Norwegian School of Economics

Bergen, Spring 2021

Explaining Individual Predictions on Financially Distressed Companies Using Shapley Values

Henrik Rodahl Dokset and Eirik Vindenes

Supervisor: Håkon Otneim

Master thesis, MSc in Economics and Business Administration

Major: Business Analytics

NORWEGIAN SCHOOL OF ECONOMICS

Abstract

Prediction results from complex machine learning models can be challenging to interpret. Understanding these models is essential when trusting results in decision-making. In this master thesis, we will utilize Shapley values to explain individual predictions from a complex machine learning algorithm. Our aim is to explain why prediction models obtain their results, so people can interpret them better.

The chosen case is based on a thesis called “Predicting Financial Distress in Norway” by Zhang and Ye (2019) where they used logistic regression and random forest models. Their thesis predicts whether a company enters financial distress within the next two years or not. In this thesis, we will take advantage of the powerful algorithm in xgboost (extreme gradient boosting). To illustrate the benefits of using a complex model versus a simple model, we will also present a decision tree as our baseline.

Our explanation analysis shows that predictions made by xgboost can be explained with the Shapley value framework to obtain clear and intuitive explanations. Calculating Shapley values for a larger group of predictions enables proper understanding of the model by investigating which feature values lead to what probability increase or decrease of distress. The explanation framework enables detection of possible model bias which sometimes can lead to discrimination. We conclude that using Shapley values as an explanatory framework enables decision-makers to continue using complex machine learning models. This is important, as we find the tool satisfying relevant regulations for decisions made by automatic systems to be explained upon request.

Acknowledgements

This thesis is a part of our MSc in Economics and Business Administration at the Norwegian School of Economics (NHH). We are both majoring in Business Analytics.

We would sincerely like to thank our supervisor, Associate Professor Håkon Otneim for guidance and support. With his help and advice, our thesis was vastly improved in quality. For technical help, we would like to thank Senior Research Scientist at Norsk Regnesentral, Martin Jullum for help with the R package *shapr*.

We would also like to thank the authors behind our case, Guang Na Zhang and Fan Ye. Also, thanks to Centre for Applied Research (SNF) at NHH for providing the necessary dataset to perform our analysis.

Finally, we would like to thank family and friends for their support during this spring.

Contents

List of Tables

List of Figures

1 Introduction

Machine learning stands for most of the recent advances in technology and science (Riberio, Singh, & Guestrin, 2016a). People are impacted by tasks machine learning is applied to, especially in medical, judicial, and financial decisions. For machine learning to be used in decision-making, decision-makers need to trust the model. It creates a demand for model interpretability since a fundamental element in trusting a prediction model is to understand its behavior (Dziugaite, Ben-David, & Roy, 2020).

A general notion of machine learning is that models are viewed as black-boxes (Riberio et al., 2016a). This means the model produces results without giving any indication on why the results are obtained. When these models then are used in decision-making, an explanation issue arises. High-performance machine learning models run the risk of not being accepted ethically or legally if there is a lack of explanation (Bibal, Lognoul, de Streel et al., 2020). When considering the legal aspect of machine learning models, Bibal et al. (2020) claim there is no unique definition of explainability in law, but rather that the requirements set by law depend on which sector the model is applied to. Examples of legal requirements posed by law can be to provide the main parameters in a model or to explain how the features end up with a given result. The latter of which we will pursue in this thesis. According to the European Parliament (2019), people affected by decisions made by machine learning models have the right to an explanation. This regulation has implications in practice for actors using the models for decision-making. It is thus not enough to follow machine learning models blindly and explanations behind predictions should be given.

While simple machine learning algorithms can be interpreted directly, complex models are difficult to explain. This thesis will explore the difference using a simple decision tree and by comparing it to the black-box model xgboost (extreme gradient boosting). Interpreting predictions from a decision tree can be done directly by plotting and studying the tree model. By doing so, we can view how the model considers features and how individual predictions are calculated. This is a simple exercise in which a non-expert could do. For a xgboost model, on the other hand, the interpretation procedure is difficult due to the complexity of the model. It is possible to illustrate the model but interpreting how it works is difficult, especially for non-experts. In the end, it can be the non-experts who must make decisions and who are

affected by the machine learning models. It is therefore important for them to understand the model to trust the results.

Furthermore, there is a fundamental trade-off between machine learning interpretability and model accuracy in machine learning terms (James, Gareth, Witten et al., 2017, p. 25). Complex models will usually outperform simpler models in predictive accuracy, with the downside of losing interpretability. For small differences in accuracy between two models, the simplest model will be preferred when there is a need for model interpretability (Ribeiro, Singh, & Guestrin, 2016b). However, as we will illustrate in this thesis, model accuracy can vary between models. Losing out on too much accuracy for interpretability will not benefit decision-making. To compensate for the loss in interpretability for complex models, certain tools can be used to explain predictions. Good explanations for complex and accurate models remove the trade-off and can be considered valuable. Model-agnostic explanations systems allow the user to choose whatever machine learning algorithm they want because predictions can be explained by a generic framework for interpretability (Ribeiro et al., 2016b). Ribeiro et al (2016b) also conclude that these explanation methods are essential for users to trust machine learning models.

Another point to discuss is explanation methods can either be global or local (Aas, Jullum, & Løland, 2021). Global approaches study the whole model while local models focus on individual predictions to be explained. Global approaches could be too simple and thus not be a good representation for local behaviors (Ribeiro et al., 2016b). In this thesis, we will present a local method to explain predictions, which is based on game theory, known as Shapley values. There are many benefits of using Shapley values over other similar explanations methods. The main reason is the unique properties. Shapley value properties enable individual explanations to be fair and evenly distributed. Individual explanations are beneficial for many reasons. One of those is that there are often individual differences in a prediction model, and thus a global explanation method is not satisfactory. Kernel SHAP is an explanation method that approximate Shapley values (Aas, Jullum, et al., 2021). Approximations are necessary due to the computational problem faced with Shapley values. We will explain this in detail, but in short, Shapley values take all possible coalitions (all combinations of features) into consideration. This is a challenge when computing Shapley values for many features. Another drawback with the method is that it does not consider dependency between features, leading to inaccurate explanations when using Kernel SHAP in real-world situations. Aas, Jullum, et

al. (2021) has dealt with the problem by incorporating a dependence structure in Kernel SHAP which estimates the distributions. Their study proves that this new method is an improvement compared to other methods. The case we have chosen is taken from real-world data, hence this new method is applicable to our problem. Lundberg and Lee (2017) found Shapley values to be close to human explanations compared to other methods. This is a major benefit because the primary motivation for using explanation methods is for humans to interpret models better.

In addition to increasing trust and understanding models, explanations can also detect bias and discrimination in the model. When fitting a too simple model to a complex problem, the model is biased. Bias could appear in many ways and is therefore difficult to clearly define. We believe by computing Shapley values for a large sample of predictions, we can find out how the model considers the feature values. This is very interesting as it will enable us to open the black-box model and understand how it consider features. Suppose the model picked up noise in our training data, then this would be discovered when we study how Shapley values interact with feature values. Thus, explanations enable us to discover bias, so the model can be adjusted. Since machine learning models do not put features into context, there is a possibility for unexpected outcomes or discrimination. We will investigate this further in this thesis.

The case we will be working with to explain individual predictions is based on a thesis by Zhang and Ye (2019) which predicts financial distress in Norway. Financially distressed companies are in danger of going bankrupt. The prediction model can be used to determine if a company should receive a loan or a new business deal. Suppose a bank would investigate a company's financial health before handing out a loan. If this company is predicted to enter distress within the next few years, then it could be a bad idea to offer a loan as banks are minimizing the probability of default. Rejections on loan applications must be given with reasoning, however, and we assume companies want an explanation behind the decisions made. If an institution decided solely based on the probability output from a model, then an explanation issue arises. Hence, we believe explanation tools can possibly prove valuable for decision-makers in practice. Furthermore, the company could be wrongly predicted and run the risk of being unfairly treated. The only way to obtain some indications on a wrong prediction is to explain the prediction. Individual explanations are thus important for the assessment of a company. We will investigate some individual cases from different sectors in our data. Since Shapley values are computationally expensive, we restricted our analysis to one sector in the data. The analysis can give a good indication of how our model considers

features in this sector. We also suspect that the differences between sectors are small and thus a sector analysis is helpful to others in different sectors.

Therefore, based on prior discussion, we have formulated the following research questions:

- *How intuitive are Shapley value individual prediction explanations for black-box models?*
- *Can unfairness be disclosed by Shapley values in black-box models?*
- *To what extent can Shapley values provide explanations for how black-box models consider different feature values?*

With intuitive, we mean to what degree a non-expert can understand the explanation. For the 2nd research question, we consider that bias in the model can result in discrimination or misinterpretation of the results and lead to unfairness. We consider the detection of bias as important. The 3rd research question will require a more global approach, as we must evaluate how different explanations are given.

1.1 Thesis Overview

The thesis is split into 8 different chapters. In chapter 2 we discuss the relevant literature. In chapter 3 we introduce the Shapley value framework and present how it can be used for explaining predictions. In chapter 4, we present the chosen case discussing financial distress in Norway. Chapter 5 explores how individual predictions can be explained for a simple machine learning algorithm, which is then compared to that of extreme gradient boosting. Next, in chapter 6 we calculate Shapley values for xgboost to explain a prediction for a financially distressed company. We will also perform an extensive sector analysis by using the Shapley value prediction explanation framework when we study dependencies and feature importance. In chapter 7 we evaluate the results and discuss interpretation and implementation for real-life decision-making. Finally, in chapter 8 we will conclude on the research questions.

2 Litterature

In this chapter we will discuss the relevant literature. We will start with studying the trade-off between interpretability and complexity generally and specifically for the models relevant to the thesis. Next, the discussion will be used to explore the theory behind the need for explanation methods to explain complex machine learning algorithms. By studying the pros and cons of the methods, chapter two will continue into chapter three when we present the chosen explanation method for this thesis.

2.1 Interpretability and Complexity in Machine Learning

Interpretability in machine learning is defined as "the use of machine learning models for the extraction of relevant knowledge about domain relationships contained in data" (Murdoch, Singh, Kumbier et al., 2019). In this setting, knowledge refers to relevant insight for affected parties. Knowledge about domain relationships can be presented by visualization, natural language, or mathematic equations. Interpretable machine learning models have become a vital concern (Ribeiro et al., 2016b). The reasons are related to model selection, feature engineering, intuitive user interfaces, and the need for trusting predictions. Ribeiro et al (2016b) claims that interpretable models are preferred over non-interpretable models when accuracy is the same, but also in some cases when accuracy is lower.

As machine learning models have become important for user-face applications, the demand for interpretability in machine learning has increased (Ribeiro et al., 2016b). According to Carvalho, Pereira and Cardoso (2019), a study of google trends shows a massive increase of people searching for "machine learning interpretability" and "machine learning explainability" in the last few years. Prior to 2014, there were no searches for the interpretability aspect, while no searches for machine learning explainability prior to 2016. Hence, this leads us to believe the increase in the usage of black-box models with a need for the explanation behind the predictions has increased notably in the last five years.

Furthermore, Salleh, Talpur and Hussain (2017) argue there is an important trade-off between model accuracy and model interpretability, due to more complex models outperforming the accuracy levels of simpler models. This implies complex models often have high accuracy, but are hard to understand, which results in low interpretability. The trade-off to increase

interpretability often comes with simpler models with lower complexity and lower accuracy. In the figure below, this general trade-off is presented for the relevant decision tree type machine learning algorithms we will discuss in this thesis. Information in the plot is based on research by Duval (2019) and the Mathworks website (*Interpretability*, 2021).

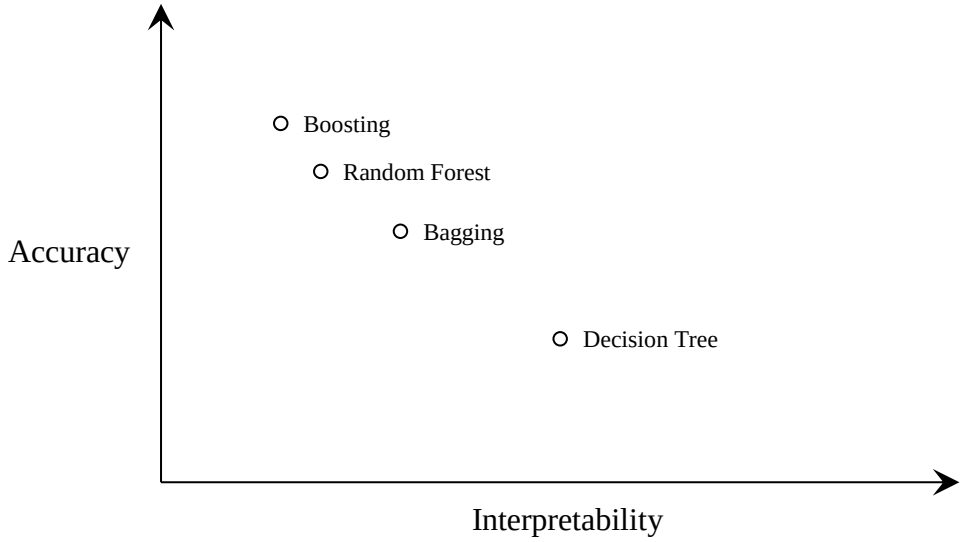

Figure 1. Accuracy/interpretability trade-off (relevant machine learning algorithms)

The trade-off between accuracy and interpretability can be discussed with the bias and variance trade-off in mind. Variance in a model refers to the change in a model when fitting it to new data. If the model varies a lot with different data sets, then it has a high variance. Flexible models can suffer from high variance due to fitting to the data too closely and, therefore, fitting to the errors and the noise of the data (James et al., 2017, pp. 23,35). This is referred to as overfitting. Solutions to reduce overfitting include cross-validation techniques and hyperparameter tuning. K-fold cross-validation is one such type of validation technique that randomly splits the training data into *n-samples* and fits all the samples. We will use this procedure along with hyperparameter tuning during modelling in section 4.2.

Bias on the other hand refers to fitting a too simple model to a complex problem (James et al., 2017, p. 35). High bias is an indication of a model underfitting, meaning the model may be too simple to estimate the problem at hand, indicated by high training errors. Simple machine learning algorithms such as linear regression models can often be biased because they are unable to pick up the complex patterns of real-life problems. They also apply simple assumptions (such as linearity) which is not applicable to complex problems. Meanwhile, more flexible models usually result in lower bias because they fit the data closer.

In this next section we will discuss the algorithms presented in **Figure 1** in terms of the bias-variance tradeoff to explore why the models become accurate when flexibility increases. We will also study the interpretability/complexity aspect of the models to discuss what makes models difficult to interpret. The single decision tree is chosen as our baseline later in this thesis in terms of interpretability and accuracy. The other three models (bagging, boosting and random forest) are ensemble learning type models which involve combining multiple algorithms to solve the same problem (Zhou, 2009).

2.1.1 Machine Learning Models

The simplest model, as presented in the bottom right of **Figure 1**, is a single decision tree. A decision tree works by giving weight to each split in the tree through recursive binary splitting (James et al., 2017, p. 303). The process can be explained by each variable being tested in the training data to estimate each node split and the different thresholds. Decision trees are easy to compute and easy to understand and interpret as the predictive process can be understood by studying the tree model directly. However, they can be very non-robust, as small changes to training data can cause big changes to the tree (James et al., 2017, p. 316). This is because decision trees suffer from high variance, implicating that fits on various samples on the same training data can result in very different tree models. Consequently, decision trees can have low accuracy on out-of-sample predictions. During our case study in chapter 4 and 5, we will illustrate the predictive process of a single decision tree to showcase how predictions can be explained directly.

Bagging, illustrated in the middle of **Figure 1**, improves on the downside of decision trees but loses out on the interpretability aspect in the process. The algorithm works by bootstrapping training data into *n-samples*, followed by classification trees being fit on all training samples *n*. To decide what the model will predict on out-of-sample test data, the model can take a majority vote by predicting what most models have decided (James et al., 2017, p. 318). Through this process, bagging gets rid of the high variance of a decision tree. The model is, however, not easily interpretable anymore, as a predictive process no longer can be illustrated. Another downside of bagging is if each modeled tree is highly correlated with the others. This can be the case if one of the variables has higher importance than the other according to the Gini Index, and hence producing many models which in large are very similar, the procedure of bagging will not be able to reduce the variance notably.

To cope with the problem of correlation between trees, a modelling procedure known as random forests can be used, illustrated in the left of **Figure 1**. Random forests are in essence bagged decision trees, but with a process of decorrelating the trees. The process is done through randomization of possible predictors *m* from the full sample of predictors *p*, with a fresh sample at each split (James et al., 2017, p. 319). The value of *m* (the number of predictors at each split from the sample of predictors *p*) can be set to $m \approx \sqrt{p}$, which means most of the predictors are not considered for most of the splits. Through this random splitting process, each tree will be different from the others, and the problem with correlation between trees is eliminated. While random forests can become more accurate than bagging methods, they can also be considered even more complex. In contrast to a simple decision tree, the predictive process of a random forest model can no longer easily be explained, at least on an individual level. Hence, random forests can be considered black-box models because they are difficult to interpret and explain (Palczewska, Palczewski, Marchese Robinson et al., 2013).

The last approach we wanted to discuss which improves accuracy from a decision tree is boosting. Boosting is like bagging, but instead of fitting different trees to a large sample of training data, boosting works by building one tree and improving on it for each iteration. According to James et al. (2017), each tree is built by information from a previously grown tree, and boosting improves the model by fitting the decision tree to the residuals of the model. The three main tuning parameters of boosting are the number of trees B, the shrinkage parameter λ (the rate of learning), and the number of splits in each tree. In contrast to bagging, a boosting model can potentially overfit. To avoid this, one can use cross-validation techniques or by tuning the model. We discuss tuning and cross-validation in section 4.2.1 when we introduce a variant of boosting known as xgboost. Like bagging and random forest, boosting models are very complex and can be tricky to interpret and explain. They do however oftentimes come with the upside of better accuracy over simpler models due to lower bias.

2.2 Importance of Model Interpretability

Doshi-Velez and Kim (2017) argue that demand for interpretability rises from incompleteness in the problem formulation. This incompleteness can be referred to as the gap between model formulation and the actual problem, which creates a barrier between optimization and evaluation. To improve the point of importance for interpretability, we propose a general situation. A machine learning model is fitted on a training set and predicts accurately on a test

set. However, when taking the model to so-called unseen data, accuracy drops. Assume that the model is a black box, meaning that we input data and get output without knowing anything about the process. So, understanding why the predictions are poor is very difficult. Now, instead, assume that the model is an interpretable model. The user can now explain the model behavior and find the reason why predictions are poor (Freitas, 2014). This situation illustrates a reason for why interpretable models are desired. Bias can be picked up, and the incompleteness in the model formulation can be reduced.

There are also other reasons why interpretable machine learning is in demand. The need for explanations can be traced back to scientific understanding and curiosity (Doshi-Velez & Kim, 2017). Humans have always wanted to gain knowledge. So, if there are some things humans do not understand, they seek explanations. In addition, they argue safety measures are regarded as a reason to illustrate the importance of interpretable machine learning. Since machine learning models often are used on real-world problems, it is essential that models are learned to be error-free. It is also necessary with interpretable models to increase social acceptance (Molnar, 2020, p. 22). These models are being used more frequently in daily life. To obtain the full value of the models, humans need to have a shared perception. Carvalho et al. (2019) argue that in most cases what a prediction tells us is not enough and that there is also a need for an explanation behind predictions. Especially, in high-stake decisions where errors and mistakes could result in biased decisions, you can end up with severe consequences. For an uninterpretable system, the affected people will be left with no explanation.

It is expected that machine learning models will have real-world problems (Carvalho et al., 2019). A normal saying is that “a model is as good as its training data”. This is because if the training data contains a biased distribution, the model could pick this up. Carvalho et al (2019) argue that this could lead to discrimination and thus unfairness. It is important to incorporate modern standards and ethics in training models. In America, a machine learning model was used to perform risk assessments of inmates (Angwin, Larson, Mattu et al., 2016). The model found African Americans more likely to commit future crimes due to the color of their skin. The prediction, therefore, contained a bias and treated African Americans unfairly. Without an explanation framework, this bias would be difficult to detect. There are two reasons for bias to be picked up by a model. Either the data is biased, or parameters used for model tuning are wrong (Carvalho et al., 2019). Explanations could possibly disclose such bias so the model

can be adjusted. Explanations can also ensure that predictions are fair by other problems in the model. Two other reasons, ethics and regulations will be elaborated in detail in section 2.2.1.

The reasons listed above prove the point of importance for interpretable machine learning. There are however situations where the user is satisfied with high predictive accuracy without the need to understand the model (Freitas, 2014). Doshi-Velez and Kim (2017) argue there generally are two situations where interpretability is of less importance. The first situation is when there are no consequences for incorrect results. The other is situations in well-known systems where the user trusts the system so there is no reason to verify the results. Examples of situations with less importance of explanations include aircraft collision avoidance systems and postal code sorting.

2.2.1 Regulations and Law in Machine Learning

In recent years, interpretable models have seen increased public awareness. Ethical standards and regulations have been developed to make sure machine learning models are verifiable, accountable, and transparent (Carvalho et al., 2019). This section will focus on what these regulations and standards are and explain the implications it makes on machine learning models.

Ethics

An independent group from the European Commission (High-Level Expert Group on AI, 2019) developed “Ethics Guidelines for Trustworthy AI”. Their report lists requirements for a trustworthy AI. They argue AI should be respectful of laws, regulations, and ethical principles. They argue the pillars of AI to be law, ethics, and robustness. AI should be technically and socially robust, which means that small changes do not cause large consequences. Furthermore, the High-Level Expert Group on AI (2019, p. 11) present four ethical principles for a trustworthy AI. The principles are respect for human autonomy, prevention of harm, fairness, and explicability. When talking about the importance of interpretable models, we discuss explicability further.

Explicability is essential for a user to be able to trust AI systems (High-Level Expert Group on AI, 2019, pp. 13, 18). Transparency is one of the requirements for explicability and for AI

to be defined as transparent, it needs to be openly communicated and have the possibility of explanation for those affected. The group states that both the technical processes and the related human decisions must be explained. This implies that accuracy might be reduced to incorporate explainability in a model unless explainability can be achieved through other means like explanations methods. When people's lives are affected by an AI system it should be possible to demand an explanation for the decision-making. Instead of lowering accuracy through a simpler model, explanation methods could be a good addition to a complex model.

Regulations

As well as the ethics of an explainable machine learning model, regulations have been imposed on such models (Carvalho et al., 2019). These regulations aim for algorithmic decisions to be verifiable, accountable, and transparent. The European General Data Protection Regulation (European Parliament, 2016) is such a regulation, enforced in 2018. As argued, transparency means that there should be a possibility of explanation. With this regulation, data subjects have the right to get explanations of decisions made by algorithms. Data subjects refer to actors affected by decisions made by AI. The right for an explanation can be distinguished in two parts (Goodman & Flaxman, 2017). The first part is the right for access and notification, meaning that the data subject has the right to access and get notified about data collected. The other part is freedom for the data subject, meaning that, as a safeguard, the data subject has "the right to obtain human intervention" (Goodman & Flaxman, 2017). An important note here is that not all decisions need to be explained, but you must be able to explain decision-making upon request. To satisfy these requirements, we will now present some explanation methods used for this purpose.

2.3 Explanation Methods

There are two main categories of explanation methods: local and global explanations (Aas, Jullum, et al., 2021). The global approach aims to explain the model by calculating which features are important for the model. Aas, Jullum, et al. (2021) states that local explanation methods on the other hand will explain how the features in the model influence a specific prediction. Complex models often behave differently from simpler models, meaning global explanations are not sufficient for individual predictions. Explanations are either model-specific or model-agnostic. Model-agnostic models imply that an explanation method can be

used to explain many different models, while a model-specific explanation method will only work for one specific model or a group of models.

2.3.1 Local Explanation Methods

There are three main model-agnostic methods (Aas, Jullum, et al., 2021). The first method is explanation vectors, proposed by Baehrens, Schroeter, Harmeling et al. (2010), this method provide a local explanation to any classifier. The explanation method finds features relevant for individual predictions and is able to detect patterns that global explanations don't pick up (Baehrens et al., 2010). There are multiple definitions of explanation vectors, and each definition could result in a different explanation. Explanations presents the features relevant for individual predictions. Different classifiers that agree on all labels would also be explained similarly. If they do not agree, however, there will be different explanations. This is natural since they are two different classifiers (Baehrens et al., 2010). Explanation vectors assumes stationarity in the data, whereas non-stationarity cases should be dealt with. This can be done by adding additional measures to the method. A drawback with the method is that it applies to classification problems but not to other problems.

The second method is local interpretable model-agnostic explanations (LIME), as proposed by Riberio et al. (2016a). This method can explain predictions for any classifier or regressor. Models that only can be interpreted globally could be approximated by the general framework of LIME to provide local explanations. It provides faithful explanations regardless of the machine learning model. Also, it provides explanations which can be interpreted by both experts and non-experts. Suppose humans for example understand a decision tree better than regression. While the model could come from a regressor, with LIME it could be explained by a decision tree. LIME is therefore a popular explanation method. Riberio et al. (2016a) also claim LIME make users trust individual predictions by its explanations. They argue that trust is essential for effective human interaction and to obtain trust, explanations of individual predictions are important. The method allows users to influence in assessing trust in the model. It is also functional for image recognition which most explanation methods lack. A drawback with the method is the lack of theoretical properties (Riberio et al., 2016a). The method does not take the dependent structure into account either. Due to this, LIME will not be sufficiently accurate in real-world situations. Aas, Jullum, et al. (2021) argue the method to be inconsistent for this reason.

The third main model-agnostic approach is Shapley values which is based on cooperative game theory (Shapley, 1953). In model explanation, this method is the only method with a strong theoretical foundation (Aas, Jullum, et al., 2021). It also has unique properties which is important in a prediction setting. Shapley values is our preferred method to use for local explanations and will be further presented in chapter 3.

2.3.2 Global Explanation Methods

Global explanation methods can be used to increase transparency in a black-box model and may even detect potential weaknesses in the model (Fisher, Rudin, & Dominici, 2019). There are several global explanation methods, both model-agnostic and model-specific. A model-agnostic method is permutation importance (Strobl, Boulesteix, Kneib et al., 2008). This method aims to estimate the effect when a feature is missing. The prediction accuracy is measured before and after the permutation of features. Another preferred model-agnostic approach for explaining classification tree problems is the Gini Index (James et al., 2017, p. 312). This can be formulated as the following:

$$G = \sum_{k=1}^{K} \hat{p}_{mk}\left(1 - \hat{p}_{mk}\right). \quad (1)$$

This formula is a measure of the variance of all classes K. The $\hat{p}_{mk}$ is the proportion of training observations in node *m* and from class *k* (James et al., 2017, p. 312). If $\hat{p}_{mk}$ takes values close to one or zero, the Gini index also takes a small value, thus, this a measure of node purity. Consequently, a small Gini value refers to a high node purity in a classification tree and observations are mainly classified in one category.

The Gini index can be used for many purposes, one of those is explanation. By taking the mean decrease in Gini index for each variable, each variable will be given an importance value (James et al. 2017, p.318). As this explanation is used to explain features based on all predictions, we can define the method as a global explanation method. For known machine learning models, there are often model-specific methods to explain from a global perspective. Feature importance is a useful tool to assess which features are the most essential overall. A feature importance does not say anything about what the feature values should be though. In model building, feature importance can be a useful tool. Suppose that the model needs to be restricted in the number of features. To select the right features, Gini index could be used.

According to Lundberg (2018), however, the Gini Index is inconsistent. He states the Gini index is biased to contribute importance to lower splits. In cases where the first split is changed, importance accuracy could decrease and thus lead to inconsistent results. This drawback is undesirable and a reason to seek other global explanation methods.

3 Shapley Values

In this chapter we will discuss the theory of Shapley values. First, the general idea behind the method and the properties will be presented in section 3.1 and 3.1.1. Then, in section 3.1.2 we will extend the theory to discuss how Shapley values are used to explain predictions in a prediction setting. Section 3.2 extends on the theory of Shapley values into the Kernel SHAP method, which reduces the computational problem. Finally, in section 3.2.3, Kernel SHAP is further extended to include dependency between variables. The dependency extension makes the method relevant for explaining predictions for real-life applications.

3.1 Shapley Values in Detail

Shapley values is a cooperative game theory developed by Lloyd S. Shapley in 1953 (Shapley, 1953). The general idea is that M players in a cooperative game is trying to maximize a payoff. In formula (2) below, $S \subseteq M = \{1, \ldots, M\}s$ is the subset with $|S|$ players and $v(S)$ is the contribution function which converts subsets of players to numbers. This is needed to calculate a numerical Shapley value. The contribution function can be defined as a value S players receive together without the rest of the players in M. The worth can be seen as the total sum of payoffs expected by S with cooperation (Aas, Jullum, et al., 2021). The payout should be assigned to players, depending on the players contribution to the total payout. So, we can define $\phi_j(v)$ as the value-added when a player *j* comes into a team and this team is averaged over all possible coalitions (Frye, de Mijolla, Cowton et al., 2020). The result is a weighted mean called Shapley values. Shapley values is a method to assign the gain to players with the assumption that the players collaborate. Therefore, as stated by Aas, Jullum, et al. (2021), ϕ_j can be referred to as the Shapley value player *j* gets, calculated by the following formula:

$$\phi_j(v) = \sum_{S \subseteq M \setminus \{j\}} \frac{|S|!\,(M - |S| - 1)!}{M!} \big(v(S \cup \{j\}) - v(S)\big), \mathrm{j} = 1, \ldots, \mathrm{M}. \qquad (2)$$

3.1.1 Shapley Properties

There are four properties which Shapley values are unique to satisfy compared to other methods. The properties are Efficiency, Symmetry, Dummy and Linearity. The properties are considered to give an even distribution and a fair payout.

Efficiency

The efficiency property ensures that the total gain is distributed (Aas, Jullum, et al., 2021). The sum of all contributions from the players must be equal to the difference between the payout and the average payout. This is illustrated with the following formula from the article by Aas, Jullum, et al. (2021):

$$\sum_{j=0}^{M} \phi_j = v(M). \tag{3}$$

Symmetry

The next property is symmetry. This property enables two players that contribute equally to have the same Shapley value. This means that if player j and k contribute the same to the payout, then $\phi_j = \phi_k$. This gives Shapley values the quality of being fair.

Dummy player

If a player does not contribute to changing the total payout in any of the coalitions, it should have the Shapley value of zero. Therefore, if player j does not impact the payout in any coalitions, then $\phi_j = 0$.

Linearity

The linearity property means that if a payout is to be explained by a combination of features, each feature can be given a Shapley value.

3.1.2 Shapley Values in a Prediction Setting

Shapley values can be used in explaining predictions by machine learning (Aas, Jullum, et al., 2021). In this setting, the total payout is considered the prediction and the players are the feature values. Shapley values are thus a measure of how the features contribute to the prediction. In the explanation setting, talking about "game" is transmitted to predicting in a dataset. "Players" are the features contributing to the gain, where the "gain" is the difference between prediction and the average off all predictions.

To understand Shapley values for prediction explanation in detail, we can first look at a prediction model $f(x)$. The contribution of the feature j can be formulated by the mean effect of feature j and the weight corresponding to the feature. To define the contribution in

prediction terms, we must calculate the difference between the predicted value and the average predicted value. The $v(s)$ function is an assumption of a contribution function which maps players to real numbers (Aas, Jullum, et al., 2021).

Suppose we have 3 features in our model, so $\mathcal{M} = \{1,2,3\}$. The possible subsets will thus be 2^3 equal to 8. By applying equation (2) to the number of features, the Shapley value calculation for feature ϕ_1 will be:

$$\phi_1 = \frac{1}{3}\big(v(\{1,2,3\}) - v(\{2,3\})\big) + \frac{1}{6}\big(v(\{1,2\}) - v(\{2\})\big) + \frac{1}{6}\big(v(\{1,3\}) - v(\{3\})\big) + \frac{1}{3}(v(\{1\}) - v(\emptyset)), \tag{4}$$

where $v(\{1,2,3)\}$ refers to the contribution given the coalition of all three features. From the equation we can see each calculation step must include subsets of features with the feature in it, subtracting all other features included in the coalition. In addition to each feature contribution, we also need to calculate the non-distributed gain $\phi_0 = E[f(x)]$. This value is defined as the fixed payoff without contributions from any features (Aas, Jullum, et al., 2021). It can be viewed as the starting point before each feature contributes to either direction. To calculate the other two Shapley values for this prediction, a similar calculation would be done with relevant coalitions two more times to obtain three Shapley values (one for each feature).

In a prediction setting, sets define our model as $f(x)$ and the corresponding prediction as $f(x^*)$. Going forward, we will start by decomposing $f(x^*)$, as stated by Aas, Jullum, et al. (2021):

$$f\,(x^*) \;=\; \phi_0 \;+\; \sum_{j=1}^{M} \phi^*{}_j. \tag{5}$$

In equation (5), $x = x*$ is a specific feature vector for the model $f(x^*)$. We can see the non-distributed gain ϕ_0 adding up with the sum of all $\phi^*{}_j$ for all features M. The difference between the prediction $y^* = f(x^*)$ and the global average prediction is explained by Shapley values. For every prediction $f(x^*)$ we compute with our prediction model, it will be explained over different sets of ϕ_j values. If we dig deeper in the contribution function $v(S)$ of a subset S when we only know the value of the subset S, the contribution function should be equal to $f(x^*)$ for the feature values $x_S \;=\; x_S^*$. This subset is given by equation (6) below.

$$v(S) \;=\; E[f(x)|x_S \;=\; x_S^*]. \tag{6}$$

Advantages and Disadvantages of Shapley Values

There are advantages and disadvantages of the usage of Shapley values in a prediction setting. The Shapley theory with its properties, presented in 3.1.1, is a clear advantage which distinguishes Shapley values from other methods (Aas, Jullum, et al., 2021). In a prediction setting, these four properties give a fair and even distribution of Shapley values for the different features. In addition, the linearity property opens for local explanation methods using Shapley values. So, each feature can be given a Shapley value to explain its influence on the prediction.

As mentioned in section 2.2.1, there are several ethical and regulation type requirements for AI to be trustworthy. Models must be transparent and data subjects have the right to get an explanation. As some machine learning models are difficult to interpret, Shapley values is a great tool for explaining the model from a global and local perspective. All Shapley properties contribute to a fair explanation in line with the requirements. In addition, Shapley values open for explanations of individual predictions and not only compares predictions in a dataset. Individual explanations are valuable for data subjects that are affected by decisions made by prediction models.

An inevitable disadvantage of Shapley values is that it requires a lot of computations to calculate (Aas, Jullum, et al., 2021). This is because there is potentially a very large number of coalitions to be computed. Suppose we have 15 variables, then the number of potential subsets will be $2^{15} = 32.768$. With more variables, this number will grow exponentially. Another disadvantage with Shapley values is that it can only be used when features are independent. This leads to an area of usage that is unrealistic and small since the most of real data is excluded. There are however methods that have the possibility of dealing with all these problems to some extent, such as the Kernel SHAP approach.

3.2 Kernel SHAP

Kernel SHAP (Shapley additive explanations) is a method based on Shapley values to explain individual feature values to a prediction (Lundberg & Lee, 2017). There are several ways of computing SHAP, we will dig deeper into Kernel SHAP. In this method, each feature will show a value of importance to a single prediction. To deal with the computational complexity of Shapley values, Kernel SHAP approximates calculations. By approximating weighted sum

in Kernel SHAP, this issue will be reduced. Kernel SHAP provides faster computational time and estimates close to real Shapley values.

3.2.1 Kernel SHAP in Detail

We now assume that $v(S)$ is known and aim to compute an alternative formula for Shapley values. There are many ways of doing this, one of which is a weighted least squares problem. This can be presented as minimizing these formulas, as stated by Aas, Jullum, et al. (2021):

$$\sum_{S \subseteq M} (v(S) - (\phi_0 + \sum_{j \in S} \phi_j))^2 k(M,S), \tag{7}$$

where $k(M,S)$ are the Shapley kernel weights and is equal to:

$$(M - 1) / (\binom{M}{|S|} |S| (M - |S|)). \tag{8}$$

Formula (8) can be rewritten to:

$$(v - Z\,\phi)^T W (v - Z\,\phi), \tag{9}$$

where Z is a matrix of all possible combinations of the M features. Also, *v* is a vector of $v(S)$ and W is a $2^M \, x \, 2^M$ matrix with $k(M,|S|)$. Lundberg and Lee (2017) prove that Shapley values can equate to:

$$\phi = (Z^T W Z)^{-1} Z^T W v. \tag{10}$$

When M contains many features, we approximate the formulation using weighted least squares. Since Shapley kernel weights have different sizes, most of the subset's S contribute marginally to Shapley values (Aas, Jullum, et al., 2021). These subsets are included in the rows of Z. To approximate a Shapley value, we sample a subset D of M from a probability distribution which follows Shapley weighted kernel. Thus, Aas, Jullum, et al. (2021) state formula (11).

$$\phi = [\,(Z_D^T W_D Z_D)^{-1} Z_D^T W_D\,] v_D = R_D v_D. \tag{11}$$

With this approximation, the $(M+1) * |D|$ matrix R_D only needs to be computed once which reduces computational complexity.

The second part of the Kernel SHAP method is computing $v(S)$. All possible feature subsets in Z are needed to compute the $v(S)$ (Aas, Jullum, et al., 2021). We previously defined $v(S)$

as the value contribution for a certain subset S, and as we recall, the contribution function is defined by:

$$v(S) = E[f(x)|x_S = x_S^*].$$

The features in subset S are x_S and what we want to explain is the feature vector x^* where x^*_S is the subset S of this vector. This means that the contribution function will give the expected predicted value on the assumption features in S take the value x^*_S. To compute the $v(S)$ function, we need to do it for different subsets S. If we assume $x_{\hat{S}}$ is a part of x but not a part of x_S we can write the formula as stated by Aas, Jullum, et al. (2021):

$$\begin{aligned} E[f(x)|x_S = x_S^*] &= E[f(x_{\hat{S}}, x_S)|x_S = x_S^*] \\ &= \int f(x_{\hat{S}}, x_S^*)\, p(x_{\hat{S}} \mid x_S = x_S^*) dx_{\hat{S}}, \end{aligned} \tag{12}$$

where, given that $x_S = x*_S$, then $p(x_{\hat{S}}| x_S = x_S^*)$ is the conditional distribution of $x_{\hat{S}}$. This distribution is necessary to compute $v(S)$. The standard Kernel SHAP method uses the training set to compute an empirical distribution of x and can be formulated as (Aas, Jullum, et al., 2021):

$$v_{KernelSHAP}(S) = \frac{1}{K}\sum_{k=1}^{K} f\left(x_{\hat{S}}^k, x_S^*\right). \tag{13}$$

Assuming that f is the function of a prediction model and $x_{\hat{S}}^k, k = 1, \ldots, K$ are samples from the training set.

3.2.2 Advantages and Disadvantages of Kernel SHAP

As argued by Aas, Jullum, et al. (2021), independence between features is very rare in real data. Therefore, they extend Kernel SHAP to incorporate dependence between features. $P(x_{\hat{S}}| x_S = x*_S)$ is the dependence assumption in Kernel SHAP. Aas, Jullum, et al. (2021) state that this is a rough assumption to an otherwise solid method. They propose to relax this assumption and instead estimate it directly and generate samples from the distribution. Their results show that the extension performs better than the original and other methods. The extension will thus remove the disadvantage of Kernel SHAP and will be presented in the next section.

Lundberg and Lee (2017) also found much stronger agreement between human explanations and SHAP than with other explanation methods. This shows the advantage Shapley values

have over other methods. The quality is important to use in our thesis since we aim to understand prediction models better. Even though Kernel SHAP approximate values to deal with the computational problems, exponential growth is still a challenge. Using Kernel SHAP on many variables will still take up a substantial amount of computational resources due to the number of subsets that are calculated in the process explained in 3.1.2. Hence, the computational disadvantage of Shapley values extends to Kernel SHAP to some degree.

3.2.3 Kernel SHAP with Dependent Variables

Since we will want to compute Kernel SHAP on a real-world data set in the next chapter of this thesis, we can suspect there will be dependency between variables. This will be further investigated in chapter 4. With dependent variables, the method we will use to explain predictions is an extension of the standard Kernel SHAP method. This extension incorporates dependency between variables, contrary to the standard Kernel SHAP method. In real data sets, variable independence is very rare, and the standard Kernel SHAP method performs poorly. This is argued by Aas, Jullum, et al. (2021) in experiments comparing the approaches. They used both mean absolute error and a skill score measure to evaluate performance in explanation methods. The incorporation of dependence is done by estimating the dependence assumption directly and thereafter generate samples for this distribution. This way, the distribution will be generated dependent of each feature.

There are four approaches for estimating the dependence assumption according to Aas, Jullum, et al. (2021). Multivariate Gaussian distribution, Gaussian copula, empirical conditional distribution, and a combined approach. We will elaborate more on these approaches and investigate which is the most applicable for our case data during the case study in chapter 4. As we know, $p(x_{\hat{S}} | \, x_S = x_S^*)$ is the dependence assumption in Kernel SHAP. As this is a poor assumption, it could be replaced with a distribution that reflects real-world data better. The multivariate Gaussian distribution replaces the marginal distribution $x_{\hat{S}}$ with a Gaussian distribution (Aas, Jullum, et al., 2021). If our case data has a distribution which is similar to multivariate Gaussian, we should use this approach. Suppose our features are far from this distribution, but our margins are close, then Aas, Jullum, et al. (2021) argue a Gaussian copula distribution may be better. Copula is an approach to isolate dependency between features (Haugh, 2016).

However, if neither the features nor the marginal distribution is close to Gaussian in our data, we cannot use such an approach. Empirical conditional distribution is a non-parametric approach which is suitable if the dependence structure and marginal distributions of x are far from normally distributed (Aas, Jullum, et al., 2021). Since there are few such approaches, they developed this method. The general idea behind the method is to sample x_S as close to x_S^* as possible and use this in the new distribution. The mathematics behind the method is complex and out of the scope of this thesis. For a more thorough review, we recommend the paper by Aas, Jullum, et al. (2021). This approach could also be combined with either multivariate or the copula approach when dimensions are higher and there is a risk of information loss. The choice of distribution depends on the distribution of features. Therefore, it is important to investigate feature distribution when conducting Kernel SHAP without the independence assumption $p(x_{\hat{S}} | \, x_S = x_S^*)$.

4 Case – Predicting Financial Distress in Norway

The case chapter is divided into three main sections. Section 4.1 introduces the case, why it is interesting for our thesis, the data treatment process, and the differences in output from the thesis we refer to. Section 4.2 introduces the chosen machine learning algorithms which will be used for the rest of the thesis both in a predictive setting and to be explained. In section 4.3 we present the prediction results.

4.1 Introduction to the Case

To efficiently explain individual predictions, there was a need for a prediction problem with available data to perform analysis on. There was also a need for the case to be interesting with good reasoning behind the predictions. At last, we ended up with a master thesis from 2019 with the title "Predicting Financial Distress in Norway" by Zhang and Ye (2019). In their thesis they try to predict which companies are likely to enter financial distress within the next two years in Norway.

The reasons for selecting this master thesis as a case for our thesis were many. First, Zhang and Ye (2019) contribute with a solid and available groundwork allowing us to take their work further. Their predictive models validate for approximately 60% accuracy when optimizing according to a true positive rate equal to the true negative rate, given a ~90/10 imbalanced dataset. We find this reasonable given the available data. Another point we found interesting with their paper was that financial distress is rare for companies but crucial for those affected. It is therefore important to detect danger-signs of distress, which is relevant both for creditors and for the companies themselves. We consider predictive models to be valuable for detecting signs of financial distress. When distress-signs are detected, it is vital to understand why companies are predicted distressed. We believe the interpretability aspect is essential for this case because decision-makers may use such models to make decisions. Companies affected by decisions made by machine learning algorithms may demand explanations behind the decisions. Therefore, really understanding *why* a model predicts a company as healthy or distressed can be useful for creditors such as banks or financial institutions, and for the companies involved.

Zhang and Ye (2019) use the mean decrease in the Gini index as a method to explain the global feature importance for the black-box random forest model. As discussed in chapter 2, when applying a complex model to a problem, the Gini index can be considered biased and inconsistent unlike Shapley values. We therefore wanted to provide a sufficient and complete explanation framework to obtain concrete and understandable explanations. Their thesis did neither consider individual predictions nor dive deep into interpreting a black-box model. We have taken on this task, and it will be the focus in the coming sections.

4.1.1 Data Treatment

Zhang and Ye (2019) did a substantial data cleaning process before they developed their prediction models. Our aim was to try to replicate the results and then explain individual predictions. We reviewed the same data set from the Centre for Applied Research at NHH (SNF) and followed Zhang and Ye's data wrangling.

To clean the data, we started with importing and merging data on company accounts for the years 2013-2015 and remove data on bankrupt companies in 2013. Then, we calculated financial ratios such as ROA and ROE to produce our dataset. Complete list of features are presented in **Table 1**. Next, a distress rating was constructed based on companies going bankrupt in 2014/2015 or on companies having a C rating for two years in a row. We continued by downsizing the data sample to 40.000 random companies to reduce computational complexity. After that, extreme outliers were converted to NA's according to values which are outside the 1^{st} or 3^{rd} quantile multiplied by 10 times the interquantile range. Finally, we reduced computational complexity by removing variables. Variables were chosen based on relevance, correlation and significance from the thesis by Zhang and Ye (2019). With some testing we found 9 variables to be the amount which is feasible to calculate Shapley values for in chapter 6.

For complete information and reasoning behind the data cleaning, we will refer to the thesis by Zhang and Ye (2019). Furthermore, during the data cleaning we had to make some assumptions and our own assessments, which resulted in a slightly different outcome. **Table 1** illustrates the remaining variables after the cleaning process. Each of the selected variables are calculated based on famous financial distressed models such as Altman's Z-scores

(Altman, 1968) and Ohlson's O-scores (Ohlson, 1980). We have adopted variable names from Zhang and Ye (2019) and changed them for intuitive reasons.

Variable	***Definition***	***Variable name***
Y	Distressed or healthy	Y: Healthy / Distressed
X1	Net income / Total assets	X1: ROA
X2	Current assets/ Current liabilities	X2: Current ratio
X3	Working Capital /Total assets	X3: Working capital / Total assets
X4	Retained earnings /Total assets	X4: Retained earnings / Total assets
X6	Sales / Total assets	X6: Asset turnover
X9	Current assets / Total assets	X9: Current assets / Total assets
X10	Net income / Equity	X10: ROE
X13	Debt /Equity	X13: Debt / Equity
X17	Log of total assets	X17: Log of total assets

Table 1. Case variables

4.1.2 Descriptive Statistics of Data

As a result of our data treatment, we have obtained a clean data set. We will now explore descriptive statistics on our case data to learn more about its features. It is also important for us, as mentioned in section 3.2.3, to find out if there are dependencies between variables. If this is the case, then the extension of Kernel SHAP is appropriate. To prove dependency between features, correlation can be studied (James et al., 2017, p. 70). To study correlation, we have plotted a correlation matrix with heat mapping in **Figure 2**.

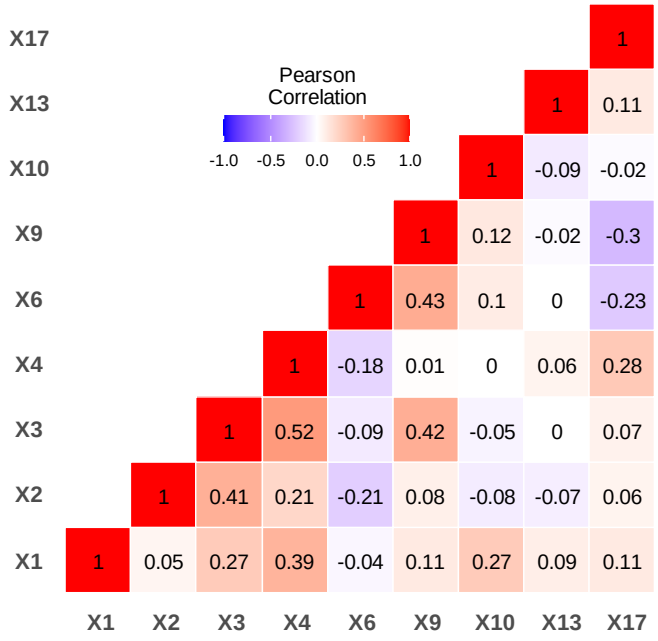

Figure 2. Feature dependence investigation with correlation

In the figure we can see that even though there is no strong correlation (due to highly correlated variables filtered out during cleaning to reduce computational complexity), there is still some correlation between variables. The strongest correlation can be observed for variables *X3: Working capital / Total assets* and *X4: Retained earnings / Total assets* (0.52) which indicates the variables are the most dependent on each other. There are some relationships which seem to be independent though. The variable *X13: Debt / Equity* is independent of *X3: Working capital / Total assets* and *X6: Asset turnover*. This leads us to the conclusion that there is dependency between many of the variables, which means the extension of Kernel SHAP is appropriate. Next, to choose dependency distribution for this method, we must study feature and marginal distributions. As we recall, there are four relevant approaches to choose from: multivariate Gaussian distribution, Gaussian copula, empirical conditional and a combined approach.

To investigate dependency, histograms for all 9 features are plotted in **Figure 3** below. The figure can be inspected to choose distribution during Shapley value calculation in chapter 6. From the figure we can observe signs of Gaussian distribution for some features, like for the variable *X17: Log of total assets*. The challenge with selecting a dependency structure such as multivariate Gaussian and Gaussian Copula is that the distributions need to be similar for all variables, or you can end up with inaccurate explanations. From the figure we observe *X3: Working capital / Total assets* and *X4: Retained earnings / Total assets* to be skewed, while *X2: Current ratio* and *X6: Asset turnover* are heavy tailed. An advantage of using the empirical conditional distribution method is that features can be different. It does not assume a pre-defined distribution but rather estimate each distribution. As the features have varying distributions, the empirical conditional approach will be our choice going forward. This is because the empirical approach performs better than other approaches with varying distributions, as illustrated by (Aas, Jullum, et al., 2021) during their simulations.

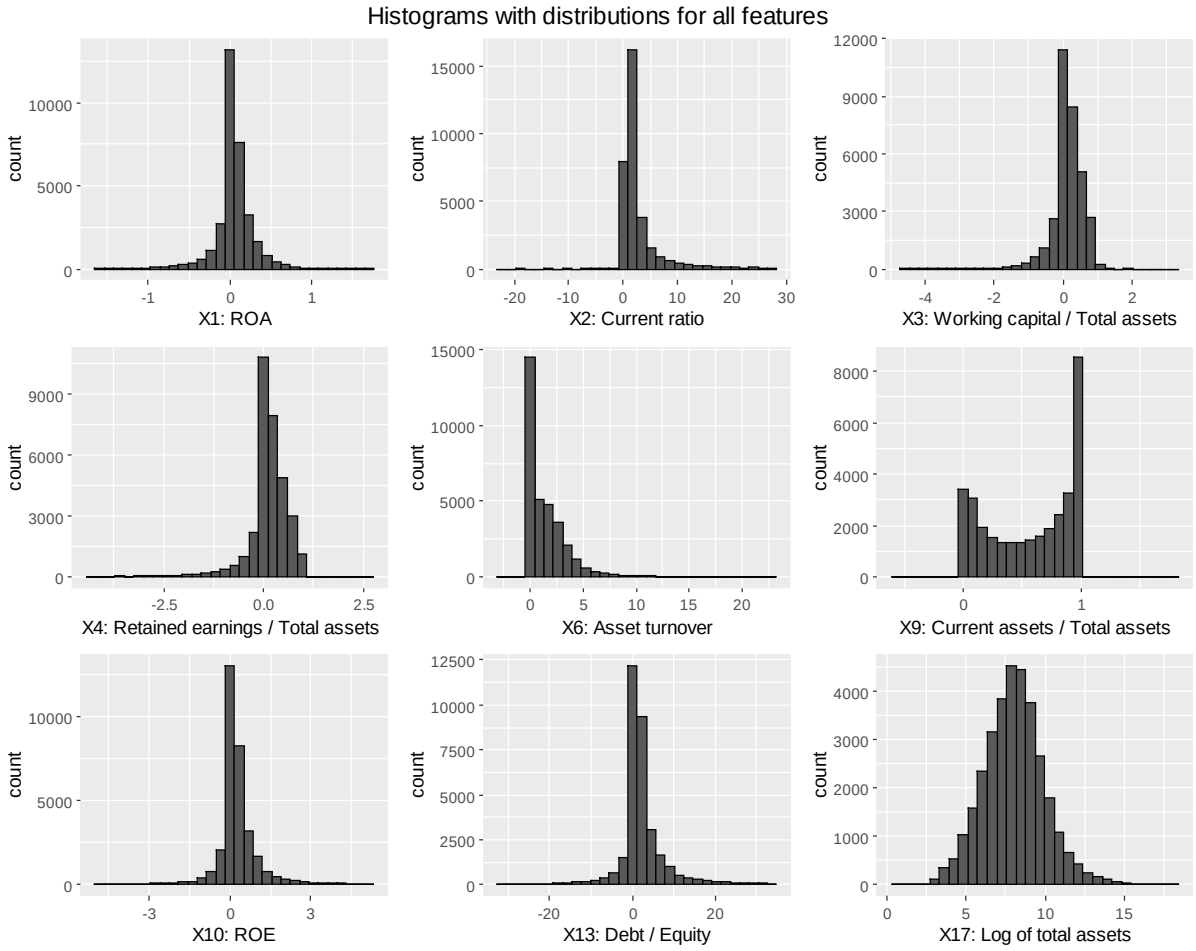

Figure 3. Histograms for all features

Another point to consider is that most companies have slightly positive measures. This can be seen in histograms from **Figure 3** but also by studying summary statistics presented in **Table 2** below. From **Figure 2**, we can observe the mean of the dependent variable *Y* to be 6.2%. This is the undistributed gain and means that without any features taken into consideration, there is a 6.2% chance for a company to enter distress. Two common variables in financial terms are return on assets (ROA) and current ratio. ROA is a measure on a company's profitability based on its assets (Zhang & Ye, 2019). Companies strive to have a high ROA as it is a sign of a healthy operation. *Current ratio* measures a company's liquidity and shows how well it can cover its current liabilities with its current assets. Having more current assets than current liabilities (ratio over 1) is important for financial health. *X3: Working capital / Total* assets is obtained by dividing working capital by total assets. This is a measure of how well-suited a company is for running daily operations. *X4: Retained earnings / Total assets* is a measure of assets funded by earnings (Zhang & Ye, 2019). This measure shows if a company is financed by retained earnings and is a sign of healthiness. *X6: Asset turnover* gives a

measure of how well a company can generate revenue from its assets. A high measure here is desired. *X9: Current assets / Total assets* is a liquidity measure, while return on equity (ROE) shows profit obtained from shareholders investments (Zhang & Ye, 2019). *X13: Debt / Equity* is a leverage measure and shows debt compared to equity. A high or negative ratio is considered risky for financial health. The last included variable is *X17: Log of total assets* which gives an indication of the size of the company. Historically, bigger companies tend to avoid financial distress.

We can also study the standard deviation, which tells us more about the spread of the distribution of the features. We can see that there is a large spread in the debt ratio and Current Ratio (*X13: Debt / Equity* and *X2: Current ratio*) in our data. The lowest and highest debt ratio is ranging from -31 to 34.3. This implies that there are some companies with negative debt or total assets. Negative total asset means that a company has more liabilities than assets. A negative ratio is very risky as it implies the company has more liabilities than assets, indicating a risk of bankruptcy (Fernando, 2021). Neither negative total assets nor a negative debt ratio is a good sign financially. We can also study the quantiles. The upper quantile tells us that 25% of the companies have a size (*X17: Log of total assets*) larger than 9.2, while 25% of companies have a size smaller than 6.8. Keep in mind that extreme outliers were adjusted during data cleaning which affected minimum and maximum values in the table.

Statistics	***Mean***	***St. Dev.***	***Min***	***PCTL (25)***	***PCTL (75)***	***Max***
Y: Healthy / distressed	0.062	0.242	0	0	0	1
x1: ROA	0.050	0.255	-1.600	-0.016	0.140	1.717
X2: Current ratio	2.706	3.893	-22.000	0.883	2.706	27.709
X3: Working capital / total assets	0.130	0.471	-4.611	-0.022	0.394	3.208
X4: Retained earnings / total assets	0.102	0.564	-4.326	-0.0002	0.391	2.642
X6: Asset turnover	1.474	1.951	-2.448	0.080	2.277	22.968
X9: Current assets / Total assets	0.579	0.367	-0.536	0.198	0.942	1.826
X10: ROE	0.274	0.852	-4.745	0.000	0.464	5.235
X13: Debt / Equity	2.238	6.033	-31.016	0.263	3.230	34.279
X17: Log of total assets	8.043	1.924	0.693	6.789	9.188	18.205

Table 2. Summary statistics

At last, we wanted to mention the data has a labeling for which sector the companies belong to. There are 10 different sectors, the largest being construction, followed by wholesale, finance, manufacturing and IT. The section labeling will be of specific attention to us during case analysis and sector comparison in chapter 6.

4.2 Modelling

In the thesis by Guang Na Zhang and Fan Ye from 2019, they used logistic regression and random forest models to predict financial distress. In terms of the interpretability/complexity tradeoff discussed in section 2.1, a logistic regression model can be considered a simple model, while a random forest model is complex. Going forward, we could choose to use the random forest model to predict financial distress. However, as this was already done in the thesis we refer to, we will introduce a newer and even more powerful method, known as extreme gradient boosting. This will help us to really illustrate the value of calculating Shapley values from predictions later in this thesis because the model is so difficult to understand. We will compare the model in terms of accuracy and interpretability to that of a simple decision tree.

4.2.1 Xgboost

Xgboost is a variation of boosting which stands for Extreme Gradient Boosting. Tree boosting is a popular and highly effective machine learning method (Chen & Guestrin, 2016). Xgboost as a tree boosting algorithm is renowned for execution speed and model performance and has from objective tests proven to outperform other implementations of ensemble methods such as gradient boosting and bagged decision trees in computation time (Brownlee, 2021). Brownlee (2021) also states it has also been the go-to algorithm to use in machine learning competitions as it outperforms other models in predictive accuracy.

To fit the xgboost model to the financial data, the following steps were taken. First, the data is split 80/20 into train and test data, the training part for model building and test data for final evaluation in section 4.3. The test-data will only be considered in section 4.3 when we present the results and is not part of the model building or tuning of hyperparameters. Next, we perform 5-fold cross validation on the training data with the function *xgb.cv* (Chen, He, Benesty et al., 2021). Cross-validation is used for several beneficial properties. It is beneficial for tuning of hyperparameters, helps to avoid overfitting, and it will give more accurate approximations to the true AUC as it considers all the training data for training and testing. By testing different hyperparameter values during k-fold cross validation, we end up with the parameters presented in **Table 3**. Explanations are derived from Chen et al. (2021).

Parameters	***Chosen***	***Explanation***
Evaluation metric	AUC	Evaluation metric for the xgboost model. By setting this to AUC we choose to optimize xgboost according to the AUC criterion.
Early stopping rounds	20	Will stop boosting iterations if the model does not improve for 20 rounds.
Eta	0.03	Controls the learning rate and can range from 0 to 1. Choosing a low value makes the model more robust to overfitting, but slower to compute.
Max depth	2	The maximum depth of a tree. Can be decreased or increased to see whether it improves the model or not.
Number of rounds	200	Number of boosting iterations.
Number of folds	5	Number of folds.

Table 3. Xgboost parameters

The evaluation metric is set to AUC to optimize the model according to this criterion. We chose AUC for its intuitive properties of considering different aspects of the confusion matrix. Early stopping rounds is set to 20, again to avoid overfitting. This means if AUC does not improve for 20 consecutive boosting iterations, then the model will revert to the best AUC result. When deciding *eta*, we tested different values and found the value of 0.03 to provide the best results. For max depth, different values were also tested, and two branches of a tree were found to provide the best result. At last, number of boosting iterations was set to 200 as we are working with a fast program on a relatively small dataset of about 26.000 training observations, and computation time is not a problem. We tried implementing more parameters during model tuning but found no significant improvement over the iterations. Below is **Figure 3** which presents the mean-AUC during 5-fold cross validation.

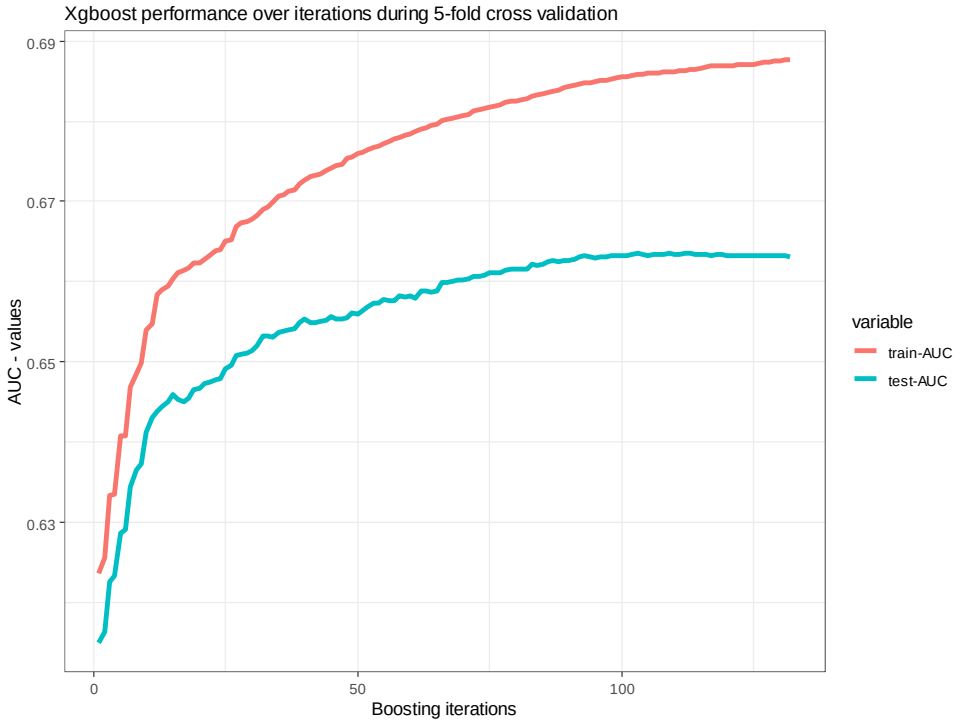

Figure 3. AUC results for xgboost during 5-fold cross validation

From the figure we observe increasing AUC values (y-axis) over xgboost iterations (x-axis). Both train-AUC and test-AUC starts with rapid increases, but then continues flattening out. After about 100 iterations, the model is about to start overfitting, observed with test-AUC completely flattening out. This is where the model stops due to the *early_stopping_rounds* hyperparameter, and we end up with mean train-AUC of 0.688 and mean test-AUC of 0.665.

4.2.2 Decision Tree

For comparison to xgboost in terms of performance and interpretability, we have chosen a simple decision tree as our baseline. Decision trees have the added benefit over xgboost of being able to provide the *why* to each explanation by studying the tree (Baehrens et al., 2010). As discussed in section 2.1.1, however, decision trees can be very inaccurate due to their high variance. In section 4.3 we will see how our decision tree performs in terms of accuracy during final evaluation.

By following the same validation set approach as for boosting, we split the data in the same 80% for training (to be used in the modelling process) and 20% for testing (to be used for evaluation in section 4.3). To build a decision tree, we will use functions from the R package

rpart (Recursive Partitioning and Regression Trees) (Therneau, Atkinson, & Ripley, 2019). According to documentation on CRAN, the package implements ideas and techniques found in CART by Breiman, Leo, Friedman et al. (1984). During modelling we use the default parameters in the model but will adjust for the imbalanced nature of the data to obtain a true positive rate equal to the true negative rate. This is done by including the *parms* parameter in the *rpart* function in R, aiming to give proper weighting to each of the two classes. Without this weighting, the model would be biased towards the majority class and ignore distress predictions. Next, we consider additional tree-pruning of the tree by investigating the complexity parameter (CP) with the function *printcp (Therneau et al., 2019)*. This is used to decide the optimal tree size by evaluating the cost of adding more variables. Cross validation error for the different splits is presented in **Table 4**.

Tree	***Cost complexity criterion***	***Number of splits***	***relative error***	***x-error***	***x-standard deviation***
1	0.153	0	1	1	0.024
2	0.037	1	0.847	0.871	0.020
3	0.011	2	0.810	0.847	0.017
4	0.010	3	0.799	0.835	0.017

Table 4. Tree splits and complexity parameter

We see from the table that "x-error" is lowest when $CP = 0.010$, which implies the lowest cross-validation error occurs when the model uses three splits (the current model). To test the condition, we could apply tree-pruning to the model and cross validate results for different test instances. However, as the differences in "xerror" are so small, we choose to go forward with this model.

4.3 Prediction Results

During k-fold cross validation with xgboost we found the hyperparameters which provided the best results. These parameters were then chosen to build a complete model on all the training data (80% of the data). In this section we will explore how the models perform during final evaluation when trying to try to predict the remaining 20% of the sample which have not been considered yet.

For evaluation, we have opted for the receiver operating characteristics curve (ROC curve). This curve compares the true positive rate (TPR) with the false positive rate (FPR) summarized

over all possible thresholds (James et al., 2017, p. 147). For our case, TPR refers to the probability that a company which will become distressed is predicted to be distressed, while FPR refers to the probability that a healthy company is wrongly predicted to become distressed. The area under curve (AUC) describes the accuracy of the model in terms of this tradeoff. The best possible AUC value is a value of 1, and for such a situation the ROC curve will hug the top left corner of the graph. For our case, an AUC of 1 would imply that the model can classify all distressed companies while not classifying any false positives. In **Figure 4** below, ROC curve for the decision tree and xgboost is presented.

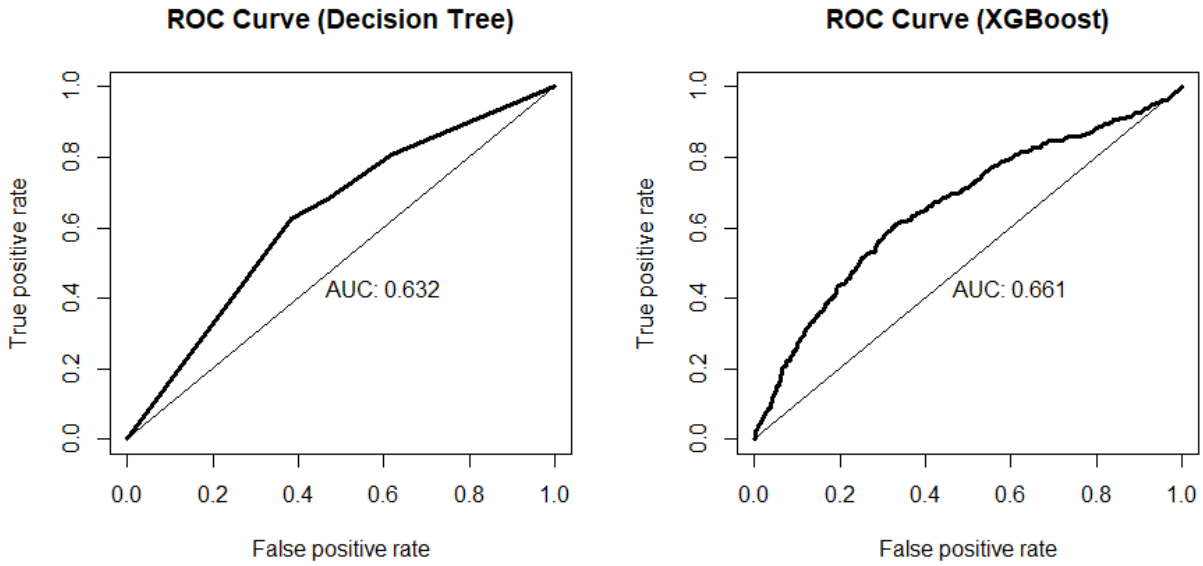

Figure 4. AUC results for the decision tree and xgboost

The y-axes in **Figure 4** show the true positive rate for different thresholds, while the x-axes show the false positive rates. The black lines through the diagonals indicate an AUC of 0.5. This is the worst-case scenario for the classification model, as it displays no discrimination, indicating the model has no ability to distinguish between distressed and healthy firms. For the decision tree, we observe an AUC of 0.632, while for xgboost we observe an AUC of 0.661. Hence, xgboost performs better at predicting healthy and distressed firms correct during evaluation.

When comparing the results to the Zhang and Ye's results, we consider AUC's for both models to be acceptable. We find the results to be according to our expectations since single decision trees can be simple and less accurate than a complex model such as xgboost. The results for xgboost is similar to the results in the thesis by Zhang and Ye (2019). We believe the AUC

for both models can be improved with additional tuning. Xgboost can be improved with optimization of hyperparameters, while the decision tree can possibly be improved with tree-pruning. Additionally, more variables can also be included to improve on the accuracy. For our part, however, additional tuning and data wrangling is beyond the goal of this thesis as we are our focus is on explaining the predictions, not optimizing accuracy.

4.3.1 Threshold for Distress

To decide whether the models have predicted companies as healthy or distressed in the coming sections, we must decide a threshold. A threshold can be tricky to decide because of the imbalanced nature of the dataset with 6.2% distressed companies. An easy way to achieve high accuracy would be to predict all companies to be healthy, which would achieve 93.8% accuracy, but this would wrongly classify all distressed firms as healthy. We consider the discussion in the thesis by Zhang and Ye (2019), and choose to opt for the same method as them, to decide a threshold according to a true positive rate (sensitivity) equal to the true negative rate (specificity). The threshold which gives this result is estimated to be 0.06 for xgboost and 0.5 for the decision tree. This means any companies with probabilities (prediction values) above these thresholds are predicted to be distressed, while companies with probabilities below the thresholds are predicted to be healthy. Selecting threshold for xgboost is a difficult task since small changes will impact many companies. When we talk about decision-makers in practice in this thesis, this is the threshold to use to decide loan offers or loan rejections. Companies with probabilities way higher than the thresholds can be considered as the model assigning high probability for distress. Very low probabilities indicate low probability of distress. Since we aim to use our most accurate model in decision-making, we will present summary statistics for xgboost in **Table 5**. Pay specific attention to the minimum and maximum values for the prediction object. For xgboost, the minimum probability for distress is 1.6% while the maximum probability is 40.5%. In the next chapter we will discuss the interpretation of these models, given the mentioned thresholds for a selected case company in financial distress.

Statistic	***N***	***Mean***	***St. Dev.***	***Min***	***Pctl (25)***	***Pctl (75)***	***Max***
prediction	6,611	0.064	0.038	0.016	0.038	0.077	0.405

Table 5. Summary of prediction object for xgboost

5 Interpretation of Prediction Models

As discussed in section 2.1, there is a tradeoff between interpretability and complexity for machine learning models. This chapter will explore this trade-off by trying to explain a prediction by the algorithms xgboost and decision tree for a chosen case-company. In section 5.1 the motivation and need for prediction explanations for the case study will be elaborated. Section 5.2 will discuss the choice of case sector and present financial ratios for the selected case-company. We will evaluate these ratios from an economic standpoint and a predictive standpoint. Section 5.3 will investigate the prediction the two chosen machine learning algorithms made for our case-company during evaluation in section 4.3. By investigating model plots, we will try to explain the prediction directly from both models. The explanations will show that a prediction made by the decision tree can be interpreted by studying the model directly, while a prediction by xgboost cannot. The chapter creates motivation for using the Shapley value framework as an explanation tool in chapter 6.

5.1 Case Motivation for Prediction Explanations

The cost of financial distress does not only affect the company itself, but also external stakeholders such as shareholders, creditors and investors (Zhang & Ye, 2019). In their thesis they argue that bankruptcy costs extend into a social problem because it leads to unemployment, losses on debt by creditors and increased volatility in the market. Hence, systems to scan and detect drivers of distress can benefit society on several levels. We believe there are two organs which may benefit especially from such systems.

The first is the government, which can benefit on a social level from detecting signs of financial distress. To cope with the mentioned market volatility, unemployment, and social costs, systems to help the government to detect signs of distress and act against them can prove valuable. In some cases, financial distress can lead to bankruptcy. Costs related to bankrupt companies affect a very wide range of parties (Branch, 2002). Due to this wide range of both direct and indirect parties affected, it is hard to give an exact measure of the magnitude of the cost of bankruptcy. Before going into bankruptcy, many companies are dealing with financial distress which relates to a substantial part of these costs. Financial distress is thus undesired for those directly affected but also for the society which will be affected indirectly.

The second are creditors, who have a big interest in the health of the companies they give out loans to. This is because losses on unpaid loans can prove costly for banks. Deciding a better trade-off between risk and return can help to improve long-term profits by reducing the downside risk. There are also costs related to bankruptcy securities and pre-bankruptcy debt. The analysis by Branch (2002) shows that costs associated with debt accounts for 16% (for managing financially distressed companies) and 28% due to bankruptcy. To decide which companies to give out loans to and to what interest rate, credit ratings can be used. However, not all companies are rated, and for those companies, the rating may be done by the banks themselves. To assign credit ratings, creditors need accurate and reliable systems to decide which companies to provide loans to and to what interest rate.

In section 4.2 we modelled such systems by predicting financial distress in Norway. The machine learning models made a distress prediction for all 6611 companies in the test data during final evaluation in section 4.3 and gave a probability for distress for the different companies. We did, however, not give any explanation behind the predictions. It was merely a numeric probability which gave a prediction for *distress* or *healthy* depending on whether the companies had a probability higher or lower than the decided threshold. To give proper value to the predictions for decision-makers to use in practice, we believe predictions need proper explanations behind them. We want to provide the *why* behind the predictions. Without the *why*, predictions are merely binary answers to a complex problem. They do not say how the models came to their conclusions. Furthermore, prediction explanations will open for evaluation and proper understanding behind the underlying mechanisms for the predictions. This can help add to the analysis by the government and creditors for what drives companies to distress, however, not to be confused by causality in the real world. To explore and display the predictive process and how to interpret the models, we have chosen a specific company which will be discussed in the next section.

Furthermore, for our case study, companies can be referred to as data subjects as they are affected by automated decisions. From section 2.2.1 we discussed that data subjects have the right to get explanations of decisions made by algorithms. This regulation has implications in practice since it is not enough to base a decision on a validated machine learning model. All models used in decision-making should have the ability to provide explanations. We believe that our case is a typical situation where this comes into play. For these reasons, reliable and

accurate explanations are important. Using Shapley values as an explanation framework will satisfy these criterias.

5.2 Presentation of Case-Company

The case-company is chosen among companies with high probability for distress from our xgboost model. We chose a company among the distress predictions we found representative in the data. Companies with the highest probability of entering distress can be seen as extreme cases and thus not representative. We considered whether to choose a company randomly with high probability of distress by xgboost, or if the company in fact is entering distress (a true positive). We decided the latter, as it may provide a more valuable analysis. We also considered the different sectors in the data and decided on a company from the largest sector (construction). This choice was made because we needed a representative sample which is large enough when we dig deeper into sector analysis in section 6.2. For the analysis, only the financial ratios presented in section 4.1.1 will be considered. To compare the financial ratios in the data, we will be looking at the mean for each variable in terms of total mean and sector mean, presented in **Table 6** below.

Features	***Case-company***	***Mean***	***Sector mean***
X1: ROA	-0.24	0.05	0.04
X2: Current ratio	0.34	2.71	2.97
X3: Working capital / Total assets	-1.06	0.13	0.09
X4: Retained earnings / Total assets	-0.73	0.10	0.13
X6: Asset turnover	1.36	1.47	0.79
X9: Current assets / Total assets	0.56	0.58	0.41
X10: ROE	0.39	0.27	0.23
X13: Debt / Equity	-2.63	2.24	2.60
X17: Log of total assets	6.76	8.04	8.46

Table 6. Case-company

By comparing the ratios for the company to the mean in the data, we can see some differences. For example, we can see that *X1: ROA* and *X3: Working capital / Total assets* are low compared to the mean. From an economic standpoint, poor ratios could be a sign of distress. A negative debt ratio (*X13: Debt / Equity*) can also cause trouble for the company and is considered risky by other institutions. We can also see the company has lower efficiency and liquidity compared to other companies in the same sector. This is a sign of our case-company

not performing very well. It is difficult though to point out which variables are most important. It could be the negative *X1: ROA* or *X13: Debt / Equity* which is not sustainable in the long run, or it could possibly be other features or a combination of contributing factors. The feature *X6: Asset turnover* is close to the mean, but the sector mean is far lower. This is an interesting find, but we cannot say anything conclusive about factors that leads to distress from an economic standpoint.

Our case-company has been labeled as distressed and is therefore expected to enter this state within two years. For our case-company, there are some possible reasons to explain the financial situation directly. Financial distress is, however, often complex and not due to a single factor. It could therefore be a situation in which many factors are contributing and combined they result in financial distress. If a creditor or the government knew beforehand which companies would enter financial distress in the future, they would likely want to know which factors contributes to the situation. For the case-company itself, it can be well known why it has financial issues. Information from prediction models can possibly supplement or confirm the hypothesis of the company. In the next section we will continue our analysis by looking at the prediction models and attempt to explain the prediction for the case-company.

5.3 Interpretation of Predictions for Case-Company

The two models we presented in section 4.2 made a prediction for our case-company during evaluation in section 4.3. According to the threshold we decided in section 4.3.1, our chosen case-company has either been predicted to be healthy or distressed, depending on the probability output from the model. This will be the focus of the coming sections 5.3.1 and 5.3.2. Through this exploration, we hope to illustrate how prediction explanations can be given directly, if possible.

5.3.1 Decision Tree

Our first and simple model is a decision tree, which, according to reviewed theory in section 2.1.1, has the advantage of being interpretable. During final evaluation, the decision tree predicted our case-company with a probability for distress higher than the decided threshold. As discussed earlier in this chapter though, a binary answer is of little value to draw real insight from a prediction model. Assume a creditor for example wants a prediction explanation for

this company. Then it is important to understand why the company is classified as distressed. Why is it classified as distressed, and which variables made the model predict this outcome? This question is easy to answer with a decision tree. To provide a prediction explanation by the tree model, we can investigate the model designed during modelling in section 4.2.2. We produce the decision tree with the rpart package and include distress statistics. The decision tree is presented in **Figure 5**, with distress statistics included for each variable.

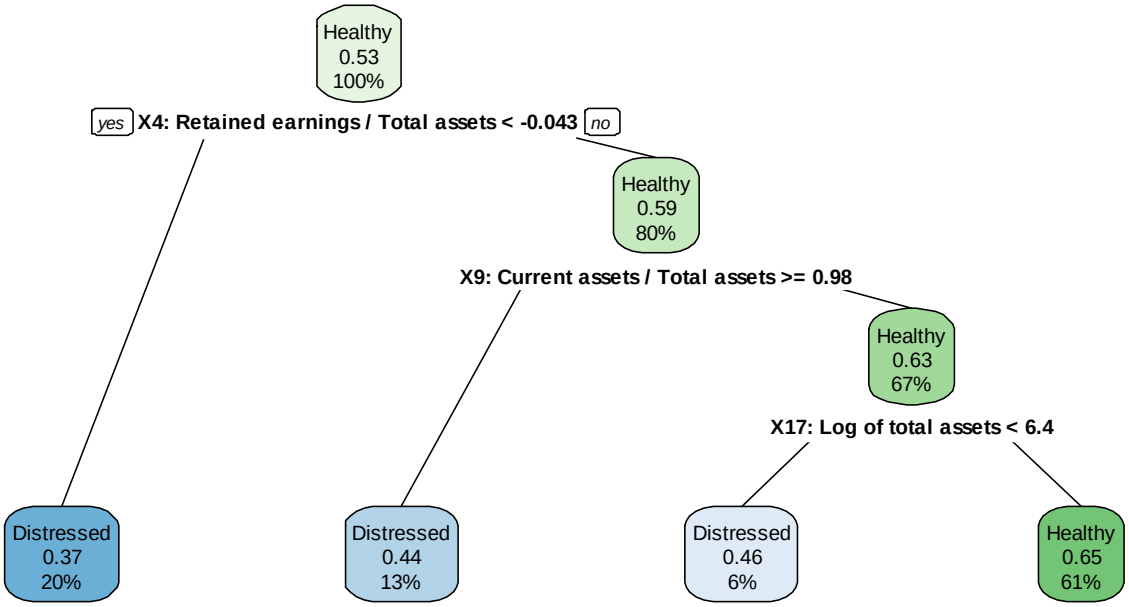

Figure 5. Binary classification tree

To decide why the company was predicted distressed, we can study the tree by investigating the branches and the different splits. From the tree, we can observe *X4: Retained earnings / Total assets* to be the most important explanatory variable with a threshold for where it will split the tree. If a company has a value for this feature lower than -0.043, then the company is predicted to be distressed, if not, then we continue to the next branch in the tree. In the next branch we follow the same intuition. If the ratio *X9: Current assets / Total assets* is greater than 0.98, then the company is predicted to be distressed, if not we will continue to the next node. In the third split we observe whether the variable *X17: Log of total assets* is less than 6.4. If the answer to this condition is yes, then the company is predicted to be distressed, else it is predicted to be healthy. The answer to every condition must be no for a company to be predicted as healthy. To explain why our case-company was predicted to be distressed, we compare the company ratios to the conditions. The prediction explanation is presented as a comparison between the model and the ratios, presented in **Table 7** below.

Variable	*Company ratio*	*Condition*	*Condition ratio*	*Answer*
X4: Retained earnings / Total assets	-0.73	$<$	-0.043	Yes
X9: Current assets / Total assets	0.56	$\geq$	0.98	No
X17: Log of total assets	6.76	$<$	6.4	No

Table 7. Prediction explanation for case-company by decision tree

By studying the table, we can see the case-company has ratios which give the answer yes to the first condition, and no to the other two conditions. Since the company has a ratio lower than the first condition, it is predicted to be distressed. **Table 7** in conjunction with **Figure 5** can be considered a prediction explanation for our case-company. The explanation provides the *why* for how the model came to the distress conclusion. In this way, a creditor or the government could study such a figure when there is a need for explanations behind predictions.

This explanation gives a very simple answer to a complex problem in the way that the feature *X4: Retained earnings / Total assets* alone is enough to decide a distress prediction. Furthermore, although decision trees are easy to interpret, they can oftentimes be quite inaccurate due to their high variance as discussed earlier. In the next section we will try to explain the same prediction by xgboost to explore if this can be done directly, such as for the decision tree.

5.3.2 Xgboost

The theory behind the complexity of xgboost is explained in section 2.1.1 and 4.2.1. To try to explain this very complex model, we have plotted the first tree during iterations (tree 0) using the *xgb.plot.tree* function in R (Chen et al., 2021). The tree is presented in **Figure 6**. By studying the tree, we observe the first split to be by the variable *X13: Debt / Equity*. The tree goes to the next node if our company is over or under 0.001 for this variable. For our case-company this is true as the variable value is higher than the condition. We consider the next node and since *X3: Working capital / Total assets* $= -1.06 < -0.26$ we end up in the leaf 0-3. This means we have a marginal value of approximate -0.01 (the difference between these two values), which is the contribution to the prediction. The leaf shows the cover which describes the second order of training data classified to the leaf. Value is the margin of the

leaf's contribution to the prediction. In the nodes, gain gives the idea of the importance of the node in the model.

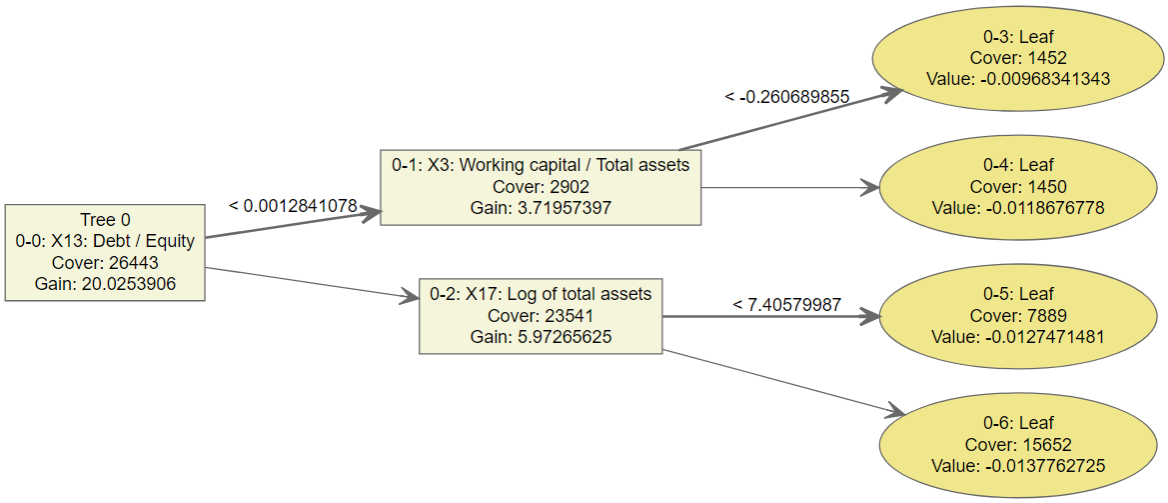

Figure 6. Tree 0 in xgboost model

It is difficult to extract a single answer from this model even though it is illustrated. In **Figure 6** we only study tree 0, but as it is an ensemble method there could possibly be many more similar trees. Number of trees will be equal to the number of iterations (200 in our case). This means 200 trees need to be evaluated to understand the outcome from a prediction. Each of these plots gives a hint of how the features contribute to the prediction. It is possible to extract an explanation from these plots, but it is time-consuming and very technical. Especially non-experts will struggle to interpret the model. Due to the complexity of the xgboost, and our inability to extract intuitive answers directly, there is a need for an explanation method. That is the focus of the next chapter, where we will investigate Shapley values to explain predictions on financial distress.

6 Shapley Value Prediction Explanation

In this chapter we will use the package *shapr* (Sellereite, Jullum, Redelmeier et al., 2021) in the programming language R to answer our first research question regarding intuitive explanations. The package implements the extension of Kernel SHAP which considers dependency between features, described in section 3.2.3. Section 6.1 will continue the investigation of the case-company which entered financial distress. We will calculate Shapley values for the prediction of this company to try to explain the prediction. In section 6.2 we will perform sector analysis by comparing Shapley values of our case-company to that of Shapley values for high and low probabilities in the same sector. Section 6.3 investigates feature dependence and feature importance in a global setting for the construction sector. This will help us answer our research questions about unfairness in the model and in explaining how a black-box model considers features.

6.1 Explaining an Individual Prediction by Xgboost

In chapter 5 we concluded that explaining predictions from xgboost is not possible to do directly as it is for a decision tree. However, as xgboost has higher accuracy compared to a simple tree model, there are benefits of using such a complex model. We believe that by investigating the prediction for a selected company which entered financial distress (and had a high probability of entering distress), there is potential for interesting interpretations of the black-box model xgboost.

To calculate Shapley values in R, the following procedures were taken. First, an explainer object was produced by running the function *shapr* from the shapr package, using the feature values from the training data and the model produced by xgboost in section 4.2.1. Second, the global average prediction was calculated as the mean of the distress variable in the training data, which equaled to $y = 0.062$. The global average is, as discussed earlier, the prediction without any features and is considered the undistributed gain. Lastly, Shapley values were calculated using the *explain* function in R with the abovementioned inputs, and by setting the hyperparameter *approach* to *empirical*. The reasoning behind this approach selection was discussed in part 4.1.2 during evaluation of distributions. Shapley values for the selected case-company including the feature values are presented in **Figure 7**.

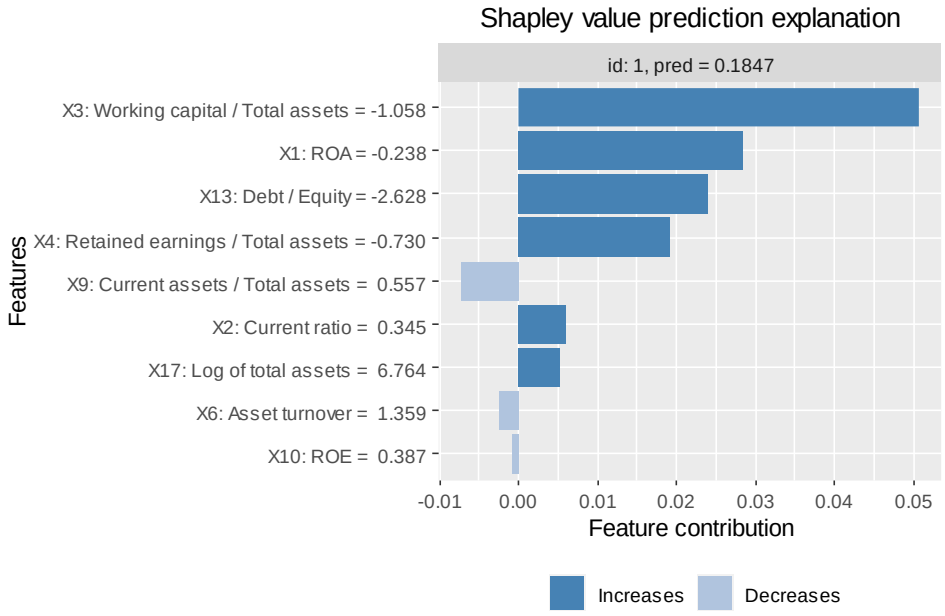

Figure 7. Prediction explanation for the case-company

The left-hand side of the graph presents the different features with the feature values (the test data input) next to them. We will refer to feature values as the values in the test data, while Shapley values are the explanation values on the x-axis. The average prediction (the undistributed gain) gives all companies a 6.2% chance of entering distress before the features impacts the likelihood. All features with negative Shapley values (light blue bars in graph) will contribute to labeling the company healthier by decreasing the probability for distress. Positive Shapley values (dark blue) will increase this likelihood. The combined prediction value for this company is 0.1847 (the probability of distress), presented in the graph header. With our distress threshold set at 0.06, this company has a high probability for distress according to our model. The Shapley values explain the difference between the undistributed gain of the prediction and the prediction value. By taking this difference, we can see the features combined result in a $0.1847 - 0.062 = 12.27\%$ probability increase of distress. We can also see how Shapley values properties come into play. The features are evenly distributed, and low impact features receive a low Shapley value. Suppose for example *X1: ROA* was left out of the model and all other features contributed to the prediction. Adding this feature in random order will give a different prediction value. The Shapley value for *X1: ROA* is the average change ROA will have on the prediction value when it is added to the feature mix.

Interpretation of Shapley Values for Case-company-Company

By investigating the Shapley values, we observe that most of the values are positive (and hence that most of the values increase the probability of distress). Since we have an undistributed gain over the threshold, we need features on average to decrease the probability to classify a company as healthy. We observe the feature *X3: Working capital / Total assets* as the feature which increases the probability of distress the most. It can therefore be considered the most important distress feature in the model for this company. According to Shapley values, this feature will increase the prediction by approximately 5%. The variable *X9: Current assets / Total assets* is among features which reduce distress probability. This feature affect the probability to a small degree. Each of the four most influential features can individually shift the prediction over the distress threshold. Combined, the features result in a high probability of distress. In our review of the feature values, we suspected that there were several features that could contribute to labeling the company distressed. The variables *X1: ROA* and *X13: Debt / Equity* were suspected to influence distress. Now we know we can confirm this suspicion and that the variables *X3: Working capital / Total assets* and *X4: Retained earnings / Total assets* add to the explanation.

In chapter 7 we will reflect on Shapley values and their implications in practice. Remember that these features are dependent on each other, so increasing sales will affect several features. All features should therefore be considered in context of each other since they are dependent. It will be up to the individual subject to find concrete actions to improve financial health. Shapley values present the most important features to prioritize according to the model. It is therefore up to decision-makers to find out how much weight they assign to prediction models. This case shows the combination of the different feature values that leads to a high predicted distress probability. To investigate whether this is an unusual case or if it is a general trend, we will in the next section compare our explanation to individual prediction explanations made on other sectors. Using Shapley values, we would also like to investigate whether there are any unexpected outcomes in the xgboost model or not. This will be the focus of the coming sections where we will explore prediction explanations for companies in different sectors and sizes. We believe exploring the different patterns of the explanations can help us open the black box of xgboost to understand how it works.

Comparisons of Shapley Value Prediction Explanations

To decide whether the Shapley values of our case-company are representative for distressed companies, we extracted four additional companies with high probability of distress from four different sectors. The prediction explanations are presented in **Figure 8**. The different sectors are manufacturing (in the top left), telecom/IT/Tech (top right), Wholesale/Retail (bottom left) and finance (bottom right). These sectors are together with the construction sector the largest sectors in the data.

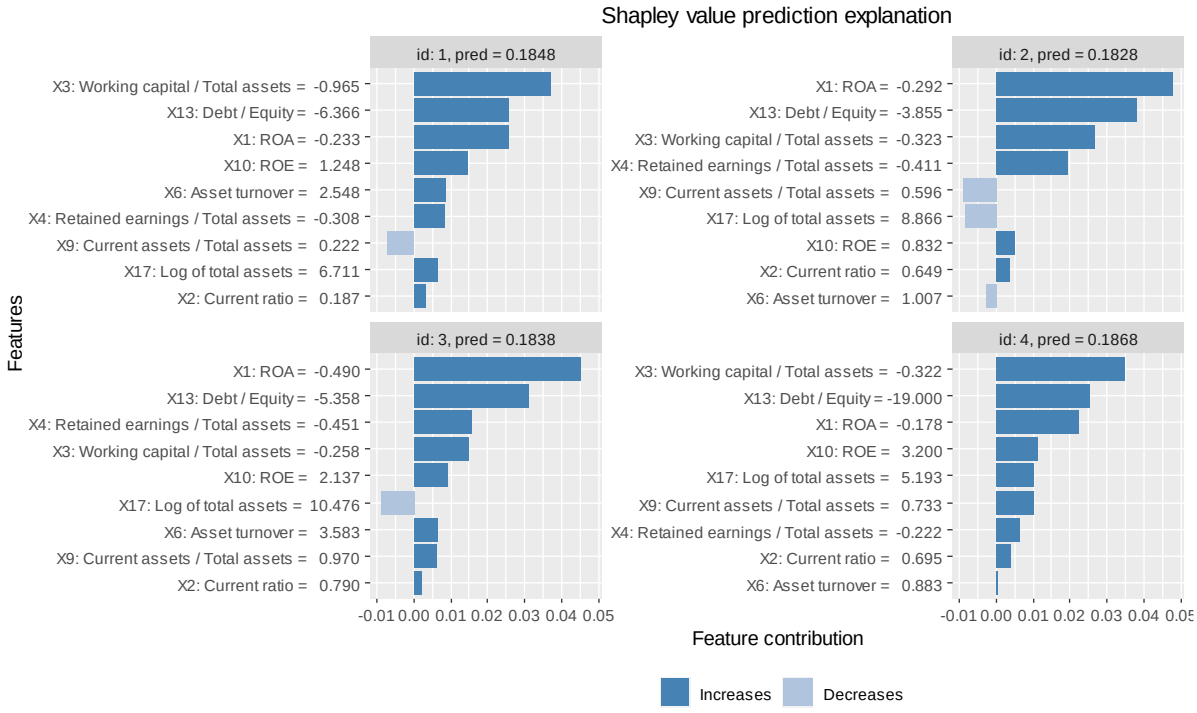

Figure 8. Individual Shapley prediction explanations for the four sectors: Manufacturing (top left), Telecom/IT/Tech (top right), Wholesale/Retail (bottom left), Finance (bottom right)

From the figure we observe almost all features contribute towards a higher distress probability. The features that increase predicted distress probability the most are the variables *X1: ROA*, *X13: Debt / Equity* and *X3: Working capital / Total assets*. Therefore, the most important features do not vary between the selected companies, and we consider the prediction explanations to be similar. For the least contributing Shapley values, however, we observe some variation. By comparing the variable *X10: ROE* between **Figure 7** from section 6.1 and **Figure 8** for example, we can see that this feature is the least contributing feature for our case-company, while it varies how much of an influence it has for the predicted distress probability for other companies.

6.2 Shapley Value Sector Analysis

Our case-company is in the construction sector which consists of 12.202 companies in the data, or 36.92% of the sample. We believe investigating predictions and prediction explanations for this sector specifically can possibly help us find good comparisons to our explanation of why the case-company is predicted to enter financial distress. Since computing Shapley values is computationally expensive, we needed to restrict ourselves to a smaller data set. The construction sector was a suitable choice because it had enough predictions to make a good analysis while at the same time was just small enough to compute Shapley values for all predictions. These companies were also comparable to each other and our case-company. We do not know whether the construction sector is similar to other sectors without further analysis. Of course, there could be some different specific feature values across sectors. We believe, however, that there are small differences between the sectors since this feature is not included during training of the model. The model will therefore treat each company the same regardless of which sector it comes from.

6.2.1 Individual Comparison Analysis for the Construction Sector

We will study prediction explanations for four additional companies. This time we will study explanations for companies with a low probability of distress. The companies are in the same sector as the case-company (construction) with around the same size (*X17: Log of total assets*). Therefore, they should have a similar base of operations as our case-company. We believe by studying differences between the distress explanation of the case-company and healthy explanations, we can discover which features are more important than others. The four prediction explanations are presented in **Figure 9**.

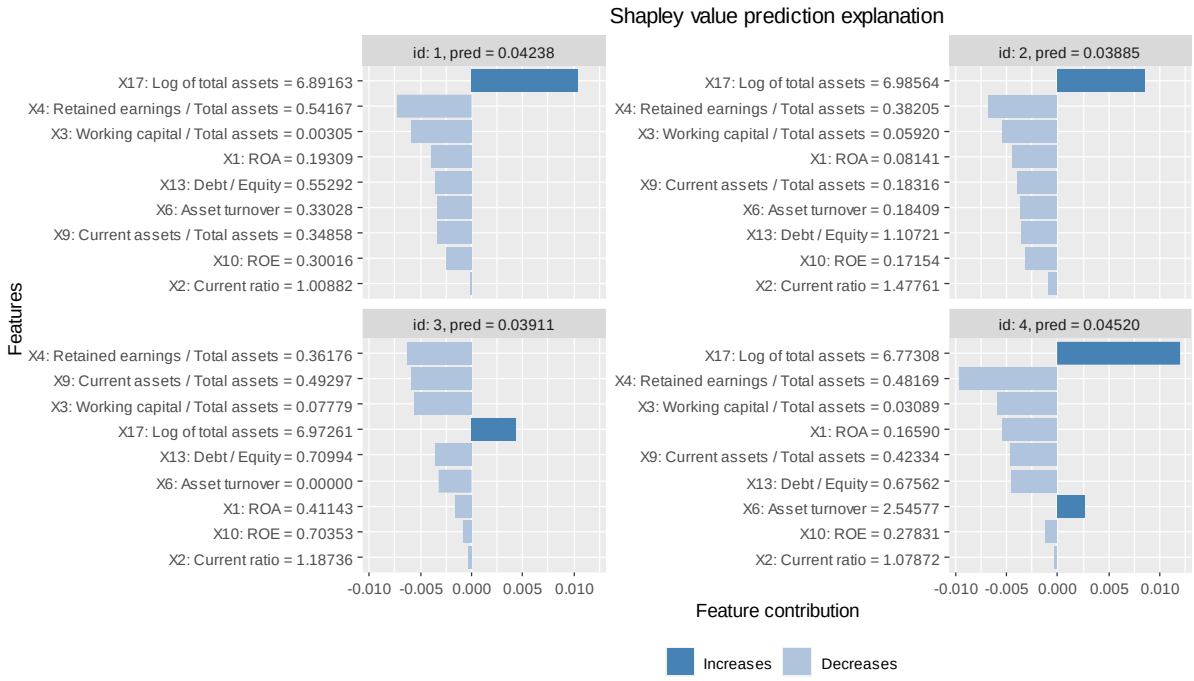

Figure 9. Prediction explanations for low probabilities of distress (construction sector)

By studying the plot, we observe all companies to have a low probability of distress according to our model. The probability is under the decided threshold of 0.06, and we consider the companies to be predicted healthy. If a creditor should use our model in a risk assessment, they would view these companies as low-risk companies. All companies have the variable *X4: Retained earnings / Total assets* as their most contributing feature for decreasing distress probability. The four plots in the figure look similar, with some minor differences overall. We observe all features to be contributing to a healthy prediction, with the common exception of the variable *X17: Log of total assets*, which is also the most important feature for three out of four companies. This gives us an indication that for a company to have this size is a contributor of increasing distress probability. It also raises the question of which of the feature values leads to which Shapley values. We will investigate this in further detail in section 6.3.

We can also observe the variable *X6: Asset turnover* to be an exception of probability reduction in the bottom right of the figure as it increases the probability of distress for this company. The exception also has the highest feature value of 2.54, which makes us believe a high asset turnover may increase the probability of distress in the model. However, it is important to keep in mind that these values are dependent features, and that they should be considered in relation to each other. Interactions and dependencies will be further explored in section 6.3.

6.2.2 Sector Analysis for Top and Bottom firms in Construction Sector

Next, we would like to look at Shapley values for a greater group of predictions. This will help us decide how much each feature is affecting prediction outcomes. The focus of this section will be on distress predictions and healthy predictions separately, while global feature importance will be considered at the end of the chapter. We believe this investigation will really help us highlight which features contribute to a healthy prediction or a distress prediction. By exploring a larger sample, we will also be able to pick up potential outliers from **Figure 8** and **Figure 9**. To do this, we first compute Shapley values for the companies with the lowest probabilities of distress by taking the average of the Shapley values for the bottom 30 predictions. To highlight the direction of the Shapley values we take actual values and not absolutes in this section. Mean of the low probabilities for distress are presented in **Figure 10**.

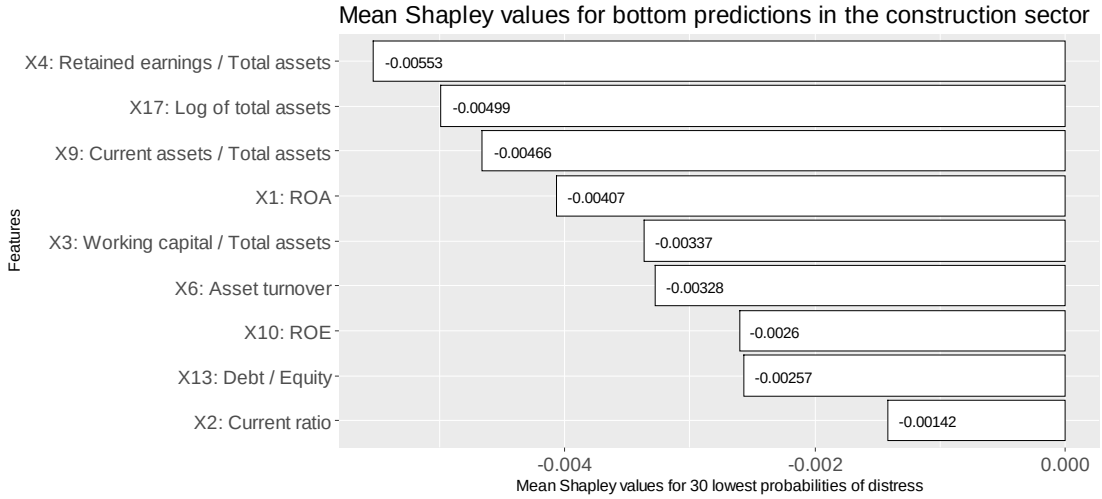

Figure 10. Mean Shapley value prediction explanation for companies with low probability of distress in the construction sector

Observing the figure, we see that the variable *X4: Retained earnings / Total assets* is contributing the most to a low probability of distress on average over the 30 predictions, while the variable *X17: Log of total assets* is contributing second most. The plot shows that the variable *X4: Retained earnings / Total assets* is the most important for healthy predictions on average. This variable is also negative and shows that when we considered companies with similar size in the last section, we may have found some unusual cases. This is because the variable is now decreasing the probability of distress, not increasing it. In section 6.2.1, we also saw the variable *X3: Working capital / Total assets* to be the third most influential feature. This is not the case from the figure below, where the feature *X9: Current assets / Total assets*

this time is decreasing distress probabilities more than *X3: Working capital / Total assets*. Next, we do the same calculation for 30 of the predictions with the highest probability of distress in the construction sector and compute the mean of the Shapley values. Results are presented in **Figure 11** below.

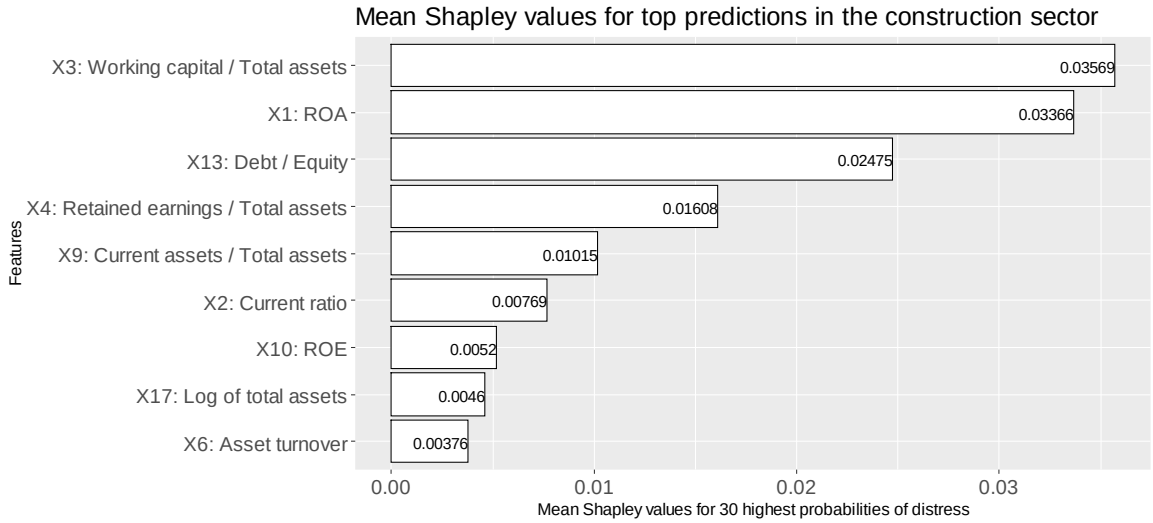

Figure 11. Mean Shapley value prediction explanation for companies with high probability of distress in the construction sector

When studying the figure, we can this time determine the variable *X3: Working capital / Total assets* to be the most influential feature in increasing distress probability in the xgboost model, while the variable *X1: ROA* is the second most important variable. When we interpreted predictions for distressed companies previously in this chapter, we also found this feature to be important. Now, we can see that it is a general trend for distress predictions. By comparing **Figure 10** and **Figure 11** we can spot similarities. However, there are fewer features to be influential for companies with high probability of distress. The graphs are thus more skewed for these companies. This means that there are bigger differences between the top and bottom influential features. Global feature importance for the construction sector will be of focus at the end of the chapter.

6.3 Shapley Value Dependency Investigation

In this section we will explore how feature values relate to Shapley values in the construction sector. Generally, using this analysis we may be able to uncover which feature values leads to which Shapley values. By performing this analysis, we can also say more about the requirements of feature values and what they need to be to obtain high Shapley values. Keep

in mind that all features must be seen in relation to each other. Since we now are exploring an overview of the prediction explanation for the case, we can also possibly disclose illogic economic patterns in the broader setting in this section. The Shapley value dependency interactions are presented in **Figure 12**. Shapley values are on the y-axes and feature values are on the x-axes. Looking at the plots, there are two things we can initially see. First, there are clear patterns for many of the plots. This is important because clear patterns will help us understand how the xgboost model considers different feature values and may help us discover unexpected model behaviors. If the plots had just shown noise and no distinguished patterns, it would be difficult to interpret the plots. Second, there are some outcomes that are unexpected, in the way the patterns interfere with economic reasoning. If we recall what certain feature values mean, *X1: ROA*, *X4: Retained earnings / Total assets*, *X2: Current ratio* and *X3: Working capital / Total assets* are examples of variables which should be high to indicate healthiness. Our xgboost model seems to disagree in some cases. Interesting finds will be pointed out and further discussed in chapter 7.

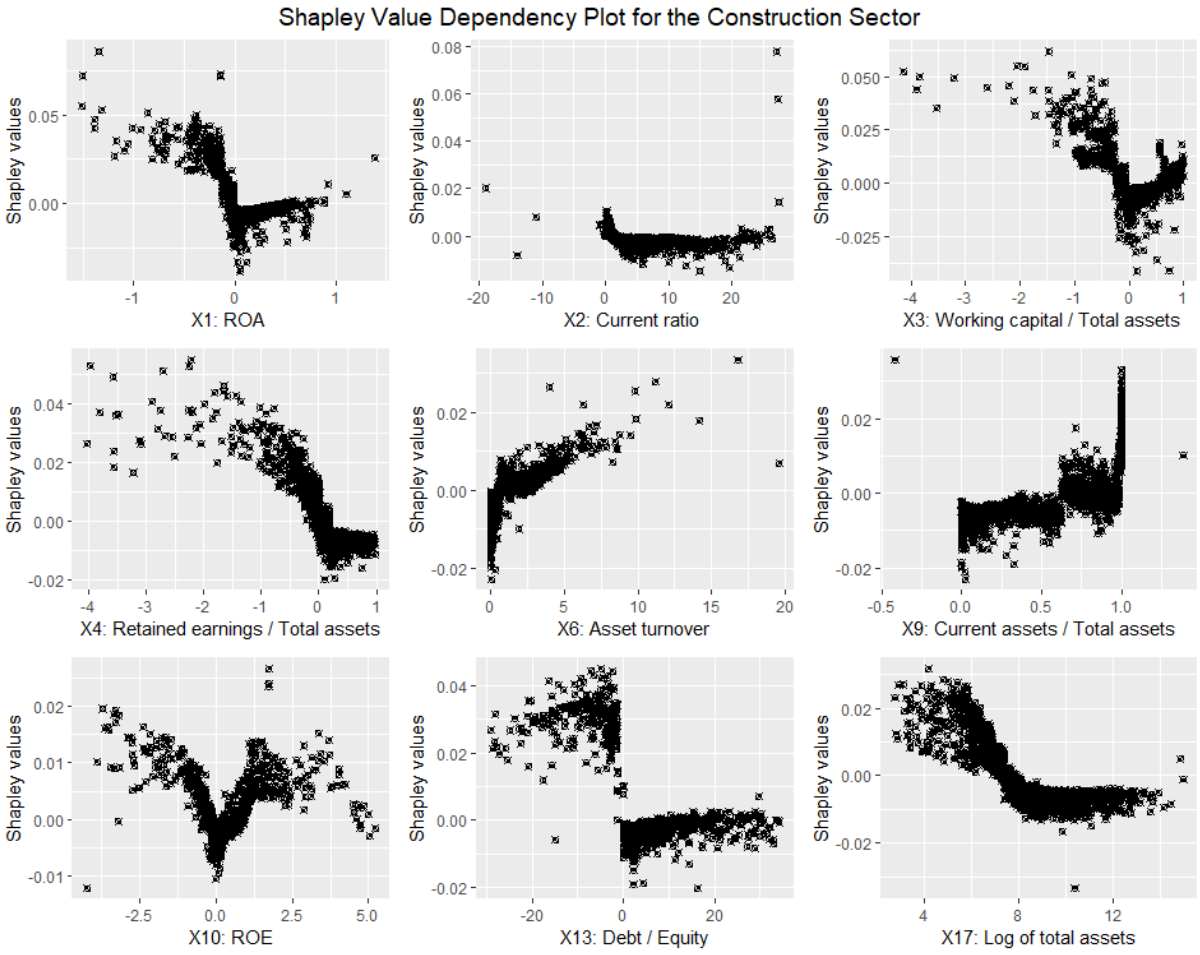

Figure 12. Global Shapley value dependency plot for the construction sector

Observing **Figure 12**, we can start by evaluating the first variable *X1: ROA* (in the top left of the figure). We can see there is a clear pattern in the way a negative ROA increases the probability of distress. Companies with a ROA equal or larger than zero will receive a negative Shapley value and will thus decrease the probability of distress. There is a positive trend when ROA increases which is unexpected and will be further discussed in chapter 7. *X2: Current ratio* does not give strong indications of a high or low prediction value in the figure. Going back to earlier sections in this chapter, we can recall that this feature was quite unimportant for previous prediction explanations. No large swing in Shapley values gives further indication that this feature is of less importance. For the next features, *X3: Working capital / Total assets* and *X4: Retained earnings / Total assets*, we can see a similar trend as with ROA. Negative feature values increase the probability of distress. Also, like ROA, there is a positive trend for feature values larger than zero. Feature values close to zero decrease the distress probability the most.

Next, *X6: Asset turnover* is an interesting feature. Looking at the figure, we observe that only feature values of zero or close to zero will decrease the distress probability. If feature values are higher, we can see a clear trend of increasing the probability. In economic terms, we consider a high asset turnover to be a positive factor for a company. Therefore, this is unexpected and will be further discussed in the next chapter when we consider economic intuition. The next feature, *X9: Current assets / Total assets*, gives a clear cutoff at approximately 0.6. Feature values under 0.6 will increase the probability of distress while values above will either increase or decrease it. Another insight to draw from this plot is that there are many observations with the value of 1.0 (a situation which occurs when current assets = total assets), but where the Shapley value varies between companies. This is perhaps strange since it effectively means the same for all companies. Therefore, it is a good time to mention again that each feature value must be considered in relation to the other values in the feature mix. This is the reason for why a feature value of 1.0 gives a stronger probability increase of distress for some companies than for others in the model.

Next, we consider the variable *X10: ROE* in **Figure 12**, for which we can observe an interesting *"v"* pattern. The plot shows increasing Shapley values on either side of the feature value of 0. For feature values increasing or decreasing from 0, this means the distress probability is also increasing. There are also a few outliers in the plot which make the interpretation less clear. However, high ROE values are increasing the probability of distress

most of the time, which is unexpected. This will be discussed in conjunction with *X1: ROA* as they obtain similar results. Furthermore, we consider the variable *X13: Debt / Equity*. This was one of the most influential features from our analysis in earlier sections. From the plot we can observe the following clear distinction: for most companies, negative feature values will increase the distress probability while positive values decreases it. A ratio close to 0 provides the largest probability decrease which coincides with economic theory and is thus expected. The effect of a slightly negative feature value compared to a value of zero is furthermore quite substantial. Companies having a negative *X13: Debt / Equity* ratio will, according to the model, have an increase in distress probability between 0.5% and 4%.

The last feature to consider is the variable *X17: Log of total assets*. Findings from Zhang and Ye (2019) showed that smaller companies have a higher probability of entering financial distress. Our model seems to agree on this, where we can observe from the bottom right plot in **Figure 12** that companies with a size of approximately 8 or lower will receive a positive Shapley value and thus have an increase in distress probability. The probability increases proportional to a smaller company size, and thus the smallest companies will be affected the most in terms of a probability increase of distress. Companies with size greater than 8 on the other hand will decrease the probability, with a flat trend above this threshold. As this variable is a measure of size, it could require human interpretation. This will also be discussed in the next chapter.

6.3.1 Feature Inspection Comparing Figures for ROA

In section 6.2.2 we concluded the variable *X1: ROA* to be one of the most important variables for both distress predictions and healthy predictions. Elaborating further on the analysis in **Figure 12**, we have extracted this feature and highlighted the top and bottom 30 predictions, presented in **Figure 13**. We did this to see if there are differences between the top and bottom predictions. These are the same companies as introduced in section 6.2.2, with bottom predictions (lowest probabilities of distress) illustrated with green points and top predictions (highest probabilities of distress) with red.

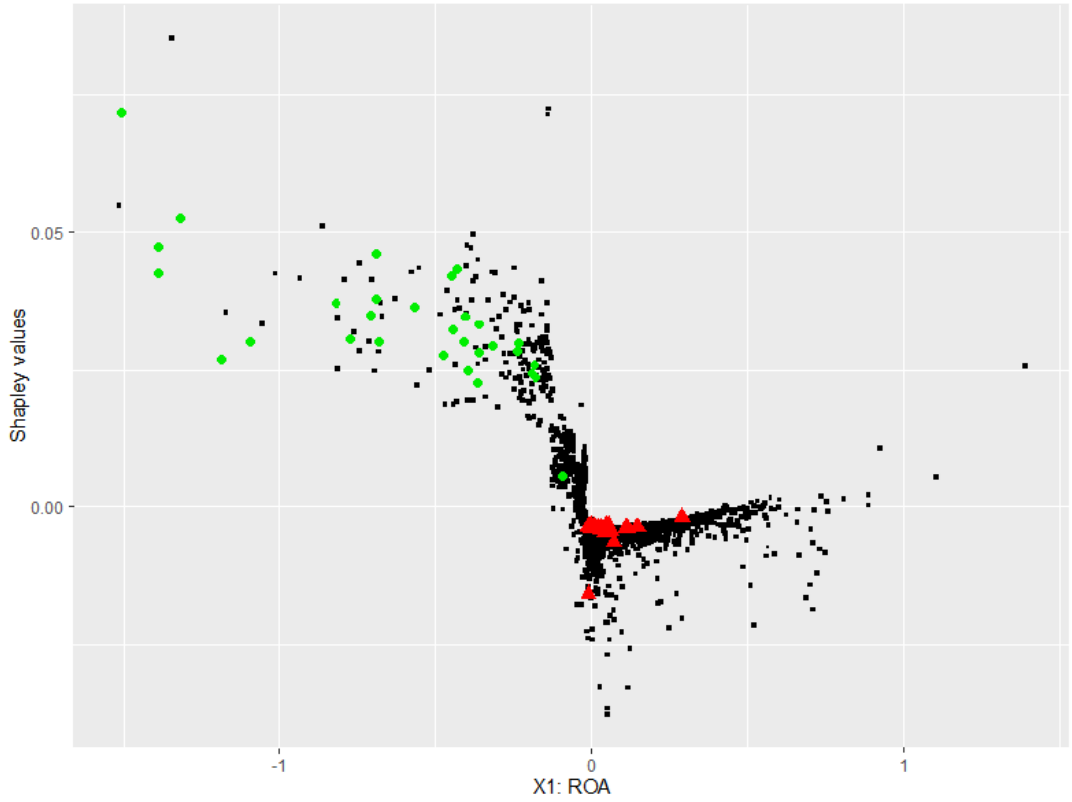

Figure 13. Feature dependence plot with interaction effects for the construction sector of the lowest 30 distress probabilities (red) and highest distress probabilities (green)

From the figure, we can observe companies with the highest probability of entering distress to have a negative ROA. If ROA is negative, almost all cases give a positive Shapley value. The bottom predictions (companies with lowest probability of distress) have ROA values going from about 0 to 0.4, giving Shapley values under 0 which decreases the probability of distress in the model. The red points are fairly collected and separated from the green points. The largest negative Shapley values can be observed around the feature value of 0. The feature inspection shows that negative ROA values impacts predicted distress probability by a decisive amount. If we were to consider the upper green points, for example, they have a probability increase of 5% which is quite substantial for the model. This means the ROA ratios for these companies by themselves make it almost impossible for the company to obtain a healthy prediction due to the distribution of negative Shapley values seen from **Figure 10**. Consequently, this plot illustrates the point of importance of companies to improve their ROA ratio to be considered "healthy" given the distress threshold of 6%.

6.3.2 Global feature importance

Lastly, we wanted to inspect global feature importance for the construction sector. This will help us decide which features the Shapley value framework considers the most important in the xgboost model. To compute importance, we calculated average absolute Shapley values for all features in the construction sector and took the mean of the values for each feature. We calculated absolute values this time as we do not want to see how much the feature values affect the prediction outcome on average, but rather how influential the features are on the model in a larger setting. Global feature importance is plotted in **Figure 14**.

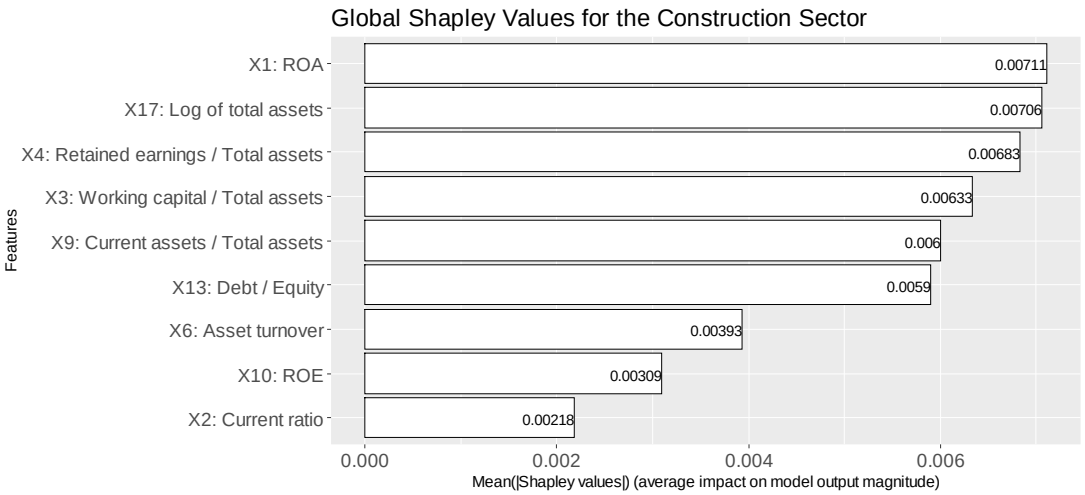

Figure 14. Global feature importance for the construction sector

From the plot we can see that many features are influencing the average prediction to a similar extent. When comparing the global Shapley values, the variable *X1: ROA* is the most contributing feature, close to the size of a company (*X17: Log of total assets*) in terms of effect on distress probability. This means these features are most important during prediction. Since we use absolute values, this relates to both an increase and decrease in the average prediction value. Many important features reflect well on the complexity of our prediction case. *X2: Current ratio* on the other hand is the least contributing feature with the smallest average Shapley value. This was suspected in **Figure 12**, due to the lack of variety in the plot. **Figure 14** gives an indication on how an unseen company will be evaluated, but as we know from previous assessments, there are individual differences.

7 Reflections

This chapter will reflect on the results, the method, and the explanations. First, in section 7.1, we will evaluate on the prediction explanations made in chapter 6. In section 7.2 we will evaluate on the method, such as its potential and relevant challenges. Finally, in section 7.3, we will discuss the implications the method can have for decision-makers in practice.

As indicated in the introduction, prediction explanations can detect possible bias in a model. In chapter 2, we defined bias as an error that occurs when the model is too simple for the problem. As this is a wide term and is interpreted differently, we will explain what we mean when we use the term bias in this chapter. When a model is applied to a complex problem, you can achieve unexpected outcomes. We studied feature dependencies in **Figure 12** and discovered results which were unexpected. Results that are unexpected relative to standard economic theory are therefore considered biased in our model. Xgboost is a complex model which is usually low on bias, but as we know from our literature chapter, bias can arise from training data or model tuning. In this situation we suspect that our model has been tuned to provoke some bias or due to the restricted number of features. We also believe that there is a biased distribution in our data, as *X17: Log of total assets* possibly discriminates smaller companies. These biases will be discussed in section 7.1 and 7.3 since they have implications for decision-makers in practice.

7.1 Evaluation of Results

In the previous chapter we provided explanations for both individual predictions and groups of predictions. In terms of economic intuition, some of these prediction explanations did not make economic sense. In section 4.1.1 we discussed how the different economic ratios relate to economic healthiness. From **Figure 12** in section 6.3 we did see many patterns that were in line or partly in line with discussed economic theory. For the variable *X1: ROA* for example, a negative ROA negatively affected the predicted probabilities which made sense in economic terms. The positive Shapley value trend for ROA values over zero (increasing distress probability with increasing ROA values over 0) is, on the other hand, not in line with economic theory. A high ROA should be a sign of healthiness. Other features did also not contribute as expected, like the variable *X6: Asset turnover*. A high asset turnover should pull in the direction of decreasing distress probability, but our model suggests otherwise. This may be

the result of the complex model picking up a trend in the data or a bias. A benefit with explaining the model from a global perspective is that it enables the opportunity for discovering possible bias. If detected, the model can be re-trained to exclude the bias and thus improve performance and fairness.

There are several possible explanations for why such biases occur. A possible explanation could be that 9 variables are not sufficient to predict a complex problem such as financial distress. Even though the model uses all variables, that may not be sufficient to explain enough of the process, potentially leading to some unexpected outcomes. The number of features had to be restricted in our model due to the computational problem with Shapley values. Consequently, we may have a too simple model for a complex case. Financial distress is in most cases complex and not due to one single factor. Another possible explanation is that the irreducible error is large enough to influence the predictions, which ultimately affects the explanations. Noise in the data could hinder the detection of patterns for our model resulting in model bias due to a simpler model than the problem.

To conclude whether our entire model is biased, we need to properly view the whole model and not just the construction sector. We expect there could be some differences in feature values between sectors. Comparing the construction sector to the mean of the whole data showed that most features were close to the mean, but *X6: Asset turnover* was much lower in this sector. Therefore, a possible explanation could be that a high asset turnover in fact is a characteristic of poor performing companies in the construction sector. Without further analysis across sectors, this remains unsolved. We can suspect that the model contains bias but cannot conclude whether this is a bias in our model.

Financial distress is often complex, and it may often be the interaction of many variables that causes unexpected outcomes. It is not as simple as saying that one thing needs to change. While it can be tempting to follow the explanation by the decision tree, it probably may give a misinterpretation as it is not telling the whole story. This is a good example to illustrate decision-makers should not follow explanations blindly. Even though these models are complex, they do not consider the context of the features, as they only search for advanced patterns. This means that when applied to other problems, some explanations could be illogic. Therefore, it is important to have a correct understanding of the explanations.

Calculating Shapley values for a larger sample enables us to detect bias and therefore disclose possible discrimination in the model. The companies from the construction sector presented in **Figure 9** also gave an interesting result. These were companies with low probability of distress. Shapley values decreased the probability of distress except for the feature *X17: Log of total assets,* which increased the probability. From the plot in the bottom right in **Figure 12**, we can see the reason for this, as a small feature value gives a positive Shapley value. For a feature value of 8 and lower, Shapley values are positive, and hence increases the distress probability. The intuitive explanation for these companies is that they are well-performing, but their *X17: Log of total assets* is the reason for them not to be in the top healthy group of predictions. This feature is a measure of size and thus not a feature easily changed. We know that small companies are more likely to enter financial distress than others. We also found this to be true from our model. It can however be considered unreasonable not to place these companies in the top healthy group. They all have other feature values to suggest that they should be in this group. Since they are well-performing, these companies may even increase in size over time (growth). If they can hold other indicators at the same level, they should then be predicted in the top healthy group. This is an example of how the model could discriminate. Machine learning model does not set features into context as humans do. The simple solution would be to exclude this feature from the model. This would likely decrease the predictive accuracy and is therefore undesired. Explanation methods like Shapley values allow for the model to be unchanged and leave the decision-maker to decide if this feature discriminates smaller companies or if it is a real concern. We consider this an important advantage of the Shapley value framework in the way that it picks up patterns which cannot be discovered from observing the model.

7.2 Evaluation of Method

Shapley values is a theoretical strong framework to provide both local and global explanations. It can be applied to all types of prediction models and problems. Shapley values also have properties which make the method desirable over other methods. Compared to our explanation of a decision tree, it provides a more complete explanation. There are, however, some challenges with Shapley values. The first challenge lies in the complexity of computing the explanations. Kernel SHAP reduces this challenge by approximating Shapley values, but with a high number of features, the computation time increases substantially. There are ways of dealing with this issue, such as reducing features, but since this would affect prediction

accuracy, it is undesirable. Another solution would be to group features by dependency or other attributes. For our case, this can be done by grouping features based on their classification or by grouping in terms of dependency (correlation) of variables. A classification grouping could be to group by the type of ratio, efficiency, liquidity, leverage, and profitability. Shapley values had then explained a company's prediction based on these groupings. The downside of grouping, however, is that it makes prediction explanations less specific. Grouping of variables is therefore a trade-off between the value of obtaining specific explanations and to the degree of computational complexity. We solved this issue restricting number of variables and by selecting a sector with a size small enough to be able to compute Shapley values for all predictions in that sector. By doing this, we obtained a sector specific analysis which enabled us to compare companies in the same sector. The sector is close to the mean of the whole data. We therefore believe it is a rather small difference between this analysis and the whole model. Smaller sectors with large deviation in relation to the mean could, however, obtain other explanations. Without further analysis though, we do not know how this analysis generalizes to other sectors.

Another challenge with Shapley values is that it is hard to know the exact dependence structure. Dependence is estimated and chosen based on observations on the data. We used the most applicable method for our data, the empirical conditional distribution. A method that is proven to have weaknesses but to be more accurate than traditional dependence estimates (Aas, Nagler, Jullum et al., 2021). The advantage with this method is that it estimates each feature and does not assume all feature distributions to be the same. Estimates could end up being different from the real dependency and thus give inaccurate estimates. This will have implications on the Shapley values computed. To cope with the issue, better estimates of dependency are necessary. We use the methods which are proven by Aas, Jullum, et al. (2021) to be more accurate than other Shapley-based methods. To the best of our knowledge, Shapley values are the preferred explanation method and the extension we use performs best on real-world data. These qualities were important when choosing an explanation framework, taking regulation and ethical aspects into consideration as well.

Interpretability in machine learning was defined in chapter 2. For models to be interpretable, non-experts should receive an explanation based on visualization and natural language (Murdoch et al., 2019). As our result shows, Shapley values provide intuitive explanations for individual predictions. The plots from individual predictions can easily be interpreted by a

non-expert although a short introduction to the plots and how the framework functions is useful. Knowledge of how the non-distributed gain works and how each feature either increases or decreases the probability from this baseline must be known. The data subject receiving the explanation, is of course familiar with the feature values for its company. Through explanations, these feature values will be set into context of how they contribute to the company according to the model. To discuss more concretely for the case at hand, we will in the next section discuss different implications Shapley values can have for decision-makers in practice.

7.3 Implications for Decision-Making in Practice

From chapter 6 we explored how individual predictions from a black-box model can be explained by using Shapley values. The chosen case considers predictions on financial distress, which we found relevant to explain predictions from. In our opinion, the predictions and explanations are relevant mostly to banks and financial institutions when assessing credit ratings for companies. Institutions are mostly focused on reducing the probability of default. The importance of model interpretations was discussed in section 2.2 with the focus of ethics guidelines for a trustworthy AI, regulations and disclosure of bias. Thus, there are several potential implications the method can have generally for decision-makers in practice, and specifically for the case studied.

The first implication is that Shapley values as an explanation framework enables decision-makers to continue using complex algorithms in the future. As data subjects have an increasing right to demand explanations behind decisions made by automatic tools, decision-makers may be required to explain decisions. For the case discussed, this is especially true in the case of loan applications. A company may demand an explanation behind the reason for why an institution rejects a loan application. To provide proper explanations, financial institutions naturally need to understand their model. To understand a prediction model, we considered two potential solutions. You can either revert to a simpler model (such as a decision tree presented earlier) or stick with a complex model and explain it using an explanation framework. Simpler models are undesirable when there is a substantial loss in accuracy. Accuracy is important as it provides practical value of a model. Thus, a decision-makers may be left with accurate but complex models which they do not completely understand. To cope with the problem of explanation, we have shown how Shapley values can be used to explain

predictions. We argue the explanations provide simple and intuitive answers for the predictions. Therefore, the framework removes the downside of complex and uninterpretable models.

The second implication Shapley values have for decision-makers in practice is that they enable modelers to disclose possible bias in a model. As we saw for xgboost, the model was difficult to understand. It was merely a probability output for distress (the *what*), and it was hard to say which features influenced the predicted distress probabilities to which degree (the *why*). Estimating Shapley values for a larger group of individual predictions from the construction sector enabled us to understand which feature values resulted in which Shapley values. From the explanation graphs we extracted some interesting insights in the way features in relation to Shapley values provide varying economic sense. Some graphs were perfectly reasonable, like how negative *X1: ROA* values increases predicted distress probability. It made less sense, however, how increasing *X10: ROE* values in fact were increasing distress probabilities. Thus, we argue Shapley values enable humans to consider the results before deciding.

The variable *X17: Log of total assets* was also discriminating against small companies. This variable was found from section 6.2.1 in **Figure 9** to be the only variable which was increasing distress probability. The figure showed a special situation because in economic terms the selected companies would, all else considered, be perfectly healthy. Their only possibility to be predicted distressed was their small size. Without the Shapley value explanation framework, the biased prediction contribution for this feature would not be disclosed. We consider the discovery of the bias as important because companies can be rejected by a factor which in fact is out of their control. Thus, we argue the Shapley framework to have the ability to disclose possible bias. An important finding, as it opens for relevant human intervention when explanations are not satisfactory. For our case, the explanation framework could open for certain companies to have their loan application accepted despite their small size.

However, our case predicts implications for companies, not individuals. It is therefore less prone to ethical dilemmas. Banks who can use this model can argue that they find the variable *X17: Log of total assets* relevant. It is not their job to take risks and finance startups. As Shapley values can be applied to other problems as well, we find this interesting to discuss. Discrimination is not legal in the EU according to article 21 in the General Data Protection Regulation (European Parliament, 2012). To propose a similar example, a car insurance

company can use a machine learning model for risk assessments. It would be tempting to include variables for gender and age in this model as young male drivers are known to crash more often (*Road traffic injuries*, 2020). Including gender as a variable, however, is considered discrimination and is thus not allowed. The first regulation against artificial intelligence (AI) is yet to be established, but the European Commission has developed a proposal for a regulation specific to AI (European Commission, 2021). At the time of writing, there seems to be gaps to avoid current regulations. A possible bypass of regulations can be to include variables which correlate with the discriminating variables. With increased focus on regulations in AI, a common ground and harmonized rules would be beneficial. Shapley values is a framework for detecting such illegal actions and is therefore a useful tool.

Another implication in practice with Shapley values is the computational problem. Imagine again a bank using a complex model such as xgboost to determine which companies to provide loans to. To improve our model, other features could be included. Those companies receiving a decline could refer to their right to an explanation. With increasing features, computational time will grow exponentially. If there are many companies asking for explanations, a computational problem would appear. In practice, this would mean that the bank needs to take this possibility into account. Either reduce features or deal with the computational time. This issue is probably not apparent today, but with the increase in usage of machine learning models, this could have implications in practice in the future. We believe banks and creditors can use such explanations to provide reasons behind decisions made by automatic tools.

8 Conclusion

In this thesis we have explored how the Shapley value framework can be used to explain individual and global predictions. We will in this chapter summarize by looking at the research questions to discuss whether our results have answered them or not. Since we had three different research questions, we will answer them individually.

The first research question was: *How intuitive are Shapley value individual prediction explanations for black-box models?* To answer this question, we computed Shapley values for some selected predictions. The complexity of the black-box model in xgboost was discussed in chapter 5. While a decision tree was easily interpreted, xgboost on the other hand would prove quite difficult to understand, even for an expert. Shapley values did, however, make predictions by xgboost interpretable. We consider a simple explanation of how to interpret a Shapley value graph as sufficient to understand explanations, and we argue even a non-expert could understand it. The plots which illustrate whether a feature value either increases or decreases a probability is a useful tool to provide complete and intuitive explanations. While our case considers how Shapley values explain predictions made by the complex model xgboost, Shapley values can be applied to any model and give similar explanations. We can thus conclude Shapley values give intuitive explanations for black-box models.

The second research question was: *Can unfairness be disclosed by Shapley values in black-box models?* To find this out, we had to compute Shapley values for all predictions in the test set. Bias in the model could thus be detected, which could lead to discrimination. This was a difficult task due to the computational complexity of the method. By restricting ourselves to one sector in the data, we were nevertheless able to compute Shapley values for all predictions in the construction sector. The graphs provided indications of possible faults in the model, both in terms of economic unreasonable results and possible bias leading to discrimination and unfairness. As we discussed, *X17: Log of total assets* was interpreted by xgboost in an interesting way by how it was discriminating against small companies. To conclude whether there is a global bias in the model or not, we would be required to compute Shapley values for all sectors. It is, however, still challenging to conclude, as every feature must be considered in relation to each other. We can therefore just suspect a possible bias in our model. These results show that Shapley values can detect patterns in the model. How these patterns and unfairness

are interpreted and weighted is up to decision-makers. We do conclude, nevertheless, Shapley values to be a satisfactory tool in assessing unfairness in a black-box model.

The last research question was: *To what extent can Shapley values provide explanations for how black-box models consider different feature values?* To answer this question, our analysis in section 6.3 is relevant. Due to the computational complexity, we had to restrict our analysis to one sector in the data. Since we computed Shapley values for all companies in the construction sector, we can say something about how xgboost considers these companies. **Figure 12** presents exactly this, in terms of how xgboost examines features when predicting financial distress in the construction sector. We discussed earlier some clear trends and patterns in the plots. Imagine if the results were noise and uncorrelated data points, then it would be impossible to understand the model's behavior. Since we have obtained some clear results, we can say for example that a negative *X1: ROA* feature value will in most cases lead to Shapley values which increase the predicted distress probability. *X1: ROA* is also an important feature as shown by our global importance figure. Since we obtained such strong trends and patterns for the chosen sample size, we conclude that Shapley values can provide explanations for how black-box models consider feature values.

Future Work

Finally, we would like to discuss potential for future work in the era of interpretable machine learning. First, computing Shapley values for the whole data is possible. However, it will require a powerful machine and time. We feel our analysis answers our research questions, but an analysis of the whole data would validate our results even further. Extending the analysis could incorporate comparisons of differences between sectors and a complete global feature importance analysis. Doing this will also enable for a better investigation of the illogic economic patterns. Reasons based on sector specific patterns could either be confirmed or excluded. Other future work could be to compute Shapley values for other prediction models. The results could enable for further understanding of the differences between machine learning models. It can also be interesting to compare how local and global explanations differ between models.

9 References

Aas, K., Jullum, M., & Løland, A. (2021). Explaining individual predictions when features are dependent: More accurate approximations to Shapley values. *Artificial Intelligence*, *298*, 103502. https://doi.org/https://doi.org/10.1016/j.artint.2021.103502

Aas, K., Nagler, T., Jullum, M., & Løland, A. (2021). Explaining predictive models using Shapley values and non-parametric vine copulas. *arXiv preprint arXiv:2102.06416*.

Altman, E. I. (1968). FINANCIAL RATIOS, DISCRIMINANT ANALYSIS AND THE PREDICTION OF CORPORATE BANKRUPTCY. *Journal of Finance*, *23*(4), 589-609. https://EconPapers.repec.org/RePEc:bla:jfinan:v:23:y:1968:i:4:p:589-609

Angwin, J., Larson, J., Mattu, S., & Kirchner, L. (2016, 23-04-2016). *Machine Bias*. ProPublica. Retrieved 05.20.2021 from https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., & Müller, K.-R. (2010). How to explain individual classification decisions. *The Journal of Machine Learning Research*, *11*, 1803-1831.

Bibal, A., Lognoul, M., de Streel, A., & Frénay, B. (2020). Legal requirements on explainability in machine learning. *Artificial Intelligence and Law*. https://doi.org/10.1007/s10506-020-09270-4

Branch, B. (2002). The costs of bankruptcy: A review. *International Review of Financial Analysis*, *11*(1), 39-57. https://doi.org/https://doi.org/10.1016/S1057-5219(01)00068-0

Breiman, Leo, Friedman, Jerome, J. Stone, Charles, Olshen, & R.A. (1984). *Classification and Regression Trees*.

Brownlee, J. (2021). *A Gentle Introduction to XGBoost for Applied Machine Learning*. Retrieved 18.03.2021 from https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/

Carvalho, D. V., Pereira, E. M., & Cardoso, J. S. (2019). Machine Learning Interpretability: A Survey on Methods and Metrics. *Electronics*, *8*(8), 832. https://www.mdpi.com/2079-9292/8/8/832

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining,

Chen, T., He, T., Benesty, M., Khotilovich, V., Tang , Y., Cho, H., Chen, K., Mitchell, R., Cano, I., Zhou, T., Li, M., Xie, J., Lin, M., Geng, Y., Li, Y., & XGBoost contributors. (2021). *Extreme Gradient Boosting*. https://github.com/dmlc/xgboost

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Duval, A. (2019). Explainable Artificial Intelligence (XAI). https://www.researchgate.net/publication/332209054_Explainable_Artificial_Intelligence_XAI

Dziugaite, G. K., Ben-David, S., & Roy, D. M. (2020). Enforcing Interpretability and its Statistical Impacts: Trade-offs between Accuracy and Interpretability. *arXiv preprint arXiv:2010.13764*.

Regulation of the European Parliament and of the council laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain union

legislative acts, (2021). https://digital-strategy.ec.europa.eu/en/library/proposal-regulation-laying-down-harmonised-rules-artificial-intelligence

Charter of fundamental rights of the European Union, (2012). https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:12012P/TXT&from=EN

Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance), (2016). https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679

Fernando, J. (2021). *Debt-To-Equity Ratio (D/E)*. Retrieved 05.26.2021 from https://www.investopedia.com/terms/d/debtequityratio.asp

Fisher, A., Rudin, C., & Dominici, F. (2019). All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. *Journal of Machine Learning Research*, *20*(177), 1-81.

Freitas, A. A. (2014). Comprehensible classification models: a position paper. *SIGKDD Explor. Newsl.*, *15*(1), 1–10. https://doi.org/10.1145/2594473.2594475

Frye, C., de Mijolla, D., Cowton, L., Stanley, M., & Feige, I. (2020). Shapley-based explainability on the data manifold. *arXiv preprint arXiv:2006.01272*.

Goodman, B., & Flaxman, S. (2017). European Union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, *38*(3), 50-57.

Haugh, M. (2016). An Introduction to Copulas. http://www.columbia.edu/~mh2078/QRM/Copulas.pdf

High-Level Expert Group on AI. (2019). *Ethics guidelines for trustworthy AI* [Report]. https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai

Interpretability. (2021). Mathworks. Retrieved 05.21.21 from https://se.mathworks.com/discovery/interpretability.html

James, Gareth, Witten, Daniela, Hastie, Trevor, Tibshirani, & Robert. (2017). *An Introduction to Statistical Learning: with Applications in R* (First ed.).

Lundberg, S. (2018, 17.04.2018). *Interpretable Machine Learning with XGBoost*. Retrieved 04.01.2021 from https://towardsdatascience.com/interpretable-machine-learning-with-xgboost-9ec80d148d27

Lundberg, S., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *arXiv preprint arXiv:1705.07874*.

Molnar, C. (2020). *Interpretable machine learning*. Lulu. com.

Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R., & Yu, B. (2019). Interpretable machine learning: definitions, methods, and applications. *arXiv preprint arXiv:1901.04592*.

Ohlson, J. A. (1980). Financial Ratios and the Probabilistic Prediction of Bankruptcy. *Journal of Accounting Research*, *18*(1), 109-131. https://doi.org/10.2307/2490395

Palczewska, A., Palczewski, J., Marchese Robinson, R., & Neagu, D. (2013). *Interpreting random forest models using a feature contribution method*. https://arxiv.org/abs/1312.1121

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016b). Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*.

Riberio, M. T., Singh, S., & Guestrin, C. (2016a). «Why Should I Trust You?» Explaining the Predictions of Any Classifier. https://arxiv.org/pdf/1602.04938.pdf

Road traffic injuries. (2020). WHO. Retrieved 05.25.2021 from https://www.who.int/news-room/fact-sheets/detail/road-traffic-

injuries?fbclid=IwAR2A52TkR2hnZ7cvIPdWzSUXflXo2waPEm4iF2oDanVlWJpSuK47-f0F8NE

Salleh, M., Talpur, N., & Hussain, K. (2017). *Adaptive Neuro-Fuzzy Inference System: Overview, Strengths, Limitations, and Solutions*. https://doi.org/10.1007/978-3-319-61845-6_52

Sellereite, N., Jullum, M., Redelmeier, A., Løland, A., Wahl, J. C., Lingjærde, C., & Norsk Regnesentral. (2021). *Prediction Explanation with Dependence-Aware Shapley Values*. https://norskregnesentral.github.io/shapr/

Shapley, L. S. (1953). A value for n-person games. *Contributions to the Theory of Games*, *2*(28), 307-317.

Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T., & Zeileis, A. (2008). Conditional variable importance for random forests. *BMC Bioinformatics*, *9*(1), 307. https://doi.org/10.1186/1471-2105-9-307

Therneau, T., Atkinson, B., & Ripley, B. (2019). *Recursive Partitioning and Regression Trees*. https://github.com/bethatkinson/rpart

Zhang, G. N., & Ye, F. (2019). *Predicting financial distress in Norway: using logistic regression and random forest models*

Zhou, Z.-H. (2009). Ensemble learning. *Encyclopedia of biometrics*, *1*, 270-273.