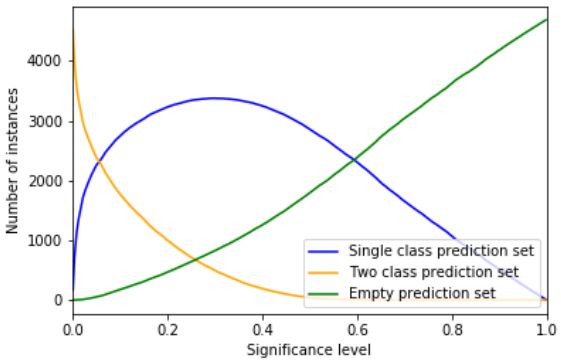


(A) Size of different types of prediction sets

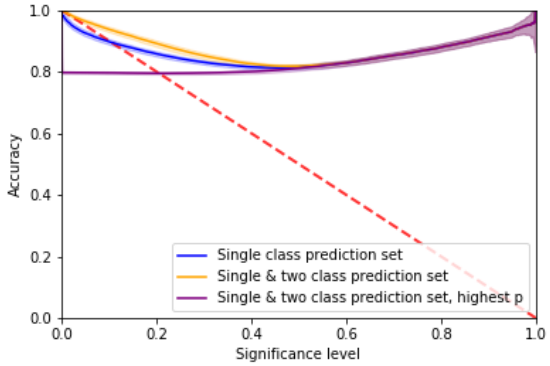


(B) Accuracy of different types of prediction sets

FIGURE A.9: *k*-nearest neighbor model

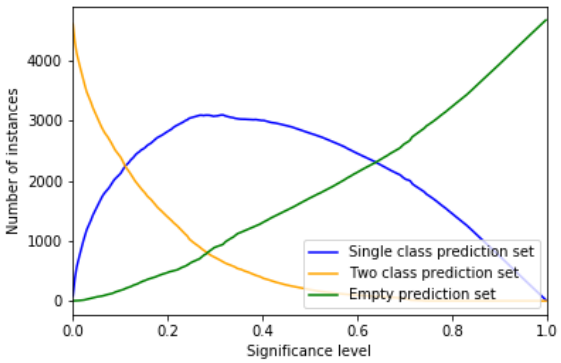


(A) Size of different types of prediction sets

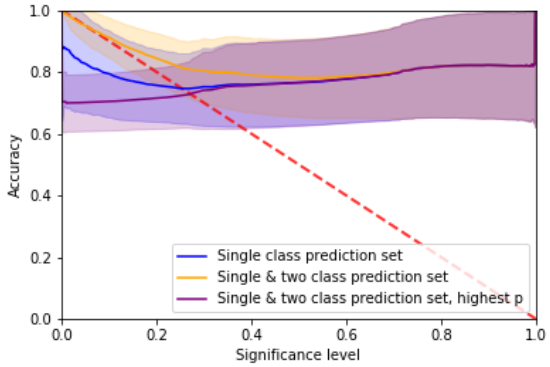


(B) Accuracy of different types of prediction sets

FIGURE A.10: Logistic regression model

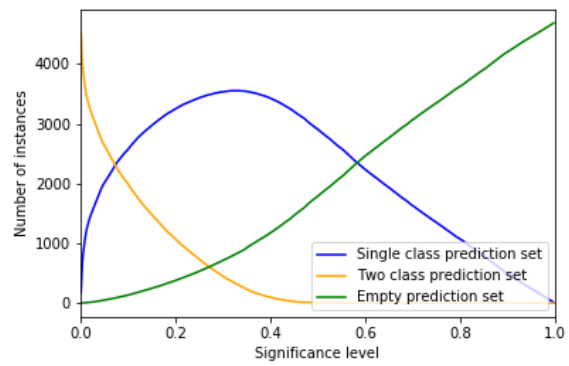


(A) Size of different types of prediction sets

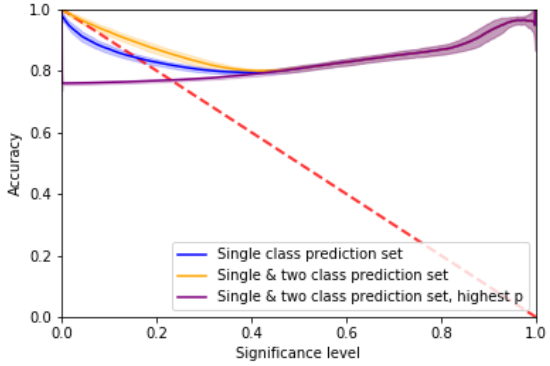


(B) Accuracy of different types of prediction sets

FIGURE A.11: Quadratic discriminant analysis model



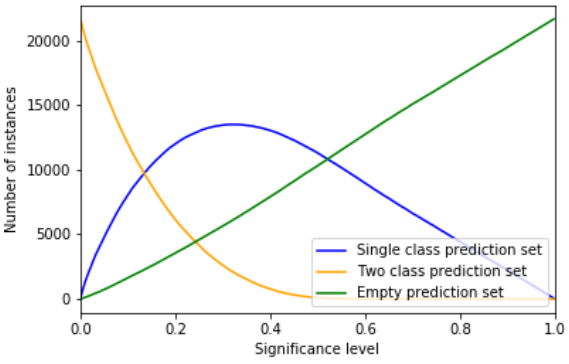
(A) Size of different types of prediction sets



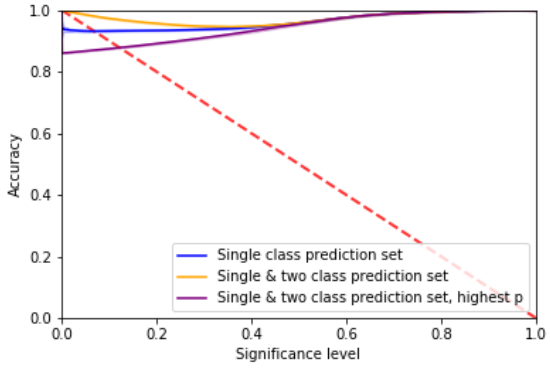
(B) Accuracy of different types of prediction sets

FIGURE A.12: Naive Bayes model

A.1.3 Adult dataset

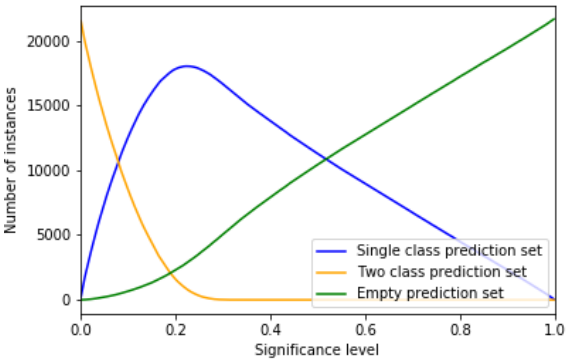


(A) Size of different types of prediction sets

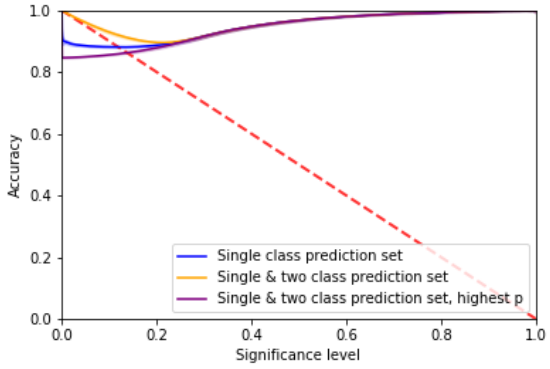


(B) Accuracy of different types of prediction sets

FIGURE A.13: Random Forest model

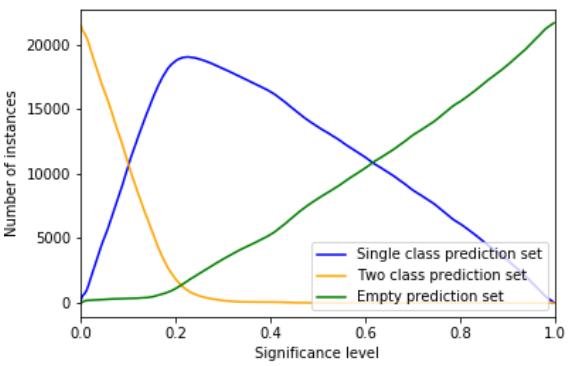


(A) Size of different types of prediction sets

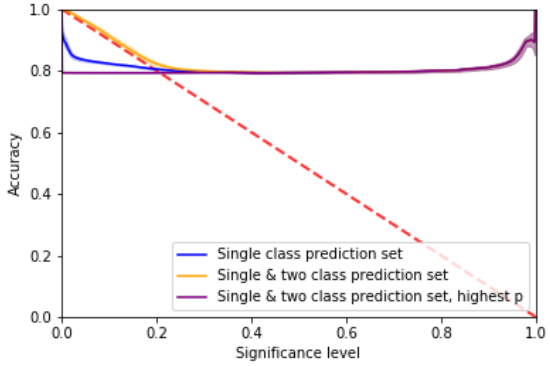


(B) Accuracy of different types of prediction sets

FIGURE A.14: XGBoost model

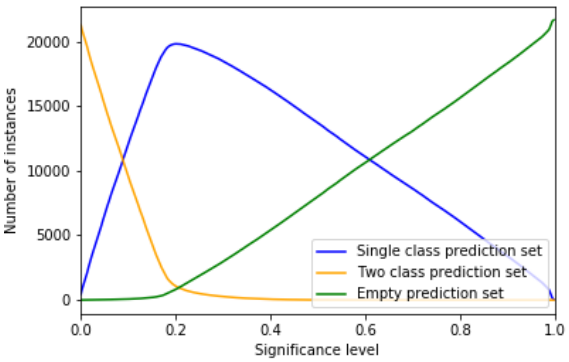


(A) Size of different types of prediction sets

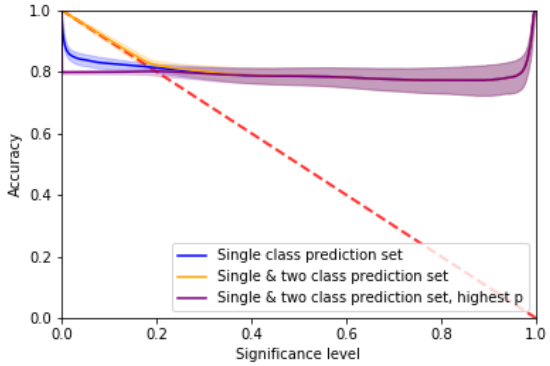


(B) Accuracy of different types of prediction sets

FIGURE A.15: *k*-nearest neighbor model

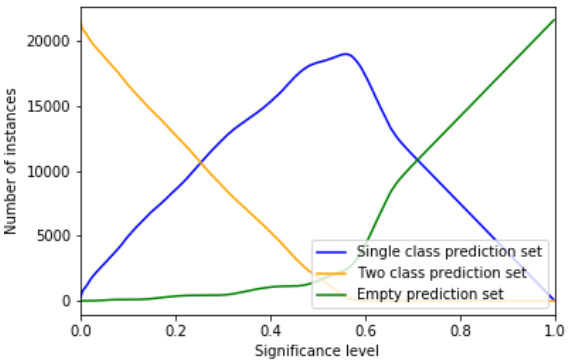


(A) Size of different types of prediction sets

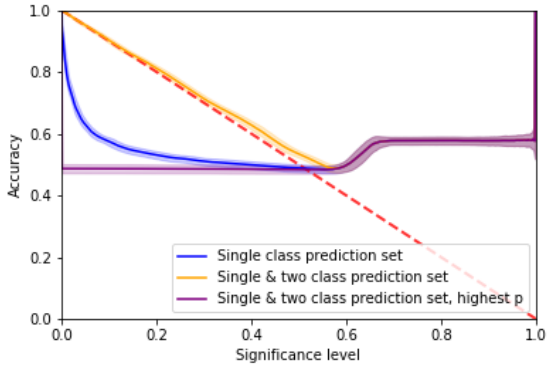


(B) Accuracy of different types of prediction sets

FIGURE A.16: Logistic regression model

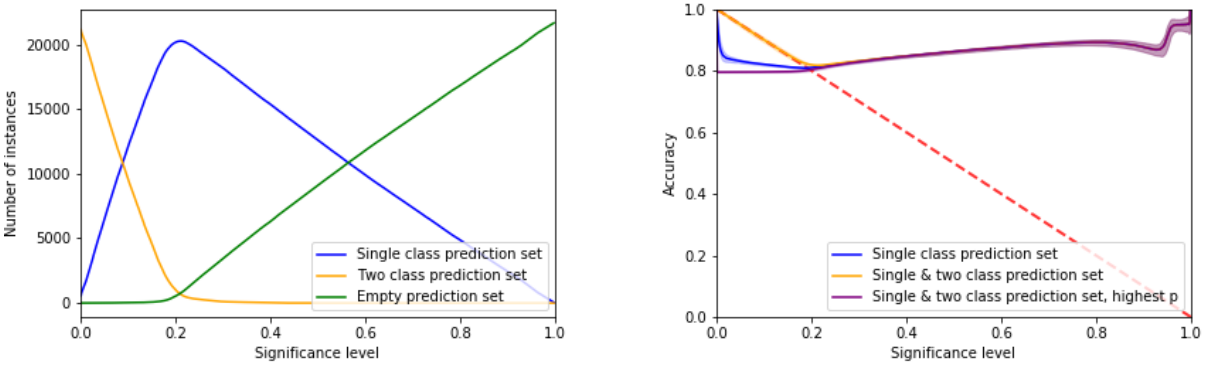


(A) Size of different types of prediction sets



(B) Accuracy of different types of prediction sets

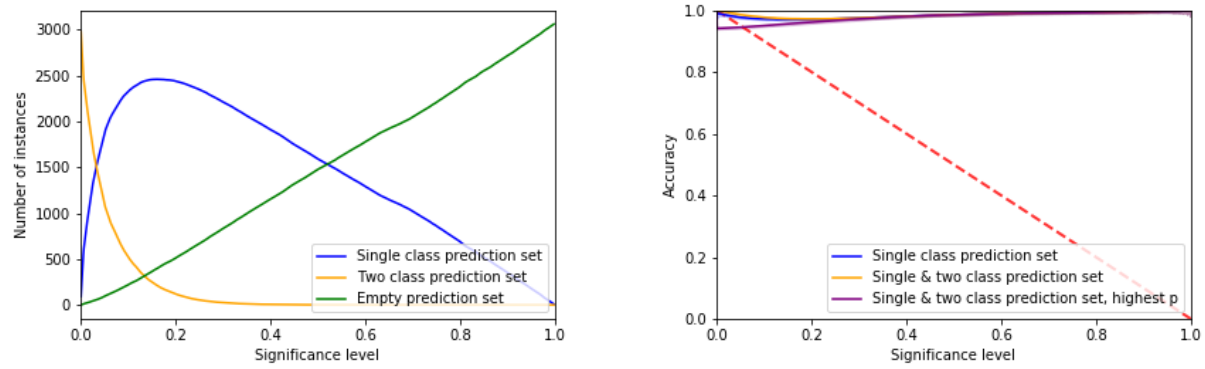
FIGURE A.17: Quadratic discriminant analysis model



(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

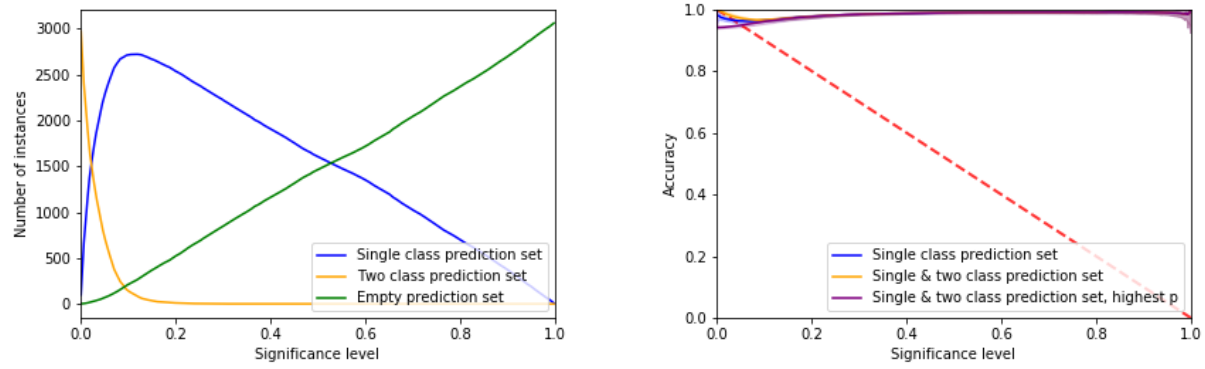
FIGURE A.18: Naive Bayes model

A.1.4 Spambase dataset



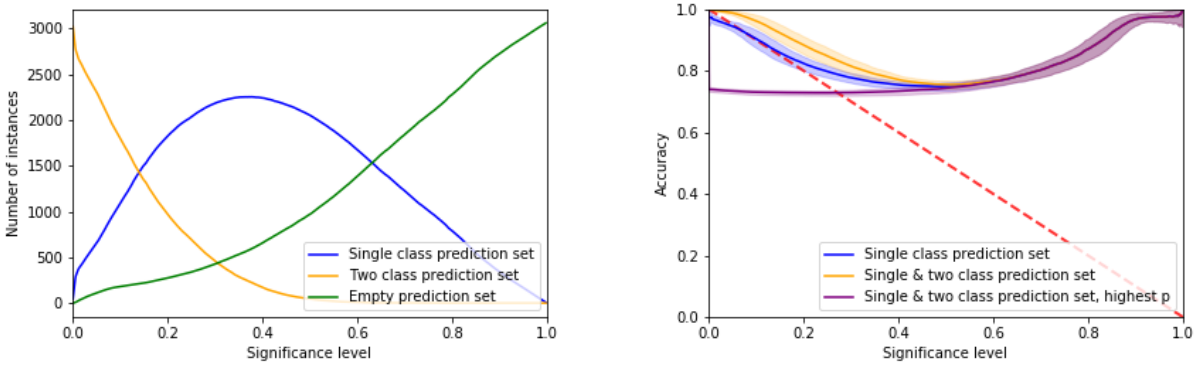
(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

FIGURE A.19: Random Forest model



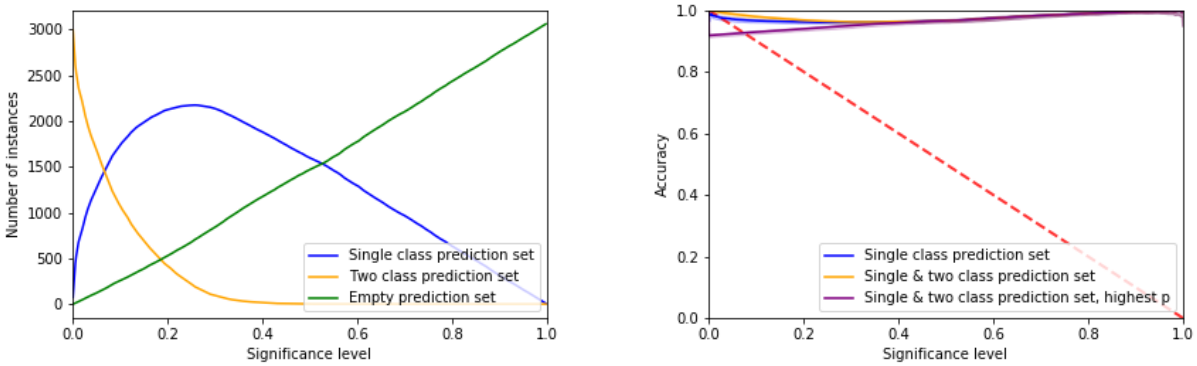
(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

FIGURE A.20: XGBoost model



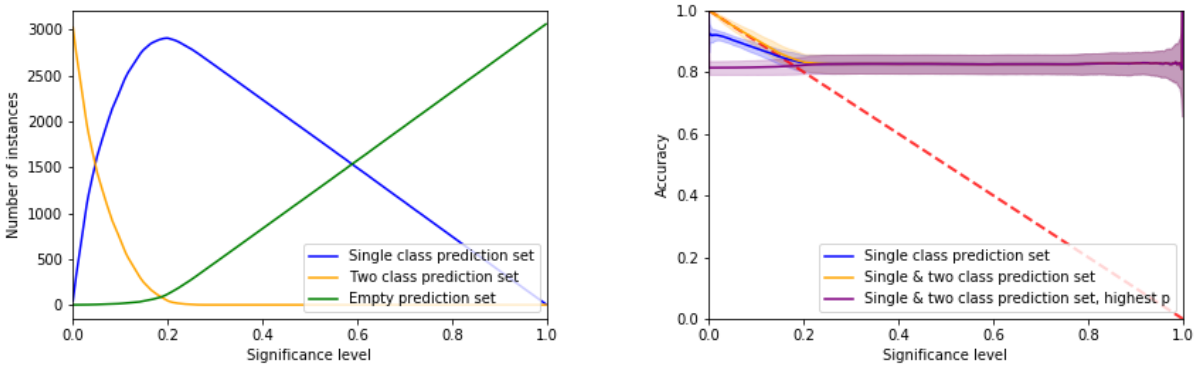
(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

FIGURE A.21: *k*-nearest neighbor model



(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

FIGURE A.22: Logistic regression model



(A) Size of different types of prediction sets (B) Accuracy of different types of prediction sets

FIGURE A.23: Quadratic discriminant analysis model

A.2 Correlation between error and metrics

TABLE A.1: The correlation between different measures and the error of individual predictions with the *churn* dataset

Metric	Random Forest		XGBoost	
	Correlation Coeff	p-value	Correlation Coeff	p-value
Conf	-0.335	2.43E-14	-0.322	4.20e-144
Cred	-0.030	0.30	-0.275	4.47e-09
Prob	-0.377	2.84E-22	-0.275	4.46e-111
Prob*Conf	-0.377	9.04E-21	-0.337	1.09e-166
Prob*Cred	-0.148	6.11E-3	-0.289	2.39e-10
Prob*Conf*Cred	-0.202	7.81E-6	-0.314	1.83e-12
Prob*(Conf+Cred)	-0.318	1.74E-14	-0.333	5.77e-15
Prob*(Conf^2+Cred)	-0.330	2.67E-16	-0.349	1.23e-16
Cred-(1-Conf)	-0.193	1.83E-5	-0.328	4.58e-14
Prob*(Cred-(1-Conf))	-0.233	5.21E-8	-0.330	2.64e-14

TABLE A.2: The correlation between different measures and the error of individual predictions with the *adult* dataset

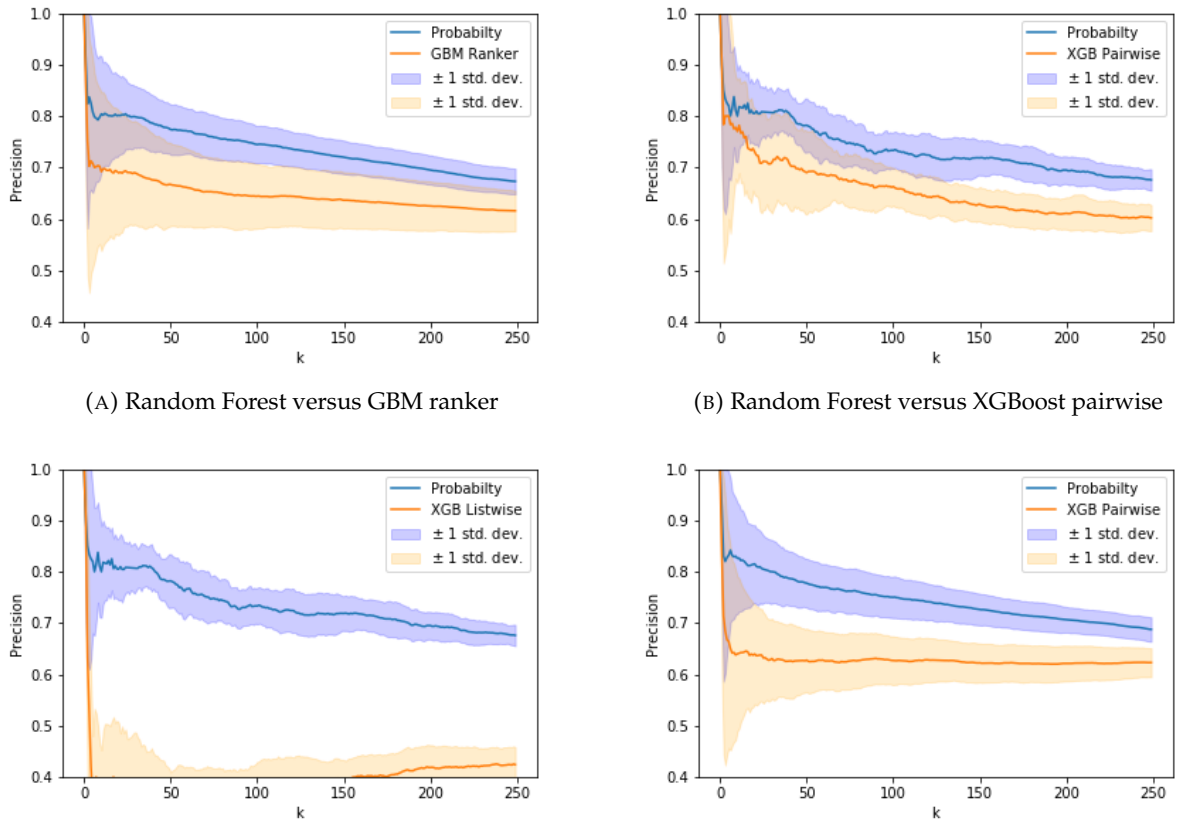
Metric	Random Forest		XGBoost	
	Correlation Coeff	p-value	Correlation Coeff	p-value
Conf	-1.61e-01	9.26e-18	-1.38e-01	3.30e-11
Cred	-2.97e-01	2.56e-60	-3.85e-01	2.80e-107
Prob	-4.38e-01	5.72e-135	-3.37e-01	7.04e-74
Prob*Conf	-3.84e-01	1.52e-99	-3.29e-01	4.88e-69
Prob*Cred	-3.33e-01	3.34e-77	-3.92e-01	1.06e-111
Prob*Conf*Cred	-3.48e-01	5.50e-84	-4.01e-01	1.19e-116
Prob*(Conf+Cred)	-4.02e-01	3.31e-112	-4.22e-01	9.96e-132
Prob*(Conf^2+Cred)	-4.00e-01	8.09e-111	-4.22e-01	7.15e-132
Cred-(1-Conf)	-3.62e-01	7.24e-91	-4.05e-01	9.76e-120
Prob*(Cred-(1-Conf))	-3.60e-01	8.98e-90	-4.07e-01	1.44e-120

TABLE A.3: The correlation between different measures and the error of individual predictions with the *spambase* dataset

Metric	Random Forest		XGBoost	
	Correlation Coeff	p-value	Correlation Coeff	p-value
Conf	-0.185	0.03	-0.089	0.228
Cred	-0.214	7.52e-05	-2.77e-01	2.35e-05
Prob	-3.835e-01	1.60e-08	-4.01e-01	5.58e-12
Prob*Conf	-3.631e-01	1.76e-08	-3.81e-01	2.41e-11
Prob*Cred	-0.231	2.89e-05	-2.81e-01	1.98e-05
Prob*Conf*Cred	-2.395e-01	1.18e-04	-2.82e-01	2.08e-05
Prob*(Conf+Cred)	-3.032e-01	4.74e-06	-3.33e-01	1.73e-07
Prob*(Conf^2+Cred)	-3.081e-01	2.28e-06	-3.34e-01	1.89e-07
Cred-(1-Conf)	-2.491e-01	8.76e-05	-2.83e-01	2.07e-05
Prob*(Cred-(1-Conf))	-2.494e-01	6.82e-05	-2.84e-01	1.95e-05

A.3 Comparison pairwise, listwise and pointwise approach

In this thesis a novel pointwise approach is proposed incorporating confidence in the ranking for a number of problems. As a baseline probability estimation by a number of machine learning models is used, where we sort by the probability predicted. In this section this baseline is compared against pairwise and listwise ranking approach to determine if this is indeed the strongest baseline. For the pairwise approaches the LightGBM ranking model is used as well as the XGBoost model [38][15]. For the listwise approach only the XGBoost model is used. For both the pairwise and listwise approach using XGBoost MAP is used to optimize. The XGBoost model uses LambdaMART for both approaches [14].



(A) Random Forest versus GBM ranker

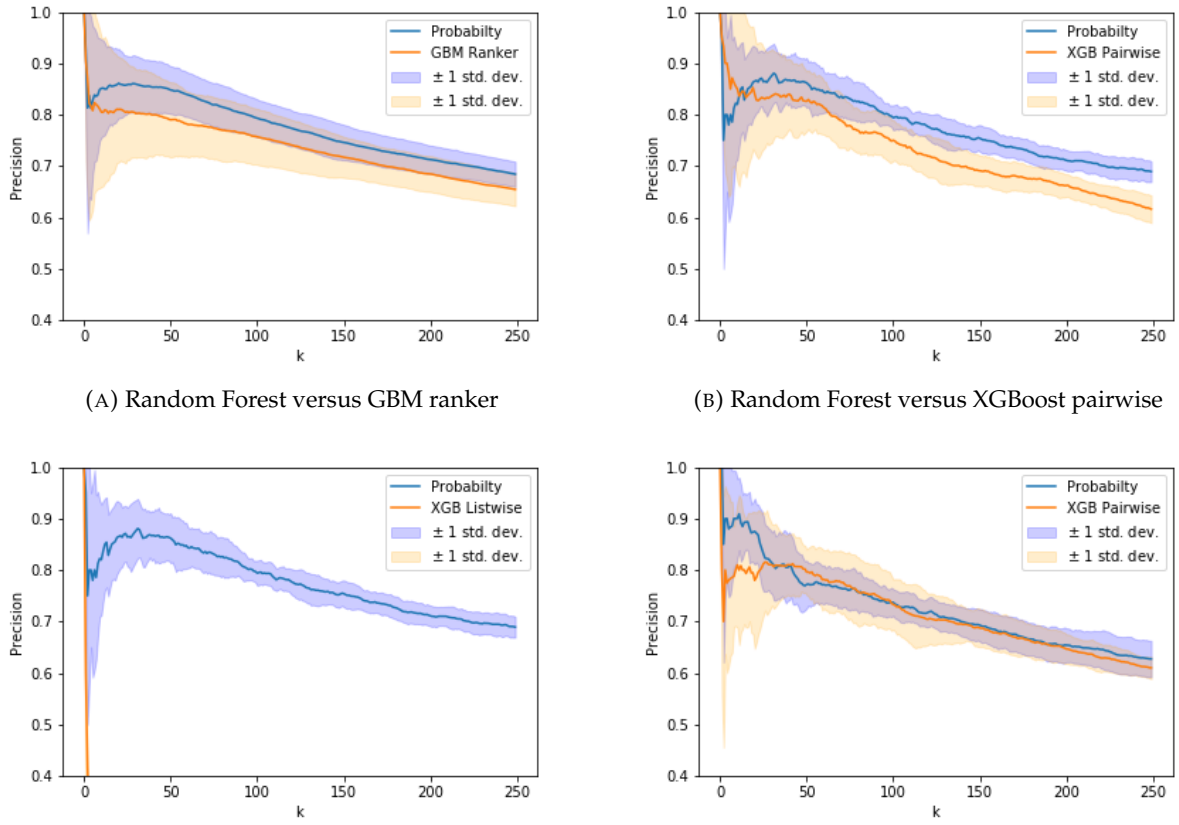
(B) Random Forest versus XGBoost pairwise

(C) Random Forest versus XGBoost listwise

(D) XGBoost versus XGBoost pairwise

FIGURE A.24: Comparing the pairwise and listwise approaches against the pointwise approach on the *inland* ship dataset

From the resulting precision@*k* it is found that the pointwise approach outperform the pairwise and listwise ranking approaches when using the best performing classification models with their underlying probability distribution estimation for this pointwise approach. The reason of these results is most likely due to the fact that the problems in this thesis are all bipartite ranking problems.



(A) Random Forest versus GBM ranker

(B) Random Forest versus XGBoost pairwise

(C) Random Forest versus XGBoost listwise

(D) XGBoost versus XGBoost pairwise

FIGURE A.25: Comparing the pairwise and listwise approaches against the pointwise approach on the *churn* dataset

A.4 Confidence reranking

Dataset	Classifier	Prec@5			Prec@10		
		Prob	Prob*Conf	p	Prob	Prob*Conf	p
Ship	rf	80.8%	72.8%	1.1E-3	80.3%	76.9%	3.5E-2
	xgb	83.6%	74.8%	6.6E-4	81.0%	72.3%	6.8E-6
	knn	65.6%	65.6%	1	66.3%	68.5%	0.26
	lr	54.2%	53.5%	0.83	57.3%	53.0%	4.2E-2
	nb	47.4%	56.0%	2.8E-3	46.9%	53.5%	2.7E-3
churn	rf	82.7%	81.6%	0.57	85.1%	83.5%	0.25
	xgb	89.3%	87.4%	0.34	87.4%	86.6%	0.56
	knn	82.0%	79.8%	0.35	80.1%	79.6%	0.77
	lr	89.0%	89.5%	0.73	85.5%	88.3%	2.0E-2
	nb	77.8%	81.0%	0.18	81.2%	82.0%	0.64
adult	rf	100%	100%	nan	100%	100%	nan
	xgb	100%	100%	nan	100%	100%	nan
	knn	99.0%	99.6%	0.31	98.4%	99.4%	3.1E-2
	lr	100%	99.6%	0.15	100%	99.3%	1.8E-2
	nb	97.2%	99.4%	5.1E-3	97.6%	99.2%	3.8E-3

Dataset	Classifier	Prec@25			Prec@50		
		Prob	Prob*Conf	p	Prob	Prob*Conf	p
Ship	rf	80.1%	77.8%	0.1	77.4%	76.1%	7.8E-2
	xgb	79.4%	71.3%	1.1E-11	77.4%	69.9%	1.9E-16
	knn	65.4%	67.8%	4.5E-2	60.7%	62.3%	5.7E-2
	lr	58.0%	50.6%	1.5E-6	52.6%	48.7%	1.3E-4
	nb	47.3%	53.0%	4.3E-5	47.3%	52.1%	3.8E-7
churn	rf	85.8%	85.9%	0.96	84.8%	85.6%	0.15
	xgb	82.0%	84.8%	4.7E-3	78.8%	80.8%	3.1E-3
	knn	79.6%	79.5%	0.91	78.1%	78.9%	0.27
	lr	84.2%	87.0%	8.5E-4	82.4%	83.8%	2.9E-2
	nb	76.0%	81.1%	1.2E-5	73.3%	79.7%	2.5E-12
adult	rf	100%	100%	nan	100%	100%	nan
	xgb	100%	100%	nan	100%	100%	nan
	knn	98.4%	99.4%	4.2E-3	98.5%	99.3%	1.9E-4
	lr	99.6%	98.6%	2.4E-3	96.8%	96.8%	0.90
	nb	97.1%	99.1%	9.2E-8	97.4%	99.0%	1.5E-9

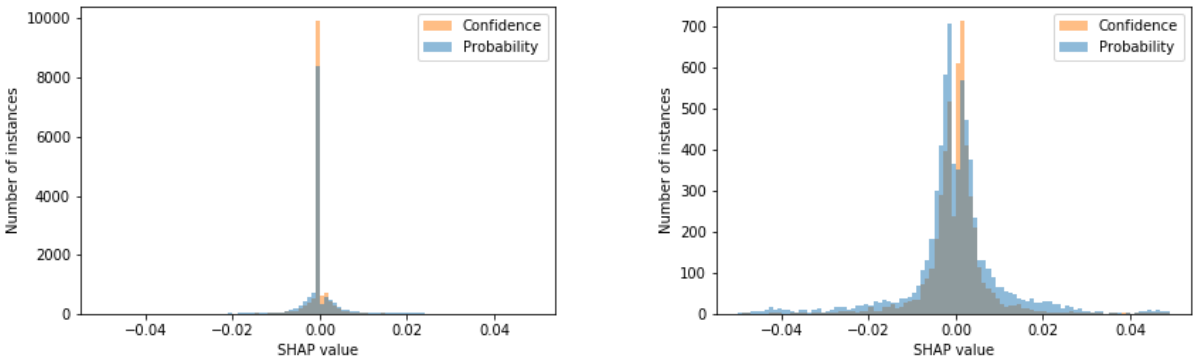
Appendix B

Explaining based on confidence with the *churn* dataset

B.1 Difference in SHAP values on *churn* dataset

TABLE B.1: Difference in SHAP values between the context of *probability* and *confidence*

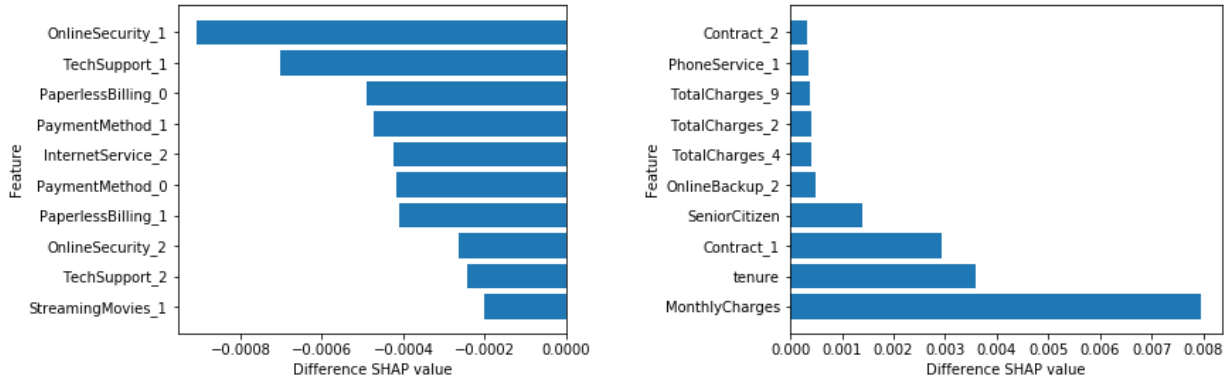
Context	Average absolute SHAP value	Without 0	# instances of 0
Probability	3.86E-3	8.33E-3	6951
Confidence	3.14E-3	8.87E-3	5309



(A) SHAP values between two contexts

(B) SHAP values minus zero instances

FIGURE B.1: Difference in SHAP values between two contexts on *churn* dataset

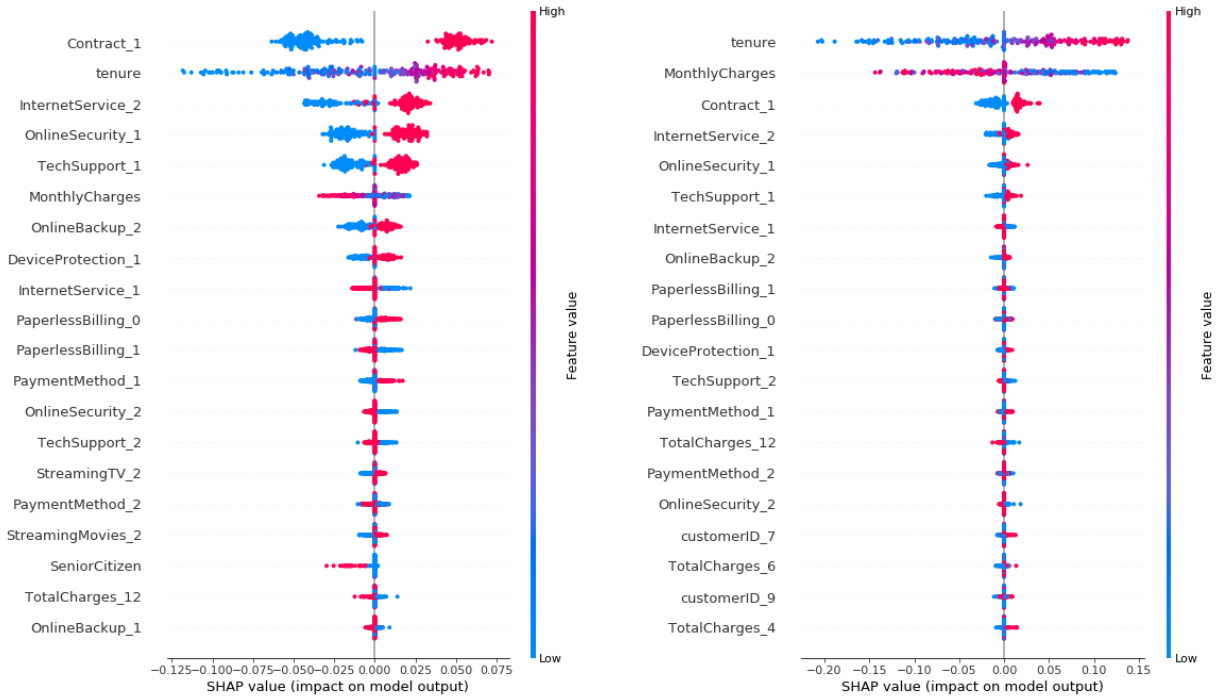


(A) Decreased SHAP values

(B) Increased SHAP values

FIGURE B.2: Difference of SHAP values between the two contexts on *churn* dataset

B.2 Global feature importances



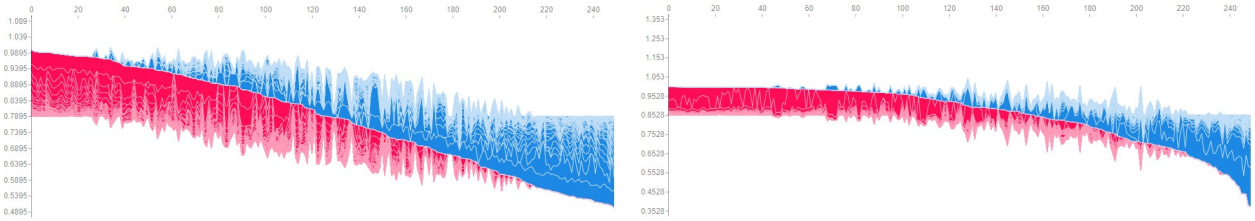
(A) Probability feature contribution

(B) Confidence feature contribution

FIGURE B.3: Summary plot of the most important features

B.3 Single prediction forces

FIGURE B.4: SHAP values of an single prediction of the RF on the *churn* dataset. The plot *above* is from the context of *probability*, the *lower one* from the context of *confidence*

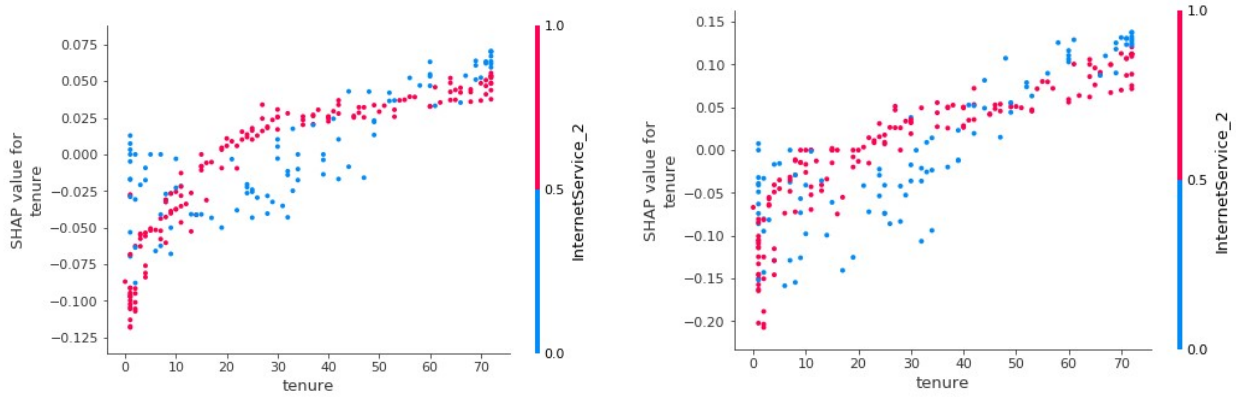


(A) Probability feature contribution of all individual prediction

(B) Confidence feature contribution of all individual prediction

FIGURE B.5: Force plot of all individual predictions

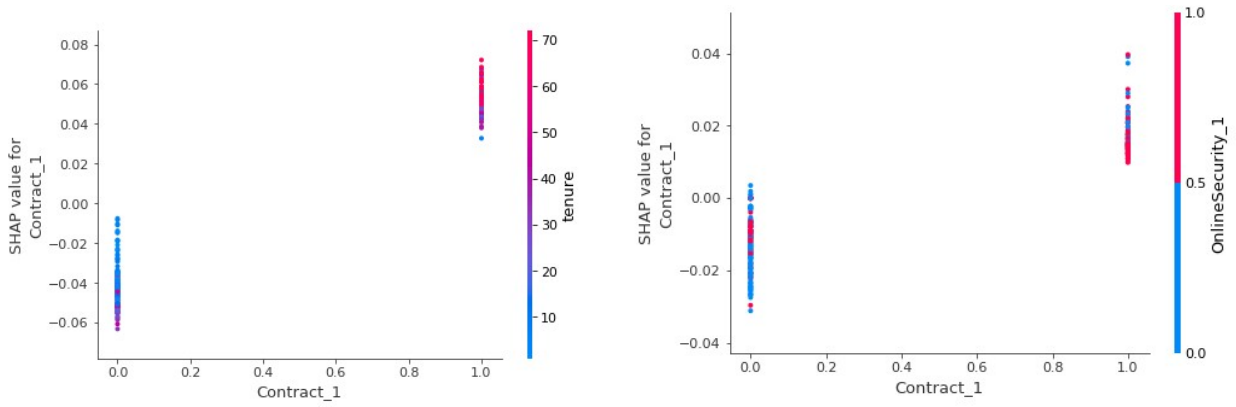
B.4 Interaction values



(A) Probability context

(B) Confidence context

FIGURE B.6: Relationship of the *tenure* feature value and SHAP value



(A) Probability context

(B) Confidence context

FIGURE B.7: Relationship of the *contract* feature value and SHAP value

Appendix C

Additional results user study

C.1 Agreement between participant and prediction

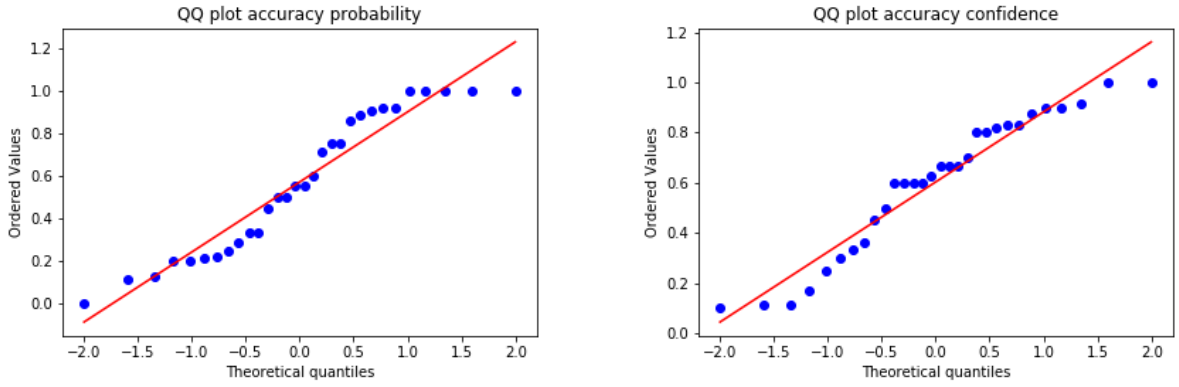
Based on the results of the user study it is possible to implicitly determine the agreement between the model and the participants of the study. While not directly asking participants whether they agree, it is possible to determine to not follow the prediction of violation. In Table C.1 the percentage of times participants followed the predictions of the model is given.

TABLE C.1: Agreement in a number of situations

	Probability	Confidence
All explanations	73.3%	75.6%
Positive explanations	74.5%	74.2%
Negative explanations	72.2%	76.1%
Correct explanations	74.0%	76.1%
Incorrect explanations	72.0%	73.4%

C.2 QQ plots accuracy of both contexts

In Figure C.1 the quantile-quantile plot of the two contexts in the user study is given. This plot serves as a graphical tool to help assessing whether the accuracy found came from a normal distribution. The quantiles of the accuracy is plotted against the quantiles of a normal distribution. If both quantiles came from the same distribution, there should be a strong linear relation (a straight line).



(A) Probability

(B) Confidence

FIGURE C.1: QQ plots between both contexts

C.3 Nonparametric hypothesis tests

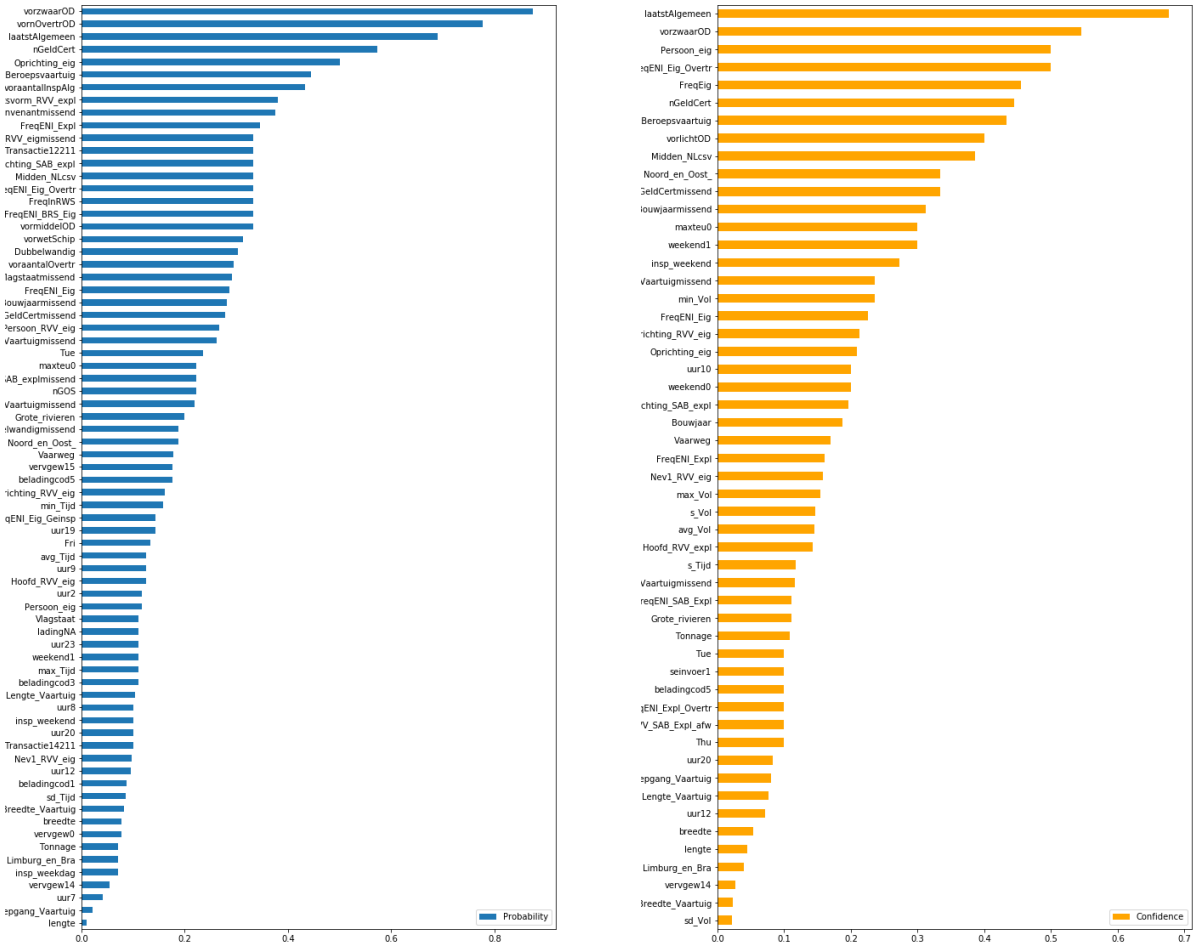
TABLE C.2: Results for Hypothesis 2 with Mann Whitney U test

	Probability		Confidence		p-value
	Mean	Std	Mean	Std	
All explanations	3.11	1.05	3.06	1.15	0.28
Positive explanations	3.11	1.12	2.86	1.16	0.08
Negative explanations	3.11	0.98	3.18	1.11	0.55
Correct explanations	**3.19**	1.04	**2.90**	1.15	**0.01**
Incorrect explanations	**2.96**	1.07	**3.26**	1.11	**0.03**

TABLE C.3: Results for Hypothesis 3 with Mann Whitney U test

	Probability		Confidence		p-value
	Mean	Std	Mean	Std	
All explanations	2.80	1.13	2.81	1.23	0.37
Positive explanations	2.80	1.21	2.69	1.19	0.44
Negative explanations	**2.80**	1.06	**2.92**	1.26	0.37
Correct explanations	2.81	1.13	2.78	1.21	0.24
Incorrect explanations	**2.78**	1.14	**2.86**	1.27	0.21

C.4 Usefulness features



(A) Explanations of probability

(B) Explanations of confidence

FIGURE C.2: Percentage of times features reported to be useful

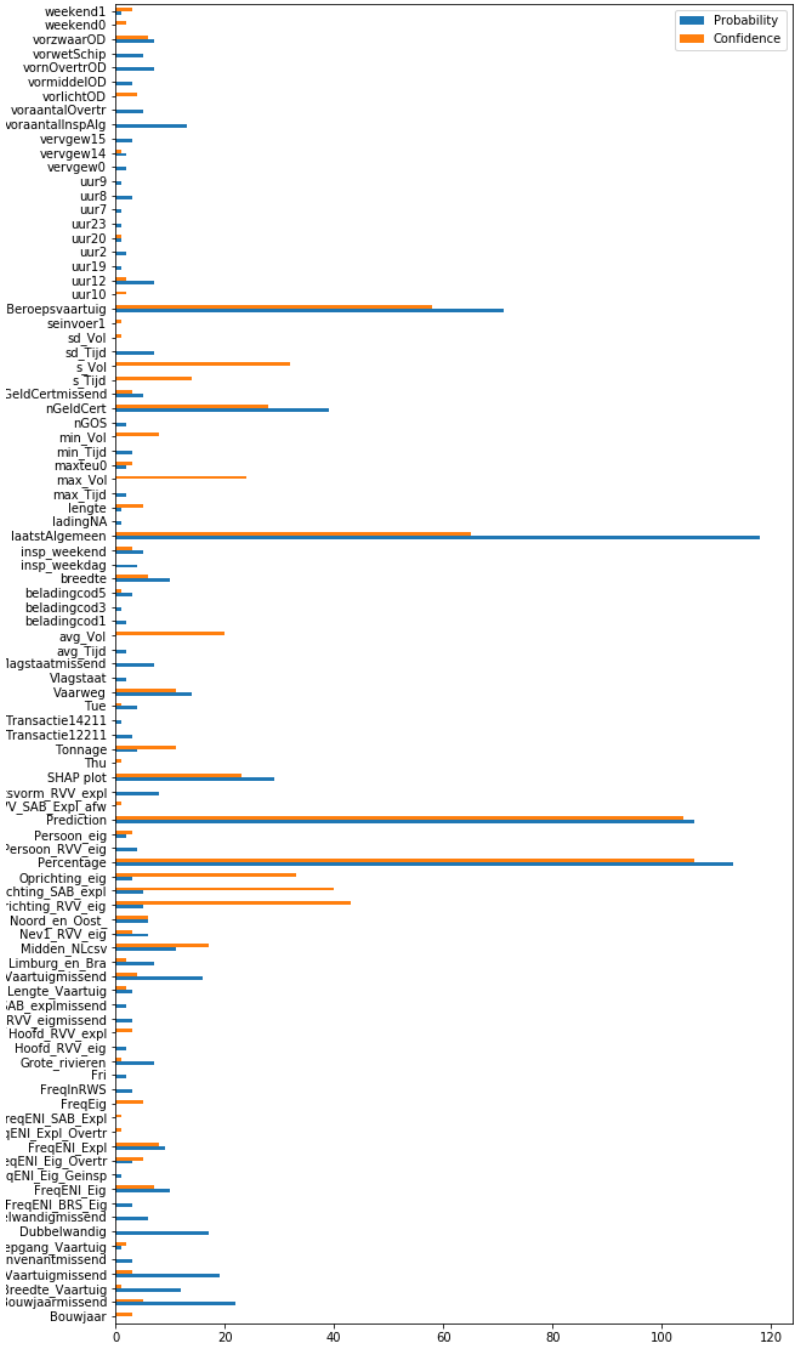


FIGURE C.3: Comparing the reported usefulness of features

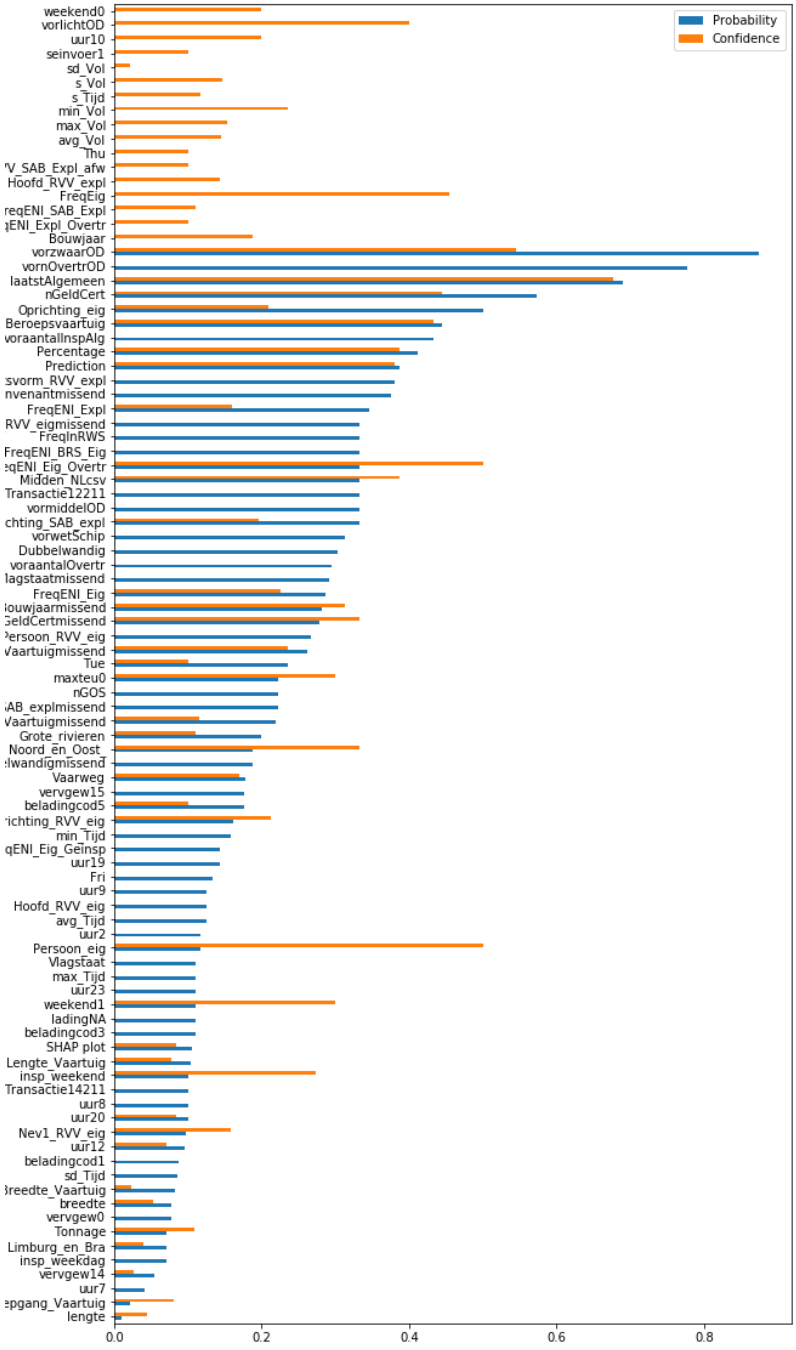


FIGURE C.4: Comparing the reported usefulness of features expressed in percentage of time shown

C.5 Power analysis

With the resulting difference between the two context not being significant for the perceived usefulness and user trust post-hoc power analysis was performed to determine the sample size needed to get significant results with a given degree of confidence. It allows to determine the probability of detecting an effect of a given size with a given level of confidence, under sample size constraints. In other words, it is the probability of rejecting the null hypothesis when it is in fact false.

Ad hoc Power analysis was performed before the experiment based on guesses of the difference between the mean of the two contexts and the standard deviation expected, with a two sample one sided test to determine whether the mean of context of probability P is different from the mean of context of confidence C. With the hypotheses being:

$$H_0 : \mu_P = \mu_C,$$

$$H_1 : \mu_P \neq \mu_C$$

The ratio between the two contexts is $\kappa = \frac{n_C}{n_P}$.

To determine the expected sample size needed the following function is used:

$$n_P = (\sigma_P^2 + \sigma_C^2/\kappa)(\frac{z_{1-\alpha} + z_{1-\beta}}{\mu_P - \mu_C}),$$

with z being the inverse of the cumulative distribution function, α being the Type I error and β the Type II error and power being $1 - \beta$. As the amount of task from a single context is equal in the user study $\kappa = 1$ and $n_P = n_C$.

Before the user study it was assumed that the difference in the mean for both the perceived usefulness and user trust would be 0.25, with $\sigma_P = 1$ and $\sigma_C = 1.2$. The difference in the standard deviation was assumed due to the larger variation in features deemed useful in the context of confidence.

With $1 - \beta = 0.8$ and $\alpha = 0.05$, $n_P = n_C = 242$ was determined to be the necessary number of tasks completed for significant difference between the two contexts. As each participant performed 20 tasks, the predicted minimum number of participants was 25.

For the determination of sample size only the perceived usefulness and user trust was used to determine the minimum number of participants, this is due to the difficulty of determining significance for *task effectiveness* with the limited number of available participants.

This is due to the distribution compared in the power analysis being a discrete binary distributions when not averaging per ship. Together with the low accuracy of the model predicting violations, the standard deviated was expected to be high.

Post hoc The actual difference between μ_P and μ_C is smaller than predicted. Therefore power analysis was performed again on the actual values to determine the sample size needed to determine significance.

For the perceived usefulness the actual values are $\mu_P = 3.11$, $\mu_C = 3.026$, $\sigma_P = 1.05$ and $\sigma_C = 1.15$. This results in $n_P = n_C = 2129$, meaning a minimum of 213 participants is needed to determine significance. For the user trust the actual values are $\mu_P = 2.80$, $\mu_C = 2.81$, $\sigma_P = 1.13$ and $\sigma_C = 1.23$. This results in $n_P = n_C = 145151$, meaning a minimum of 14515 participants is needed to determine significance.

Task effectiveness Task effectiveness was not included in the ad hoc power analysis. However, after finding no significant difference between the two contexts in *task effectiveness*, post hoc analysis was performed to determine the number of participants necessary to confirm the improvement found when using explanations from the context of confidence.

When looking at the accuracy of individual instances $\mu_P = 0.583$, $\mu_C = 0.591$, $\sigma_P = 0.493$ and $\sigma_C = 0.491$. This results in $n_P = n_C = 46333$, meaning a minimum of 2317 participants is needed to determine significance. This was not unexpected, and the reason *task effectiveness* wasn't included in the ad hoc power analysis.

In the post hoc analysis, the power analysis is also performed on the average *task effectiveness* per ship. The task effectiveness found is $\mu_P = 0.579$, $\mu_C = 0.619$, $\sigma_P = 0.232$ and $\sigma_C = 0.252$. This resulted in $n_P = n_C = 391$, meaning 391 ships. The minimum number of participants necessary to determine significance would be 196.

C.6 Time taken between contexts

While not hypothesised, the time taken of a single task is kept track off. This is to see if one of the contexts results in quicker completion of the tasks. In Figure C.5 a histogram representing the time taken of tasks is shown. Outliers are removed by only looking at quantile 0.1 to 0.9. This removes tasks where the participant took a break or the task were completed in less than 15 seconds.

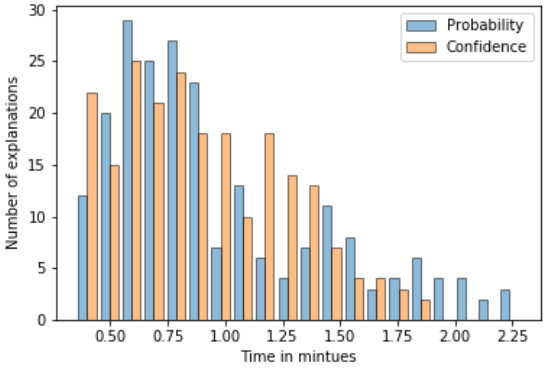


FIGURE C.5: Time taken to complete a single task

TABLE C.4: Time taken between contexts

	Probability		Confidence		p-value
	Mean	Std	Mean	Std	
All explanations	0.97	0.47	0.90	0.36	0.08

Looking at the average time taken between context there is a slight improvement when looking at confidence, however not significantly ($p = 0.08$). This is caused by a larger number of tasks performed in under half a minute. The reason could be be twofold. Firstly, inspectors notice features which cause the identification of the ship. The decision can therefore quickly be made. Secondly, the features explaining confidence were on average less useful. Less selecting of features can speed up task completion. The most tasks with explanations of probability are completed between 30 second and 1 minute, however, certain task are completed in more than 2 minutes. When looking at confidence the spread is larger when looking at most tasks (between 20 second and 1.5 minutes). However no task is completed in more than 2 minutes.