Erasmus University Rotterdam

Master thesis Econometrics:
Business Analytics & Quantitative Marketing

Uncovering the "black box"

A study on how to make machine learning techniques more interpretable in an application to loan default prediction

Jowita Osinga
479363

Supervisor
Dr. Michel van de Velden
Second assessor
Dr. Andreas Alfons

Supervisor
Xander van den Berg

September 28, 2018

Abstract

This thesis demonstrates how machine learning techniques can be made more interpretable using several methods that enhance the transparency of these techniques. First, different aspects of interpretability are outlined and various measures that allow for a comparison across methods are presented. After that, different explanation methods are applied in an empirical study on loan default prediction. It is illustrated that global interpretation can be attained by constructing variable importance measures or partial dependence plots, which give insight into the relationship between the explanatory variables and the outcome. On a local level insights into the model can be achieved by constructing feature contributions or a local linear approximation of the model, giving a detailed explanation on the prediction outcome of single instances. Some global understanding into how the model reaches its classification results can be given by inspecting the variable interactions that the classifier exploits. At last, single tree approximation can be performed, providing an approximating graphical representation of the model, allowing for understanding on a both global and local scale how decisions in the model are reached.

Keywords: *interpretation, machine learning, black box*

Contents

1 Introduction

Since the mid 1980s, there has been a rapid growth in algorithmic modelling applications and methodology. This has together with the increase in computing power of machines and the rise of 'Big Data' led to a new discipline, called machine learning (Jordan & Mitchell, 2015). Machine learning can be seen as a research community where one does not assume any underlying distribution of the data. The use of machine learning techniques provides models with high flexibility and high predictive power (see, e.g., Strobl, Boulesteix, Kneib, Augustin, & Zeileis, 2008; Palczewska, Palczewski, Robinson, & Neagu, 2014). Therefore, in settings where complex prediction problems need to be addressed, machine learning is often for practical reasons preferred. The applications include, among others, speech recognition, image recognition, prediction in financial markets and handwriting recognition (Breiman et al., 2001).

As noted by Breiman et al. (2001), in prediction problems accuracy and interpretability are in conflict. For instance, while a linear regression or a logistic regression give a relative straightforward interpretation of the relation between the explanatory and response variables, their accuracy is in many cases less than that of a machine learning technique. On the other hand, machine learning techniques are often described as opaque or 'black box': they provide little explanatory insight into the relationship between the model explanatory variables and outputs.

Interpretability in machine learning has often been neglected by focusing on accuracy (Strobl et al., 2008). However, the lack of interpretability of black box models raises both practical as well as ethical issues. For instance, in systems where safety plays a crucial role, such as self-driving cars, robotic assistants and personalised medicine, machine learning is increasingly used; this however brings the risk of possibly making the wrong decisions by learning from spurious correlations or due to a bias in the training data (Freitas, 2014; Ribeiro, Singh, & Guestrin, 2016; Guidotti, Monreale, Turini, Pedreschi, & Giannotti, 2018). In particular, data may contain human biases and prejudices which, once these are detected by the black box algorithm, can lead to automated discrimination and racism (Lowry & Macpherson, 1988; Caliskan-Islam, Bryson, & Narayanan, 2016).

As a consequence, explanation in machine learning is receiving increasingly more attention. Within the scientific community, there exist different views on explainability (Guidotti et al., 2018). Since there is no systematic classification of how interpretability should be used, this thesis addresses this issue by discussing different aspects that characterize an interpretable model and presenting several measures that allow for comparison of methods in terms of interpretability. Furthermore, the trade-off between the predictive performance and the interpretability for (non-) machine learning algorithms is addressed by firstly illustrating the difference in performance of logistic regression against machine learning techniques, after which existing techniques that increase the interpretability of black box methods are presented.

The focus lays on the explanation of two commonly used machine learning techniques: random forests and neural networks (Breiman, 2001; Olden & Jackson, 2002; Strobl et al., 2008; Palczewska et al., 2014). To illustrate and compare the results of the explanation methods, an empirical study is performed on loan performance data of the US mortgage financier Fannie Mae. The objective is to assess which loans will default in the coming year and to provide relevant insights into which factors affect the probability of default.

Techniques that enhance the transparency and reliability of machine learning algorithms in terms of interpretability are referred to as explanation methods throughout this thesis. We focus on the following: variable importance measures, feature contributions, variable interactions, partial dependence plots, local linear approximation and single tree approximation. This thesis aims to address the following research question:

How can we make machine learning algorithms more interpretable by using explanation methods in the case of loan default prediction?

This question is further divided in three subquestions:

1. What is an interpretable model?
2. How can we explain how black box models link observations to outcomes?
3. How can we get general insights into machine learning techniques?

The first subquestion addresses how to define interpretability and how to measure it. The second subquestion is related to gaining insights into which observation characteristics in the data lead to particular outcomes in the black box model, such as identifying the most important features or finding the relationship between the explanatory and response variables. The third subquestion aims to explain what the general logic behind the black box model is, such as describing how predictions are made or how the data are used in classifying objects.

The remainder of this thesis is structured as follows. First, the concept of an interpretable model is outlined in Section 2, answering the first subquestion. Then, methodology on random forests and neural networks is outlined in Section 3, after which in Section 4 techniques for getting more insights into these methods are described. Subsequently, in Section 5 the data used in this paper are presented. Section 6 outlines the results of the empirical study, which help in answering the second and third subquestion. Finally, Section 7 summarizes all findings, answers the research question and provides some angles for further research.

2 Interpretability

In this section the first subquestion of this thesis is addressed by outlining how to define interpretability of a model. We present some models that are recognised as interpretable and discuss how black box methods can become more interpretable. Also, several measures that are used to assess the interpretability of the models and methods in this thesis are discussed.

In machine learning, interpretability is defined as the ability to explain or to present in understandable terms to humans (Doshi-Velez & Kim, 2017). Several recognised interpretable models currently exist. For instance, linear models are considered easily interpretable because it is possible to identify the sign and magnitude of the influence of each explanatory variable on the response variable by interpreting the regression coefficients. Also, it is possible to detect which change in the output occurs by adjusting the input (Ribeiro et al., 2016). Another comprehensible model is a decision tree, which is considered easily interpretable because of its graphical representation. From the graph the most important features can be detected and by following each path from the root to the leaf node, local patterns can be detected (Freitas, 2014).

On the other hand, 'black box' models such as random forests and neural networks are not considered to be easily interpretable. In case decisions have to be made based on the prediction outcome of a model, an explanation of how the outcome predictions are determined or insights into the underlying logic of the model are often required (Guidotti et al., 2018). However, discussion remains what a good explanation is and how different explanations can be compared across methods. Therefore, we now turn to outlining how we measure and assess interpretability throughout this thesis, which enables to make a comparison across the different methods.

As already pointed out, in machine learning there is often a trade-off between accuracy and interpretability. Accordingly, we start by assessing the accuracy of logistic regression, random forests and a neural networks on the default prediction of our data (see Sections 5 and 6). The accuracy is measured by the accuracy score (i.e., the overall percentage of instances correctly predicted) and recall (i.e., the percentage of default cases correctly predicted). Subsequently, interpretability of these models is assessed by taking into account whether a model is globally or locally interpretable. If we are able to follow the reasoning leading to all different outcomes in general, we speak of global interpretability. Instead, local interpretability is defined by the situation in which we are able to apprehend from case to case the reasons for single predictions (Guidotti et al., 2018).

To make machine learning techniques more understandable, several techniques that try to explain the black box methods discussed in this thesis (i.e., random forests and neural networks) are applied. Each explanation method is then assessed in terms of global and local interpretability. We verify whether the magnitude and sign of the effect of each explanatory variable on the response variable can be determined. The magnitude refers to the (relative) contribution of each covariate to the prediction outcome, whereas the size indicates which relationship the covariate has with the outcome (Freitas, 2014). Another aspect considered in defining interpretability of an explanation is whether a graphical depiction can be made of how the model works, providing an understanding of the overall logic of the model (Guidotti et al., 2018). At last, for the explanation method of the single tree approximation the fidelity is considered as a measure of accuracy, which determines to which extent the predictions of the approximating technique correspond to the outcome of the black box model.

3 Methodology machine learning techniques

3.1 Logistic regression

Logistic regression is a regression model that considers a binary dependent variable y and uses a non-linear logistic functional form. The logistic regression can be formulated as follows:

$$Pr(y_i = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X)}}, \tag{1}$$

which calculates the probability of object i to belong in class 1. It follows that the regression is linear in its parameters by considering the log-odds:

$$ln(\frac{Pr(y_i = 1)}{1 - Pr(y_i = 1)}) = \beta_0 + \beta_1 X. \tag{2}$$

A change in β can be interpreted as a change in the log-odds of belonging to class 1 versus not belonging in class 1. This allows for a simple interpretation of the effect of the independent variables on the outcome and therefore we decide to use a logistic regression as a benchmark, to which we compare the performance of random forests and neural networks.

The next section discusses the methodology on decision trees, which forms the basis on the methodology of random forests. Subsequently, Section 3.3 outlines random forests, after which Section 3.4 discusses artificial neural networks.

3.2 Decision trees

A decision tree is a structure that represents how examples are classified by partitioning the decision space. The structure is formed by nodes and branches, where at each node a single variable is considered at a certain split point. Decision trees can be used for prediction of numerical variables, leading to a so called regression tree, as well as categorical variables, resulting in a classification tree.

Several ways exist to decide on which variable and at which threshold to partition the tree. A popular methodology is CART (classification and regression tree; Breiman, Friedman, Olshen, & Stone, 1984). In the case of classification trees, the optimal splitting of the variables is found by deriving the impurity of the nodes, of which the most commonly used measure is the Gini impurity. For simplicity, we restrict to the case of binary splits to illustrate how this measure is constructed. The Gini impurity $i(\tau)$, which calculates the effect of a split at node τ on the classification of the tree, is calculated as:

$$i(\tau) = 1 - p_0^2 - p_1^2 \tag{3}$$

where $p_k = \frac{n_k}{n_\tau}$ is the fraction of the n_k samples from class $k = \{0, 1\}$ out of the total of n_τ samples at node τ. The decrease in Gini impurity, $\Delta i(\tau)$, is defined as:

$$\Delta i(\tau) = i(\tau) - p_l i(\tau_l) - p_r i(\tau_r), \tag{4}$$

where p_l is the fraction of samples of the left split and p_r corresponds to samples of the right split and the split leading to the maximum decrease in impurity is selected. Choosing this maximum decrease in impurity is useful, since this minimizes the impurity, allowing for classification of the most difficult class distribution.

An important tuning parameter in the tree building process is the tree size. A tree that is too large might overfit the data, which means that the model has effectively memorized existing data points and is not able to predict unseen data correctly anymore. On the other hand, a tree which is too small might not capture the structure of the data well. The preferred strategy is to grow a tree until a specified minimum node size is reached, after which this large tree is pruned. Pruning is the process of reducing the size of the tree. Branches are removed to a point where the cost complexity criterion is minimized, which looks at the trade-off between tree size and the misclassification error.

In the case of regression trees, we find the best partition of the data by considering all variables at all split points. A greedy algorithm is applied, which finds the splitting variable x_j and split point η by constructing pairs of half-planes in the data:

$$R_1(x_j, \eta) = \{X | x_j > \eta\} \text{ and } R_2(x_j, \eta) = \{X | x_j \leq \eta\} \tag{5}$$

and solve

$$min(\underset{\bar{y}_1}{min} \sum_{x_i \in R_1(x_j,\eta)} (y_i - \bar{y}_1)^2 + \underset{\bar{y}_2}{min} \sum_{x_i \in R_2(x_j,\eta)} (y_i - \bar{y}_2)^2). \tag{6}$$

where $\bar{y}_i$ is the average value of y_i, given $x_i \in R_1(x_j, \eta)$ or given $x_i \in R_2(x_j, \eta)$, respectively. After the optimal split is found, the data is partitioned in two regions and this procedure is repeated for the remaining parts of the tree (Friedman, Hastie, & Tibshirani, 2001).

Another popular tree-building procedure is C4.5 (Quinlan, 1993), which decides on the splits of the tree by the entropy. The entropy of a tree measures how informative a variable is in splitting the data (Martens, Baesens, Van Gestel, & Vanthienen, 2007). This can be formulated as follows. Let p_0 and p_1 be the proportion of examples in class 0 and 1, respectively, with which entropy of subset S is calculated by:

$$Entropy(S) = -p_1 \log_2(p_1) - p_0 \log_2(p_0) \tag{7}$$

The entropy measures how well the data is ordered with respect to the classes, where an entropy of 1 implies $p_0 = p_1 = 0.5$, which means there is maximal disorder across the classes, and 0 implies $p_1 = 0$ or $p_0 = 0$, which is when all observations fall into the same class. To decide on which variable to split the tree, the Gain ratio is used, which is calculated as

$$Gain(S, x_j) = Entropy(S) - \sum_{v \in values(x_j)} \frac{|S_v|}{|S|} Entropy(S_v) \tag{8}$$

where $Gain(S, x_j)$ is the expected reduction in entropy because of splitting on variable x_j and S_v is the subset of the sample where variable x_j has value v.

Both CART as well as C4.5 suffer from overfitting and selection bias towards covariates with many possible splits, therefore Hothorn, Hornik, and Zeileis (2006) propose using trees based on conditional inference. A conditional inference tree is an algorithm that separates the variable selection from the splitting procedure (Sarda-Espinosa, Subbiah, & Bartz-Beielstein, 2017). The first step in the algorithm entails the variable selection, which is based on permutation tests that try to distinguish between significant and insignificant improvements, avoiding the variable selection bias that is observed with CART. Each node of the tree is represented with a vector of case weights $\mathbf{w} = (w_1, ..., w_n)$ for the n observations. The elements of the vector have nonzero elements when the observations are elements of the node and are zero otherwise. For the case weights $\mathbf{w}$ the partial null hypothesis of independence of the response variable with each of the m covariates x_j is $H_0^j : D(Y|x_j) = D(Y)$, with the global hypothesis $H_0 = \bigcap_{j=1}^m H_0^j$. If the latter hypothesis can be rejected, the association between x_j and Y is calculated and the x_j^* with strongest association is selected for the splitting. In case the hypothesis cannot be rejected, the tree expansion at that particular node is stopped (Hothorn et al., 2006).

The second step is the splitting of the variables. Permutation tests are used to find the optimal binary split of the selected covariate x_j^*. A set $A \subset x_j^*$ divides samples into two disjoint sets $\{Y|w_i > 0 \text{ and } \quad x_{j^*i} \in A\}$ and $\{Y|w_i > 0 \quad \text{and} \quad x_{j^*i} \notin A\} \quad \text{for} \quad i = 1, ..., n$. The optimal A^* is chosen by maximizing the discrepancy between the resulting samples. Subsequently, the results from the two steps are recursively repeated until the stopping criteria are met.

Decision trees enjoy the advantage of providing understandable tools for prediction. However, one problem of decision trees is that small changes in the data can lead to different splits in a tree. This can be explained by the hierarchical structure of the tree: an error at the top of the tree has an effect on all the splits below it (Friedman et al., 2001). The next section discusses the methodology on random forests, which overcome this issue.

3.3 Random forests

Random forests is a technique which builds an ensemble of decision trees. Each tree is grown by drawing random bootstrap samples from the training data. Single trees are fitted to each sample. Depending on the type of response variable(s), the predictions of the individual trees are averaged or the resulting prediction is based on the class with the most number of votes, resulting in the random forests prediction. The steps of random forests are illustrated in Algorithm 1 (Friedman et al., 2001; Breiman et al., 1984).

The hyperparameters that can be tuned in random forests are the number of decision trees and the number of variables to consider for each split. The default number of trees to

Algorithm 1 Random forests

```
for $t = 1$ to $T$ do
    Draw a bootstrap sample $\mathbf{Z}$ from the training data
    Grow a tree $Q_t$ to $\mathbf{Z}$: let $d$ and $d_{min}$ be the (minimum) node size. Then
    while $d \geq d_{min}$ do
        Select $m^*$ variables at random from $m$ variables.
        Pick the best variable/split-point among the $m^*$ based on splitting criterion$^\dagger$.
        Split the node into two daughter nodes.
return ensemble of trees $\{Q_t\}_1^T$

Make a prediction at a new point $x$:
        Classification: Let $\hat{C}_t(x)$ be the class prediction of the $t^{th}$ random forest tree.
            Then $\hat{C}_{rf}^T(x) = majority\,vote\{\hat{C}_t(x)\}_1^T$
        Regression: $\hat{f}_{rf}^T(x) = \frac{1}{T}\sum_{t=1}^{T} Q_t(x)$
```

† Gini for classification trees, mean decrease in accuracy for regression trees grow (T in line 1, Algorithm 1) is typically 500 trees, whereas the number of variables to consider for splitting (m^* in line 5, Algorithm 1) is often $\sqrt{m}$. These parameters can be tuned by performing cross-validation; the out-of-sample prediction accuracy is assessed across all possible combinations of tuning parameter values, where the chosen parameter values are the ones leading to the highest accuracy.

Random forests have been shown to achieve higher classification and prediction results with less variance, when comparing to single trees (see, e.g., Breiman, 1996, 2000; Breiman et al., 2001; Breiman, 2004; Biau, 2012; Henelius, Puolamäki, Boström, Asker, & Papapetrou, 2014; Strobl et al., 2008). Also, this method can handle a large amount of predictor variables, without overfitting, i.e., the situation in which the model has effectively memorized existing data points and is not able to predict unseen data correctly anymore. Other advantages for random forests include easy implementation and the short computation time, given little to no tuning is performed (Friedman et al., 2001; Strobl et al., 2008; Biau, 2012). However, a disadvantage of random forests is that the easy interpretation of a tree is lost.

Random forests are used in this thesis to predict loan default. The number of trees and number of variables to consider for splitting are determined by 10-fold cross-validation, to achieve the highest out-of-sample prediction accuracy. The results are compared in terms of accuracy and interpretation with logistic regression and artificial neural networks. The methodology of the latter is described in the next section.

3.4 Artificial neural networks

Artificial neural networks (ANNs) are mathematical models which are inspired by how the human brain works by replicating the behaviour of neurons. ANN models are trained by identifying correlated patterns between explanatory and response variables. ANNs are able to make a flexible function approximation of any data, even if the data are imprecise and noisy (Lek & Guégan, 1999). Besides, ANNs show a high degree of prediction accuracy, which is one of its most important advantages (Thrun, 1993).

Originally, ANNs were developed to model biological functions. More recently, they have been shown to be very powerful tools for statistical (predictive) modelling, since they are able to map the output value as a non-linear function of the input data. Therefore, many researchers find ANNs to be an attractive alternative to traditional statistical approaches such as linear regression. Nowadays, ANNs are often applied in the fields of pattern, image and speech recognition (Jordan & Mitchell, 2015).

In this thesis, one commonly used form of artificial neural networks is considered, which is a multi-layer feed-forward neural network trained by a backpropagation algorithm, also called a backpropagation network (BPN). The BPN is a supervised learning procedure, which

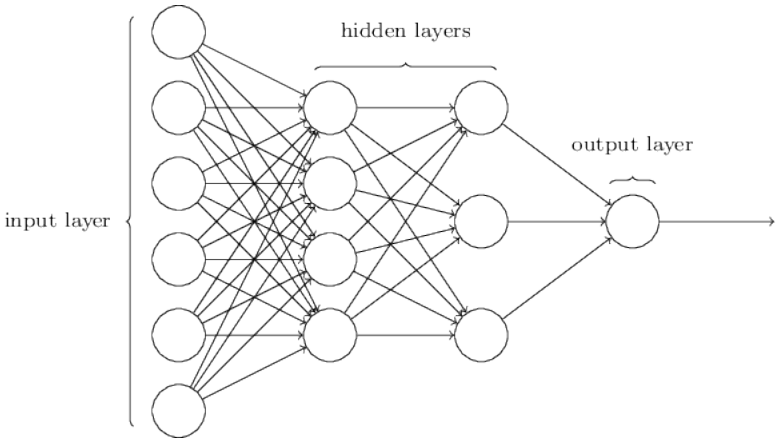


Figure 1: Graphical representation of a neural network with two hidden layers

provides a mechanism to model complex relationships between variables, see Figure 1 for a schematic overview of this network. The nodes, also called neurons, are arranged in layers, with the information flowing through the hidden layers from the input to the output layers (Lek & Guégan, 1999). The input layer consists of all the explanatory variables of the model, whereas the response variable forms the output layer.

To understand how these neurons are modelled, Figure 2 depicts a perceptron, which is a construction with one neuron from the network, connected with m other neurons, receiving m inputs and leading to 1 output. In a multi-layer model, this output serves as input for the consecutive layer. Each input variable x_j is assigned a weight w_j, after which the linear combination of all input variables and a bias term b are passed on to the activation function f, which leads to the output

$$f(\sum_{j=1}^{m} w_j x_j + b). \tag{9}$$

The initial weights, which are either randomly determined or set by the researcher, are updated based on how well the model fits the data. The training algorithm for neural networks is called gradient descent, which is a first-order iterative optimization algorithm

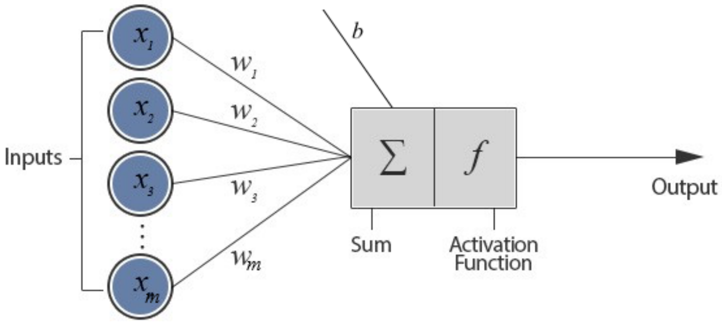


Figure 2: Model for artificial neuron

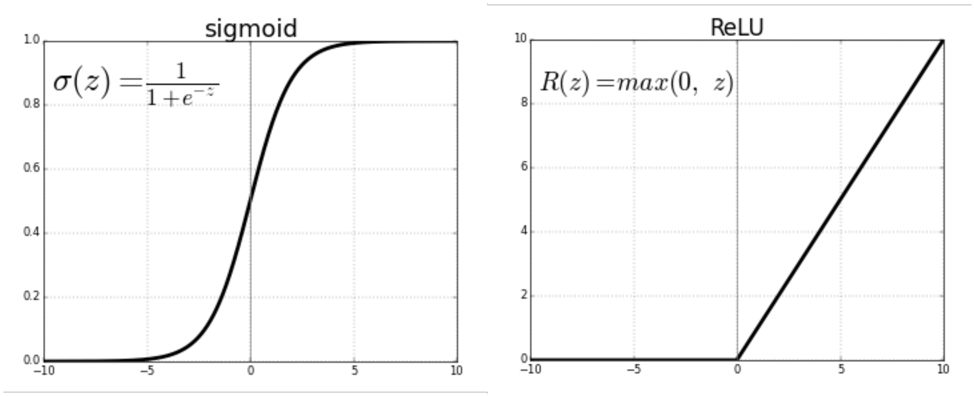


Figure 3: Sigmoid and ReLU (Rectifier Linear Unit) activation functions for neural networks

that finds the minimum of the difference between the actual and predicted outcome of the model.

Many different activation functions exist for neural networks, of which the most commonly used are the sigmoid function and the rectifier linear unit (see Figure 3). The sigmoid function is useful in problems where the objective is to predict a probability, since the function gives an output between 0 and 1. Furthermore, the function is differentiable and monotonic, which makes it possible to train the model by gradient descent. The rectifier function is often used for the hidden layers, mainly because of its quick convergence. It has a non-zero derivative at positive inputs which allows gradient based learning. However, all negative values become zero, which causes that neurons with negative valued inputs are unable to update their weights due to the zero-valued gradients.

As described, the BPN can construct complex non-linear functions through the use of multiple nodes and several hidden layers. Once the network contains (too) many nodes and hidden layers, neural networks suffer from a lack of explanatory insight, since this complicates the identification of the variable contribution to the outcome, which is problematic if the ANN is to be used in some domains where interpretation is crucial. Alternatively, ANN should be provided with the possibility of determining under which conditions the ANN makes certain decisions leading to a particular outcome, which allows for some degree of understanding of the model (Andrews, Diederich, & Tickle, 1995).

In this thesis, an artificial neural network is constructed using the *keras* and *Tensorflow* libraries in Python. We decide to create a net consisting of two hidden layers, since this allows for a non-linear model. After tuning the model with 10-fold cross validation, we decide on the size of the hidden layers of 9 and 10, respectively. The rectifier linear unit (ReLU) function is used as an activation function for the hidden layers, whereas the sigmoid function is used for constructing the output variable, i.e., the probability of default. The next section outlines methods that help in explaining black box models.

4 Insights into machine learning techniques

This section outlines several methods to gain more understanding into black box techniques. We start by discussing several variable importance measures, after which variable interactions are addressed. Next, a method for understanding the relationship between the explanatory and response variables is explained, after which a method for local interpretation of a model is presented. Subsequently, methods that allow for an approximating graphical depiction of the model are elaborated on.

4.1 Variable importance measures

We start by presenting variable importance measures for random forests, after which measures for neural networks are outlined. There exist several variable significance measures, which give some interpretation to random forests. Originally, Breiman (2001) proposes two measures: the first being variable importance, which is based on impurity, assessing at each node how well a potential split separates the cases of the resulting classes. Accumulating the optimal splits for all nodes τ in all trees T in the forest, for all variables θ leads to the Gini importance:

$$Gini(\theta) = \sum_{T} \sum_{\tau} \Delta i_{\theta}(\tau, T) \tag{10}$$

where $\Delta i(\tau)$ is formulated in Formula 4. This quantity provides a measure for feature importance, as it provides a relative ranking of all features of the training set and indicates how often a feature is selected for a split and how much they contribute to the decrease in impurity (Menze et al., 2009).

The second measure is calculated by the average loss of accuracy across all trees, caused by permutation of a particular variable. Thus, the difference in prediction accuracy before and after permuting the particular variable is calculated. Following Strobl et al. (2008), let $\mathcal{B}^{(t)}$ be the holdout sample for a tree t, with $t \in \{1, ..., T\}$. The variable importance of variable x_j for tree t can be formulated as follows:

$$VI^{(t)}(x_j) = \frac{\sum_{i \in \mathcal{B}} I(y_i = y_i^{(t)})}{|\mathcal{B}^{(t)}|} - \frac{\sum_{i \in \mathcal{B}} I(y_i = y_{i,\pi_j}^{(t)})}{|\mathcal{B}^{(t)}|}, \tag{11}$$

where I is an indicator function with $\hat{y}_i = y_i^{(t)}$ being the prediction of an observation i before permutation of x_j and $\hat{y}_i = y_{i,\pi_j}^{(t)}$ the prediction after permutation. The variable importance score for each variable can be derived by taking the average importance of all trees:

$$VI(x_j) = \frac{\sum_{t=1}^{T} VI^{(t)}(x_j)}{T}. \tag{12}$$

Unfortunately, the Gini importance shows a bias towards variables with many categories or continuous variables (Strobl, Boulesteix, Zeileis, & Hothorn, 2007; Strobl et al., 2008; Palczewska et al., 2014), as it is easier to find a cutpoint providing a high Gini score for variables with more potential cutpoints (Strobl et al., 2007). The permutation variable importance has been shown to be biased as well. In the case of correlated attributes, permutation does not accurately measure the variable importance (Strobl et al., 2008; Henelius et al., 2014). The null hypothesis of this importance measure is the independence between the predictor x_j and response Y. Permuting the variable x_j, causing a deviation from this hypothesis, could imply either the dependence of x_j and Y, or the dependence of x_j and one of the other predictor variables. The problem is that a deviation from the null hypothesis is interpreted as a dependence of x_j and Y, hereby implicitly preferring correlated variables by assigning them higher importance scores (Strobl et al., 2008; Smith, Ellis, & Pitcher, 2011).

A remedy to the biased permutation importance measures is proposed by Strobl et al. (2008), who develop a conditional permutation scheme. The difference with the regular permutation scheme proposed by Breiman (2001) is that each variable x_j is only permuted within partitions of the values of the predictors that are correlated with x_j, which allows to preserve the correlation structure with other predictor variables. This alternative approach computes more accurate variable importance measures for random forests and suffers less from bias towards correlated covariates, as demonstrated in the simulation studies of Strobl et al. (2008) and Smith et al. (2011). Similarly, concerning the biased Gini importance measure, it is suggested to use classification trees based on a conditional inference framework, instead of based on CART (Hothorn et al., 2006; Strobl et al., 2007). This framework provides more accurate importance for continuous variables and covariates with many categories.

Another variable importance measure is proposed by Palczewska et al. (2014), who introduce feature contributions. This method can be defined as follows: denote the set of classes by $C = \{C_1, C_2, ..., C_K\}$ and define P_K as the set of probabilities of an instance belonging to a certain class;

$$P_K = \{(p_1, ..., p_K) : \sum_{k=1}^{K} p_k = 1 \,\text{and}\, p_k \geq 0\}. \tag{13}$$

If a tree t predicts that an instance i belongs to a certain class C_k then $\hat{y}_{i,t} = e_k$, with e_k being a vector with 1 at position k. The final prediction is therefore the average over all trees T:

$$\hat{y}_i = \frac{1}{T}\sum_{t=1}^{T}\hat{y}_{i,t}. \tag{14}$$

Since $e_k \in P_k$, $\hat{y}_i \in P_K$ and the prediction of the random forests is the class C_K for which the k-th coordinate of $\hat{y}_i$ is the largest (Palczewska et al., 2014).

The derivation of feature contributions can be illustrated for binary classifiers as follows: firstly, the local increments of the feature contributions are calculated for each tree. A local increment of feature γ between the parent and child node is as follows:

$$LI_{\gamma}^{child} = \begin{cases} p^{child} - p^{parent}, & \text{if the split in the parent is performed over feature } \gamma, \\ 0, & \text{otherwise,} \end{cases} \tag{15}$$

where p^{τ} is the fraction of instances in a node τ belonging to the first class. The feature contribution of a feature γ in tree t for instance i is equal to

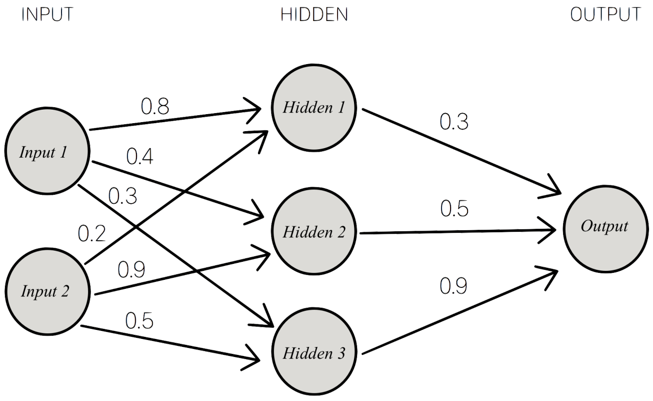
$$FC_{i,t}^{\gamma} = \sum_{\tau} LI_{\gamma}^{\tau}. \tag{16}$$

Averaging over all trees provides the contribution of a feature γ for instance i:

$$FC_{i}^{\gamma} = \frac{1}{T}\sum_{t=1}^{T} FC_{i,t}^{\gamma}. \tag{17}$$

The proposed feature contributions provide understanding into what the most significant variables are and what their contribution is towards predictions made for individual classes. Unlike variable importance measures illustrated by Equations 12 and 10, feature contributions are calculated separately for each instance. This provides detailed information about the relationship between variables and the predicted value on a local level, but can also be interpreted on a global scale by taking the median of the feature contributions, which is demonstrated in the results of the empirical study of this thesis.

Concerning artificial neural networks, variable importance measures can also be constructed. Garson (1991) proposes a method that, by using the connection weights in the neural network, determines the relative importance of each explanatory variable on the model outcome. A disadvantage of this method is that only the absolute value of the connection weights is taken into account, leading to misleading results since the influence of a negative weight is not counteracted by a positive weight (Olden & Jackson, 2002). Besides, Garson's algorithm is only applicable to neural networks with a single hidden layer and is therefore not applicable in the context of our research, since our neural network consists of two hidden layers.



*Importance*1: 0.8 * 0.3 + 0.4 * 0.5 + 0.3 * 0.9 = 0.71

*Importance*2: 0.2 * 0.3 + 0.9 * 0.5 + 0.5 * 0.9 = 0.96

Figure 4: Calculating variable importance in simple neural network

Another variable importance measure for neural networks is proposed by Olden and Jackson (2002), who calculate the importance of each variable by taking the product of the connection weights between one particular input and the output neuron, and summing this product across all hidden neurons connected to this particular input neuron. This approach differs from Garson's algorithm by the fact that the sign of the value of the connection weights is taken into account, which provides more accurate results, as shown in the simulation study of Olden, Joy, and Death (2004). Moreover, this approach can be used for a neural network with any amount of hidden layers.

For an illustration of the method of Olden and Jackson (2002) and the calculations of the importance of the input variables in the case of a single hidden layer, see Figure 4. The formal definition is as follows; let the weight connecting the ν^{th} neuron to the μ^{th} neuron in the consecutive layer be $w_{\nu\mu}$. Following this notation, the variable importance of the input variable γ given the output variable O can be derived by

$$VI(\gamma) = \sum_{\nu} w_{\gamma\nu} w_{\nu O} \tag{18}$$

where ν is the amount of neurons in the hidden layer.

In this thesis we consider a neural network with two hidden layers, for which we use the following notation. Let the weight connecting the ν^{th} neuron in the $(\lambda-1)^{th}$ layer to the μ^{th} neuron in the λ^{th} layer be $w^{\lambda}_{\nu\mu}$. The input layer is considered as the 0^{th} layer, whereas the output layer counts as the $(\Lambda+1)^{th}$ layer, with Λ being the total of hidden layers. Then for instance the connection weight of the first input variable to the third neuron in the first hidden layer is w^{1}_{13}. Following the same logic, the weight from the second hidden neuron in the second layer to a first neuron in the third layer (i.e., output layer) is w^3_{21}. The variable importance of input variable γ given output variable O can be calculated by

$$VI(\gamma) = \sum_{\nu} w^1_{\gamma\nu} \sum_{\mu} w^2_{\nu\mu} w^3_{\mu O}, \tag{19}$$

where ν is the amount of neurons in the first hidden layer and μ is the amount of neurons in the second hidden layer.

In the analysis in Section 6, the permutation, Gini and conditional variable importance measures on random forests are derived and the bias of the former two is assessed in our empirical example. Subsequently, the results of applying the feature contribution method of Palczewska et al. (2014) for random forests are discussed. In addition, the variable importance measure proposed by Olden and Jackson (2002) is applied on the results of our neural network. All results are assessed in terms of interpretability by the measures presented in Section 2. The next section presents a method that calculates the interactions between the variables of a random forests model.

4.2 Variable interactions

To gain more insights into the structure of the random forest classifier, Henelius et al. (2014) propose a randomization approach to find groups of variables whose interactions affect the predictions of the random forest classifier.

Let the dataset X be an $n \times m$ matrix of n observations and m attributes. First, the classifier is trained on the unrandomized data X and class label predictions of the classifier are made. Next, within-class permutation is performed, which entails that all observations predicted into the same class are permuted together. More specifically, each column m in the dataset X is permuted, leading to the permuted dataset X^*.

The optimal groups of attributes $\mathcal{S}$ are found by iteratively optimizing the groupings. Fidelity is used as an evaluation metric, measuring how much the classification performance changes due to randomization;

$$fid(\mathcal{S}) = E[I(y_i = y_i^*)], \tag{20}$$

where $I()$ is the indicator function, y_i are the predicted class labels on the unrandomized data, and y_i^* is the prediction of the class labels on the random permutation $X^*_{\mathcal{S}}$ on the data.

Logically, if an attribute is independent of the other attributes given a certain class, then such a permutation does not cause a drop in performance, which results in that particular attribute not being part of a group. This implies that attributes that do not belong to any group have a small impact on the model performance. In line with this, Henelius et al. (2014) show that the most important variables are usually included in one of the groups.

The just presented method is used in the analysis to gain global insights into the model mechanism of random forests. The next section discusses a method that provides insights into the relation between the explanatory variables and the response variable.

4.3 Relation between explanatory and response variables

A way to gain insights into how variables within a model are related, is by constructing partial dependence plots. These plots visualise an approximation of the underlying function $f(X)$, describing the relation between the prediction value and selected features in the model. Since these plots are limited to low-dimensional views, an approximation of $f(X)$ is visualised for a selected small subset of input variables. To produce a more comprehensive depiction of the function $f(X)$, a collection of lower dimensional plots can be made (Friedman et al., 2001).

The construction of these plots is as follows; let the vector of input predictor variables be $X' = (x_1, x_2, ..., x_j)$, consisting of a subset X_s and the complement set X_c, with $S \cup C = \{1, 2, ..., j\}$. Thus, the general function $f(X)$ will depend on all the input variables; $f(X) = f(X_s, X_c)$. The partial dependence of $f(X)$ on X_s is

$$f_s(X_s) = E_{X_s} f(X_s, X_c) \tag{21}$$

which is the marginal average of f, describing the effect of X_s on $f(X)$, by taking into account the average effects of the other variables X_c on $f(X)$. The partial dependence function is estimated by

$$\bar{f}_s(X_s) = \frac{1}{N} \sum_{i=1}^{N} f(X_s, x_{ic}) \tag{22}$$

where $\{x_{1c}, x_{2c}, ..., x_{nc}\}$ are the values of X_c occurring in the training data (Friedman et al., 2001). In case of two-dimensional partial dependence plots, the dependence of the response variables with one explanatory variable is constructed, resulting in X_s consisting of the single variable. The corresponding partial dependence is the average effect of the explanatory variable x_s on the outcome, while keeping all other variables constant.

After having identified the most important variables in the random forests and neural network models in Section 6.2, two-dimensional partial dependence plots are constructed for the dependence of $f(X)$ on each of these variables. The next section presents a method for local interpretation.

4.4 Local interpretation

A technique that approximates the original model on a local level is proposed by Ribeiro et al. (2016), who introduce LIME (Local Interpretable Model-agnostic Explanations); this method explains the predictions of black box techniques by approximating the function on a local point with an interpretable model. The idea is that each model can be approximated by a simple linear model on a local level. The formal definition of this technique is as follows; let an explanation be defined as a model $g \in G$, where G is a class of potentially interpretable models. Let $\Omega(g)$ be a measure of complexity of the explanation $g \in G$. The prediction of the relevant class of instance x_i is denoted by $f(x_i)$. Furthermore, $\pi_x(z_i)$ defines the proximity between an instance z_i to x_i. Finally, $\mathcal{L}(f, g, \pi_x)$ is the measure of how well g approximates f in the locality defined by π_x. Then the LIME explanation is defined as:

$$\xi(x_i) = \underset{g \in G}{\operatorname{argmin}} \ \mathcal{L}(f, g, \pi_x) + \Omega(g), \tag{23}$$

which tries to minimize $\mathcal{L}(f, g, \pi_x)$, while having $\Omega(g)$ low enough to still be interpretable. In this way, both interpretability and local fit are ensured. The resulting explanation provides insights into the contribution of each variable on the predicted outcome for each instance separately. This method can explain the predictions of any classifier, since different G, $\mathcal{L}$ and Ω may be used for different classifiers.

The LIME algorithm is model-agnostic, i.e., is able to handle any type of classifier. Therefore, it is applied for random forests as well as neural networks in the analysis of this thesis. The next section presents methods for graphical depiction of how black box methods work.

4.5 Graphical representation

To gain global understanding into how a black box model works, the model can be approximated by a single decision tree. This provides a comprehensible explanation because of the graphical representation of the tree (Zhou & Hooker, 2016). There exist several different algorithms for inducing a decision tree, among which the earlier discussed trees based on CART methodology and conditional inference trees. To approximate the model, the tree inducting algorithm is applied to the data where the output is changed to the predicted value (Martens et al., 2007). To improve the understandability of the tree, it can be cut to a smaller depth (Gibbons et al., 2013).

Another tool for getting a global idea of a neural network is a Neural Interpretation Diagram, which allows for a visual interpretation of the network, where the importance of the connection weights are depicted by the line thickness and colour. Furthermore, interactions between variables become clear from the joint neurons (Özesmi & Özesmi, 1999). However, the complexity of the connections between the neurons cause that the diagram, especially in the case of a data set with many variables, loses its easy interpretation. Since the data used in this thesis consist of many variables, leading to a neural network with several hidden neurons, no Neural Interpretation Diagram is derived.

Instead, single tree approximation is performed on random forests and neural networks, allowing for an approximating graphical depiction of the model, by inducing conditional inference trees. To assess the accuracy of the approximation, fidelity is used, determining to which extent the predictions of the approximation correspond to the outcome of the black box model. The next section describes the data used for the analysis of this thesis.

5 Data description and preparation

The dataset used in this thesis is acquired from Fannie Mae, one of the leading mortgage financiers of the USA. The available data provide information on the performance of single-family mortgage loans and includes information on around 22 million loans. The data range between the years 2000 to 2016 and are available per quarter. The loan performance data are divided into two parts: acquisition and performance. Acquisition includes static data at the time of a mortgage loan's origination. It contains information about the borrower such as a debt-to-income ratio, a borrower credit score, number of borrowers and first time home buyers indicators. This furthermore includes fixed loan characteristics such as original loan to value and original interest rate. Performance contains the monthly performance data of each mortgage loan, which includes dynamic information on the unpaid principle balance, interest and delinquency status. The acquisition and according performance files are combined for analysis.

The objective of the analysis in this thesis is to predict which loans will default within the next 12 months. In this thesis, default is defined as a loan being delinquent, i.e., not having met the financial obligations, for more than 90 days. To predict default, a variable is constructed to indicate that a loan defaults within next 12 months (default.12months), which serves as the quantity of interest in this thesis. We restrict the data to consist of loans that have observations for at least two years and transform all categorical variables into dummies.

Not all available variables in the Fannie Mae data set are used in the analysis, since not all provide relevant information for default prediction. We decide to follow the current literature on mortgage default (see, e.g., Fitzpatrick & Mues, 2016) and consider the following variables. Firstly, information on the borrower is used, which includes the debt-to-income ratio, credit score, a first time home buyer indicator and number of borrowers of the loan. Furthermore, static data on the loan are included, such as the loan purpose, occupancy status, loan-to-value ratio, loan size, the original interest rate, original loan term and original house price. Lastly, some monthly data are included such as the loan age, the actual months to maturity, the current unpaid principle balance and the delinquency status. A more elaborate description of all variables used in the analysis can be found in Table 1 on page 20.

Since we want to assess whether the bias in variable importance measures forms a problem in our empirical example, it is important to include both continuous and correlated variables. Figure 5 visualises the pairwise correlations of all the explanatory variables. From the graph it becomes clear that the categories of the loan purpose and the categories of the loan occupancy status show high pairwise correlations. Besides, the original interest rate, loan size, loan term, house price and unpaid principle balance are highly correlated. This outcome is in line with expectation, since some variables are closely related. For example, it is to be expected that the original loan size is correlated with the house price and the current unpaid principle balance.

Given the computational requirements of some of the used techniques in this thesis, we

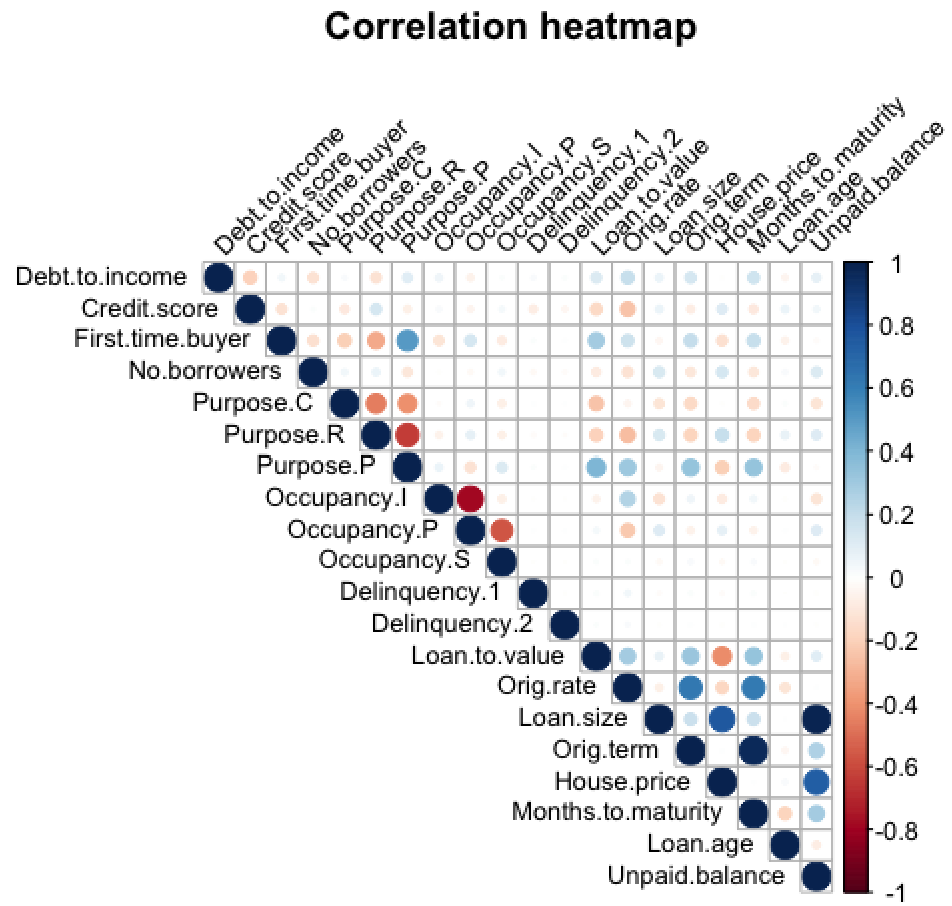


Figure 5: Heatmap of correlations of variables in Fannie Mae data set

Table 1: Data dictionary

Variable	Description	Values
default.12months	Indicator that default (=being delinquent for more than 90 days) will happen within 12 months.	Y (yes) / N (no)
Debt.to.income	Debt-to-Income ratio, calculated at origination by dividing the borrower's total monthly obligations by her/her stable monthly income.	1-64
Credit.score	Minimum of borrower and co-borrower credit score, both referring to the FICO score.	402-855
First.time.buyer	First time home buyer indicator.	Y (yes)/ N (no)
No.borrowers	Number of borrowers obligated to repay the mortgage.	1-10

Purpose	An indicator denoting if a mortgage loan is either a purchase money or refinance mortgage.	P=purchase,
		C=cash-out,
		R=no cash-out refinance.

Occupancy	Indicator denoting how the borrower used the mortgaged property at the origination date of the mortgage	P=principal residence,
		S=second home,
		I=investment.

Loan.to.value	Original combined loan-to-value. Ratio of the original loan amount and the the value of the mortgaged property.	1-112
Orig.rate	Original interest rate on a mortgage loan.	1.875-6.75
Loan.size	Original amount of the mortgage loan.	8,000-1,203,000
Orig.term	The original number of months in which borrower payments are due.	60-360
House.price	House price, calculated by dividing original amount of the morgage loan by the original loan-to-value.	10,000-25,200,000
Loan.age	Number of calendar months since the first full month the mortgage loan accrues interest.	11-73
Month.to.maturity	Number of calendar months remaining until the mortgage is payed in full.	0-351
Delinquency	The current number of days, represented in months, the obligor is delinquent, 3 = default.	0=<30 days, 1=30-59 days, 2=60-89 days, 3=90-119 days.
Unpaid.balance	Sum of current unpaid non interest bearing and principle forgiveness unpaid principle balance.	941 - 1,181,393

restrict the data to consist of only the 5 most recent available years, i.e., loans operating in the years 2012 to 2016. Since it is the objective of this thesis to explain the predictions of a model rather than to build a model with the highest predictive power, this restriction should not form an issue in our analysis. To avoid correlation between the entries of the same loan, we randomly sample one entry per loan. All loan entries currently in default are excluded, since the objective is to predict which non-default loans will default in the future.

In order to assess prediction accuracy, the resulting data, consisting of a total of over 14 million loans, are divided into a training set consisting of 70% of all loans and a hold-out set of 30%. The variable of interest, whether a loan will default within 12 months or not, is as expected unbalanced across the data: in the training set, only 23,295 out of 10,290,027 (0.22%) instances default within 12 months. This forms a problem in prediction problems, since the minority class may not be recognized properly and all loans are likely to be classified as non-default. In order to adjust for the unbalance in the data, the training data are balanced. There exist different methods for balancing the data, such as oversampling, under-sampling and Synthetic Minority Oversampling Technique (SMOTE; Chawla, Bowyer, Hall, & Kegelmeyer, 2002). The latter is used in this analysis.

SMOTE is a technique that combines under-sampling of the majority class with a special form of oversampling the minority class, which has been shown to improve the ability to recognize the minority class (Chawla et al., 2002). With regular oversampling the minority class cases are sampled with replacement from the original data. In contrast to this, SMOTE creates “synthetic” examples by using k nearest neighbours. It creates additional minority instances which are similar to the original instances. By applying this approach it is possible to increase both the minority class size and the minority class recognition. The minority class instances in the training data, i.e., the entries of loans in default, are over-sampled at 100%, which creates 23,295 additional instances defaulting within 12 month. From the loans that are not in default, we randomly sample 46,590 loans. This results in an appropriately balanced training set of 93,180 loan entries.

6 Results empirical analysis

This section starts by outlining the results of logistic regression, random forests and neural networks. After the initial comparison of both accuracy and interpretability of these techniques, the results of explanations methods are presented, where the explanations are assessed in terms of interpretability. Finally, an overview of the comparison of the presented explanation methods is given.

Table 2: Comparison of accuracy vs. interpretability

	Accuracy (%)		Interpretation	
model	accuracy score	recall	global	local
logistic regression	84.9	79.8	✓	✓
random forests	87.3	76.8	-	-
artificial neural network	82.7	82.2	-	-

6.1 Initial comparison

Table 2 compares the results of logistic regression, random forests and neural networks for default prediction on our data, in terms of accuracy and interpretation. All methods are performed on the balanced (SMOTEd) training set and the accuracy is measured on the non-balanced hold-out set. In case of neural networks, the independent variables are scaled. More specifically, the training data are scaled in a regular manner using the mean and standard deviation of each of its columns. The test data are scaled using the means and standard deviation of each corresponding column of the training data.

Table 3: Results logistic regression

Independent variable: *default within 12 months*				
	Estimate	Std. Error	z value	Pr(> |z|)
(Intercept)	7.9375	0.2190	36.25	0.0000
Debt.to.income	0.0331	0.0011	28.79	0.0000
Credit.score	-0.0179	0.0002	-83.29	0.0000
First.time.buyer	-0.1068	0.0299	-3.57	0.0004
No.borrowers	-1.0226	0.0199	-51.46	0.0000
Purpose.C	0.4697	0.0300	15.68	0.0000
Purpose.R	0.0849	0.0274	3.10	0.0020
Occupancy.I	-0.2546	0.0631	-4.04	0.0001
Occupancy.P	0.3723	0.0535	6.96	0.0000
Delinquency.1	4.1591	0.0812	51.25	0.0000
Delinquency.2	6.4785	0.2833	22.87	0.0000
Loan.to.value	0.0242	0.0011	22.18	0.0000
Orig.rate	0.6838	0.0231	29.58	0.0000
Loan.size	-0.0000	0.0000	-3.67	0.0002
Orig.term	-0.0158	0.0011	-13.85	0.0000
House.price	-0.0000	0.0000	-3.15	0.0016
Months.to.maturity	0.0157	0.0012	13.35	0.0000
Loan.age	0.0347	0.0014	24.57	0.0000
Unpaid.balance	0.0000	0.0000	2.94	0.0033

We see that random forests performs best in terms of the accuracy score, whereas artificial neural networks show the highest recall. It therefore depends on the objective of the study which method is preferred; in case the identification of defaulting loans is of main interest, which often is the case for mortgage insurers, neural networks is preferred, whereas if overall accuracy is the main priority, a random forest gives the best results.

The results of logistic regression are presented in Table 3. All variables are significant and the coefficients of the explanatory variables seem to have the expected sign. The debt-to-income ratio, the cash-out and refinance purpose of the loan, occupancy as a principle residence, being delinquent for more than 30 days, loan-to-value ratio, interest rate, amount of months to maturity, loan age and current unpaid principle balance are positively related to the probability of default within 12 months. On the other hand, the credit score, indicator of a first time home buyer, the amount of borrowers, occupancy as an investment, loan size, loan term and house price are negatively related to the probability of default within 12 months.

Concerning the interpretation, logistic regression performs well in terms of global and local interpretability. Similarly to a linear regression, the coefficients of the logistic regression provide insights into the effect of each explanatory variable on the outcome. Also, the effect of changes in the input on the output can be detected and the default prediction can be explained for each instance separately. The difference with a linear model is that the coefficient corresponding to a covariate indicates the change in the log-odds of the response variable, instead of the change in the value of the response variable.

Both random forests as well as neural networks suffer from the disadvantage that the output of these techniques cannot be directly interpreted. Therefore, the following sections present the results of methods that provide more insights into random forests and neural networks.

6.2 Variable importance

This section presents several variable importance measures, starting with the measures concerning random forests. Firstly, the variable importance measures based on the Gini criterion and permutation from the regular *randomForest* R-package are compared with the variable importance based on a conditional inference tree (using the R-package *cforest*). Secondly, the random forest using conditional variable importance is assessed in terms of interpretability.

The results of the variable importance measures are presented in Figure 6. We see that each measures varies in its scale and therefore we only look at the relative size and the order of the variable importance scores in the discussion of the results. The three variable importance measures agree on the most important variables, namely someone's credit score, both delinquency statuses and the original interest rate. Also, the least important variables

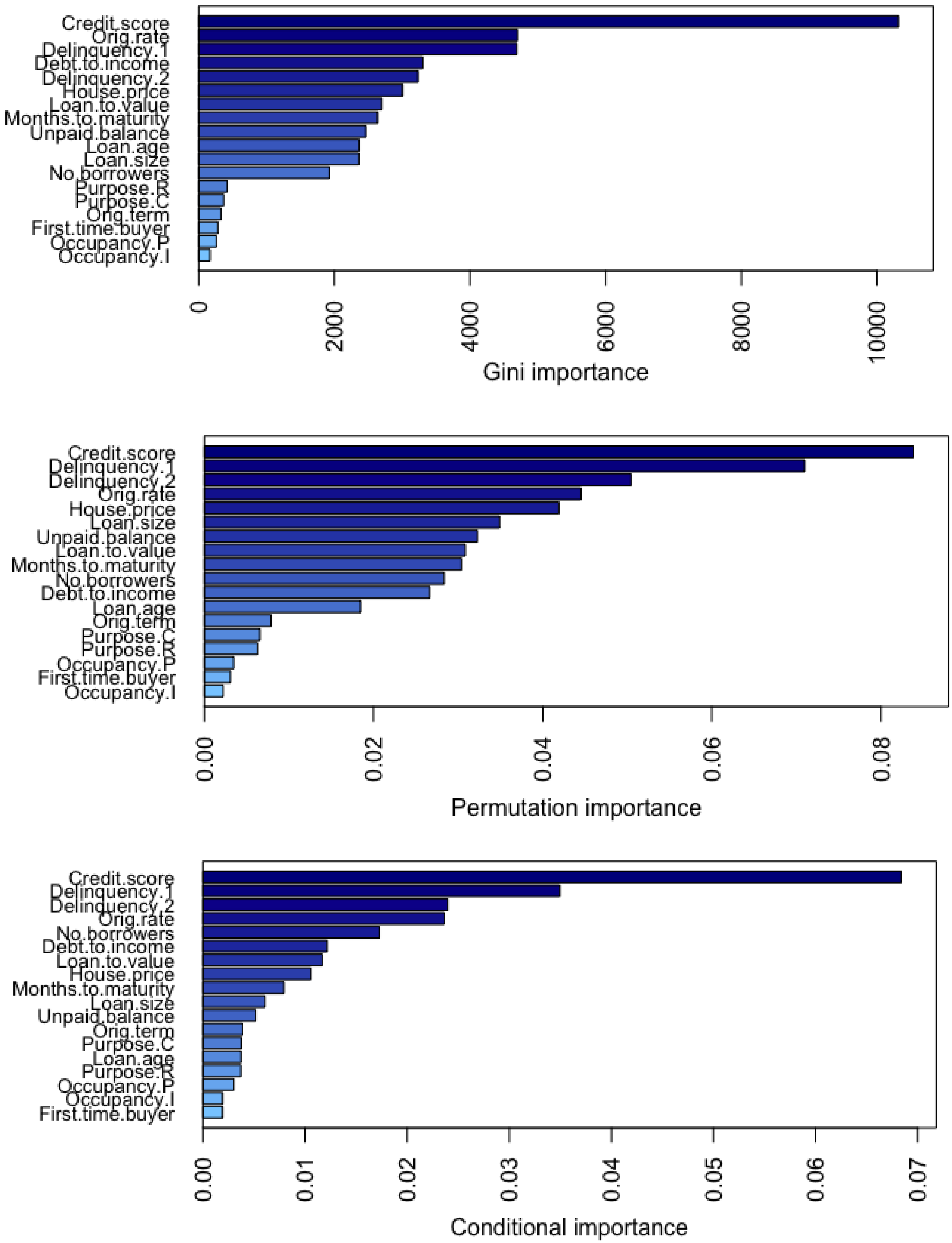


Figure 6: Variable importance measures: Gini, permutation and conditional importance

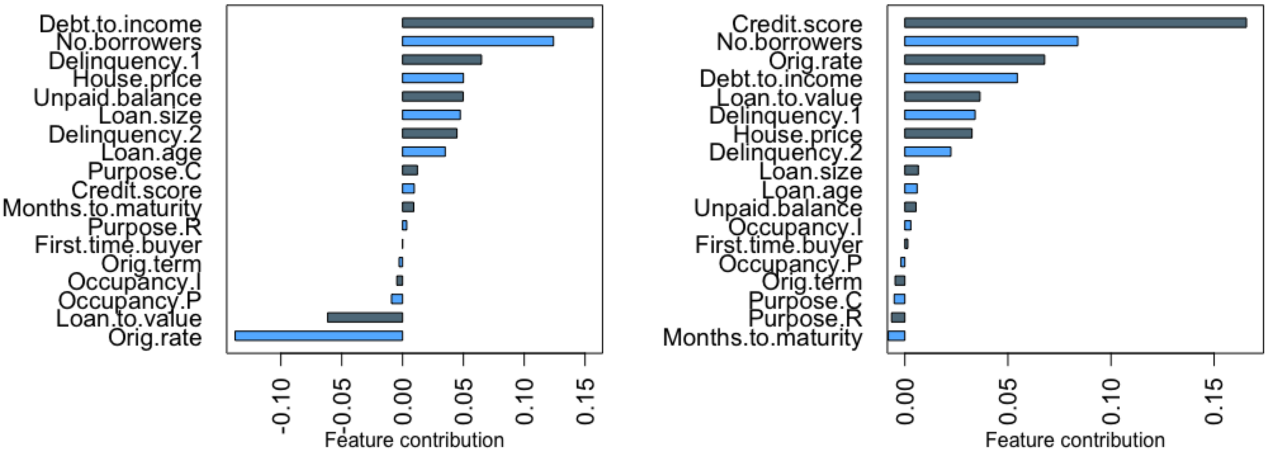


Figure 7: Feature contributions for two selected instances

seem to be the purpose of a loan, the original loan term, whether someone is a first-time home buyer and the occupancy of the mortgage property.

The Gini importance, displayed in Figure 6a seems to display a bias against the continuous variables. This can be seen by comparing the relative size of the Gini importance with the relative size of the conditional importance. The continuous variables are original interest rate, debt-to-income ratio, house price, months left to maturity, loan-to-value ratio, loan age, unpaid principle balance and loan size. Their importance is overestimated with using the Gini importance, compared to the conditional importance.

The permutation importance, displayed in Figure 6b seems to be biased against the correlated variables. The correlated variables in the used data set are original interest rate, house price, months to maturity, loan size, unpaid balance and loan age. We see that all of these variables have a high importance according to the permutation importance (subfigure 6b.), whereas the contribution of these variables seems to be lower with the conditional importance.

Since the Gini and permutation importance bias is present in the application of random forests on our data, we restrict to assessing the interpretability of the conditional importance. The conditional importance provides global interpretability by giving insight into the magnitude of the effect of each variable on the decrease of accuracy of the model. By visualising the results, the interpretation becomes straightforward. However, it does not become clear which relationship the variables have with the predicted outcome nor why the inclusion of some variables leads towards a particular prediction. We now turn to the discussion of feature contributions, which provide a remedy to some of these limitations.

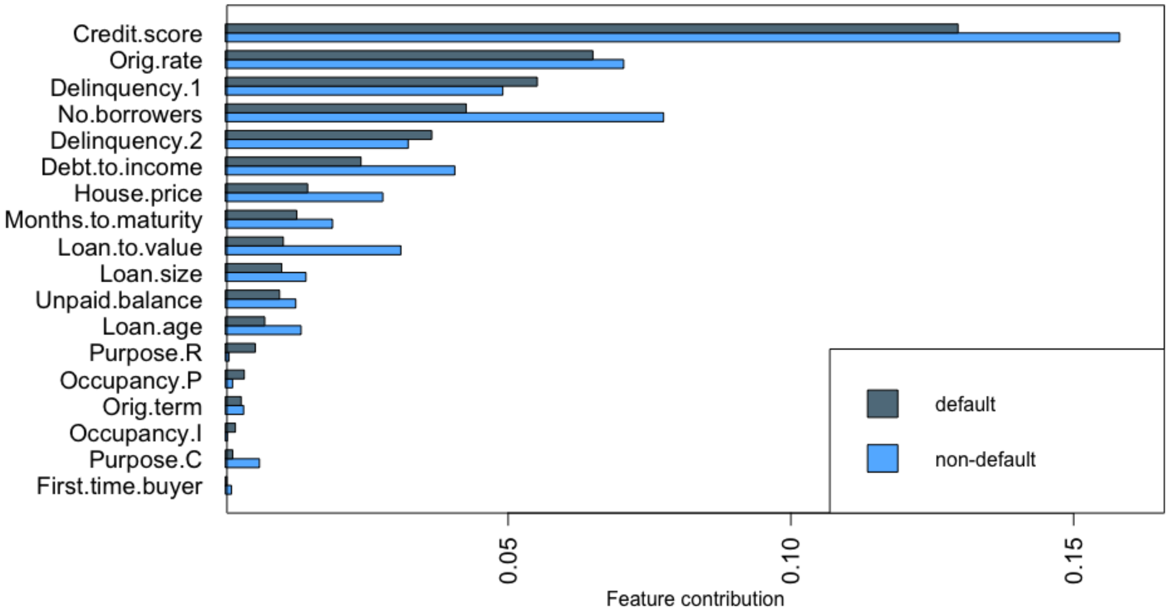


Figure 8: Median of feature contributions per variable for each class

Feature contributions are constructed for each instance in the random forests model, see Figure 7 for the results of two selected instances, which are classified as non-default. We see the importance of each variable on the predicted class, which enables local interpretation. For the first instance debt-to-income, number of borrowers and delinquency status 1 are the most important variables that positively contribute to the class label, whereas the original rate and loan-to-value contribute negatively to the class label. For the second instance the results are different and almost no negatively contributing variables are present.

To analyse feature contributions on a global scale, the approach of Palczewska et al. (2014) is followed, by constructing the median of the contributions. The result is represented in Figure 8, where the median contribution of each variable is presented per class. The presented median feature contributions can be seen as the 'standard' level for representatives of a particular class. This means that the feature contributions of instances from the training set belonging to a particular class will in most cases have contributions close to the presented median values. In general, the four most important variables in determining default are someone's credit score, the original interest rate of a loan, if someone has a delinquency status and the number of borrowers. Also, the least important variables are the purpose and the occupancy of the loan and the indicator of a first home buyer. This is in agreement with the conditional variable importance measures presented in Section 6.2.

A difference with variable importance is that feature contributions can be constructed

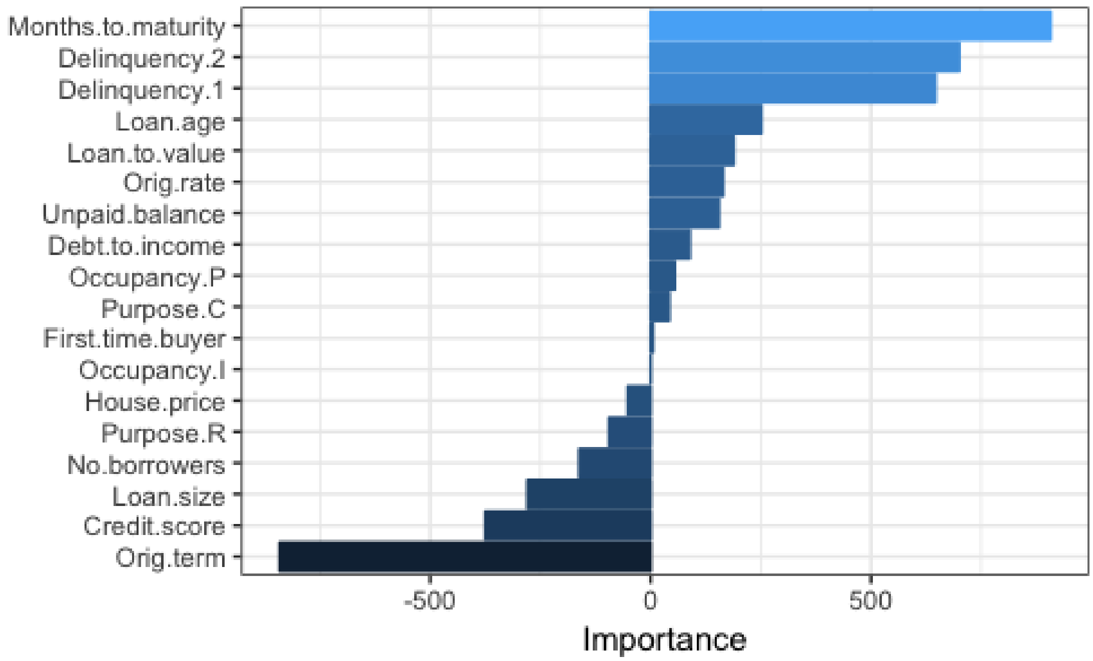


Figure 9: Olden's variable importance in default prediction by neural networks

per instance and per class. This demonstrates that, for the first instance in Figure 7 the credit score is not of large influence on the class label, whereas 'on average' it is the most important feature, according to Figure 8. This makes it possible to explain the prediction of each instance separately, providing detailed information on each case. Concerning global interpretation, we see in Figure 8 that the credit score and number of borrowers are more important in the prediction of loans that do not default, than for loans that are predicted to default within 12 months. On the other hand, a delinquency status contributes more to the prediction of defaulting loans than the loans that do not default.

In terms of interpretability, feature contributions score similarly to variable importance in terms of global interpretability. Again, it is possible to globally get insights into the magnitude of the contribution of each variable on the outcome. The addition to variable importance is that the contributions can be constructed per instance, making local interpretation possible, which also provides insights into the magnitude of the contribution of each feature.

Concerning neural networks, a variable importance measure is created by following the methodology of Olden and Jackson (2002). In Figure 9 we see that the most important variables positively influencing the probability of default are months to maturity and delinquency status. These results are in line with expectation; if the time to maturity is short, the chance of default is small. Furthermore, if a person is delinquent of at least 30 or 60 days, this is a

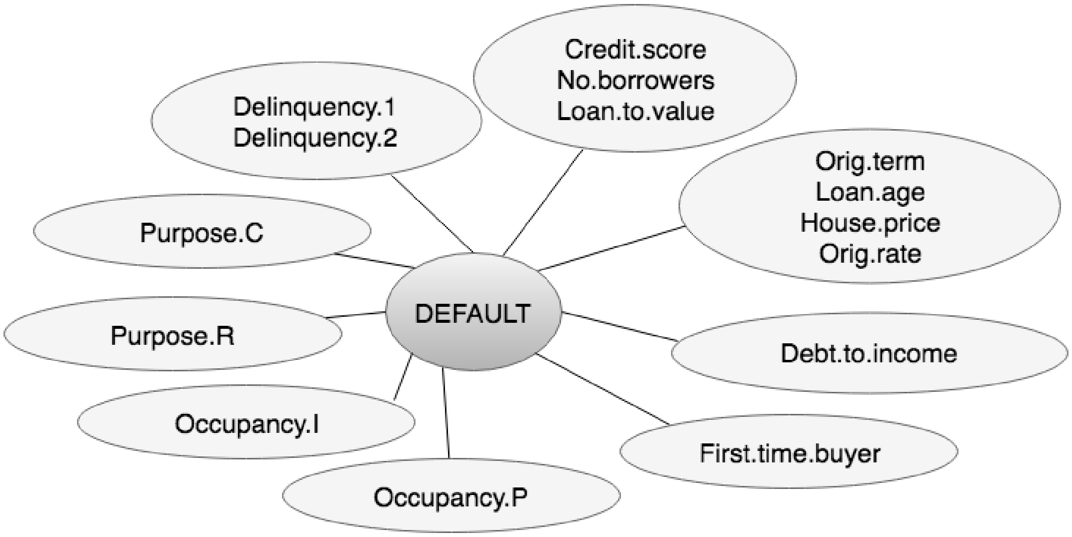


Figure 10: Groupings of variables in the random forests model

strong indicator for future default. The variables having a negative effect on the probability of default are the original term of the loan, a person's credit score and the original loan size. Also these results seem logical and coincide with the previously found results for logistic regression (see Table 3) and variable importance measures of random forests (see Figure 6).

In terms of interpretability, the variable importance for neural networks provides us with a tool for global interpretability. It gives us insight into which variables are important in determining the outcome of the model, and by which magnitude and sign. The next section presents the interactions that are present in the random forests model.

6.3 Variable interactions

In this section the structure of the random forest function is revealed by determining the interactions between variables that are considered in the classification. Groupings of variables are found by applying the randomization approach of Henelius et al. (2014) and Henelius, Ukkonen, and Puolamäki (2016). The balanced train set was used to train the random forest classifier and to determine the groupings of the attributes. The results are visualised in Figure 10. It can be seen that there exists three groupings of variables in the data consisting of two to four variables. The remainder of the variables does not show any interactions and each of these variables is part of their own singleton group.

The first grouping includes both delinquency statuses, which are both strong indicators of the borrower on future default. This grouping implies that these variables are used together in the default prediction by the random forests model. The second grouping consists of the credit score, the number of borrowers and loan-to-value ratio, of which the former two are characteristics of the borrower and the latter provides information on the loan. These three variables could be seen as indicators of financial strength of the borrower and show interaction in the random forests model, which is incorporated in training the model. The third grouping consists of the loan term, loan age, interest rate and the house price. This grouping contains mostly information on the loan and could therefore be interpreted as deciding loan characteristics that, when used together, are important in determining default within 12 months. The results show that most important variables, which are identified by the conditional importance in Figure 6c, are included in the groupings, which is in line with the findings of Henelius et al. (2014).

The identified interactions provide us information on the associations between the attributes. The groupings indicate that not all attributes in the data set are independent. Furthermore, it becomes clear which groupings the random forests model exploits in building the classifier. However, it remains unclear why certain attributes are categorized in the same group. This can either be due to interactive effects, which means that the individual attributes do not contain information on the class label on their own, or due to additive effects, which entails that the individual attribute should be combined with the attributes from its grouping to present complete information on the class label.

The interactions provide good means for understanding the random forests model on a global level. We see which interactions are used to form the random forests predictions. However, no insights are given into the effect of explanatory variables on the outcome. To address this issue, the next section discusses how the most important features of random forests and neural networks depend on the outcome by plotting partial dependence plots.

6.4 Relation between explanatory and response variables

Figure 11 presents the relation of the eight most important variables, as identified in Section 6.2, with the probability of default in our random forest, neural network or both. We see that credit score and number of borrowers are negatively related to default, whereas the delinquency statuses, the debt-to-income ratio and original interest rate show a positive relationship with default. These findings are in line with the earlier described variable importance measures and coincide with the results from the logistic model.

The two most important variables in the neural network, i.e., months to maturity and original term of the loan, have a positive and negative effect on default, respectively. The random forest classifier, however, seems not affected by the months to maturity until it reaches higher values and does not show any relation between default and the loan term.

From the figure it becomes clear that the neural network classifier was only able to detect linear relationships between the presented features and the probability of default, whereas

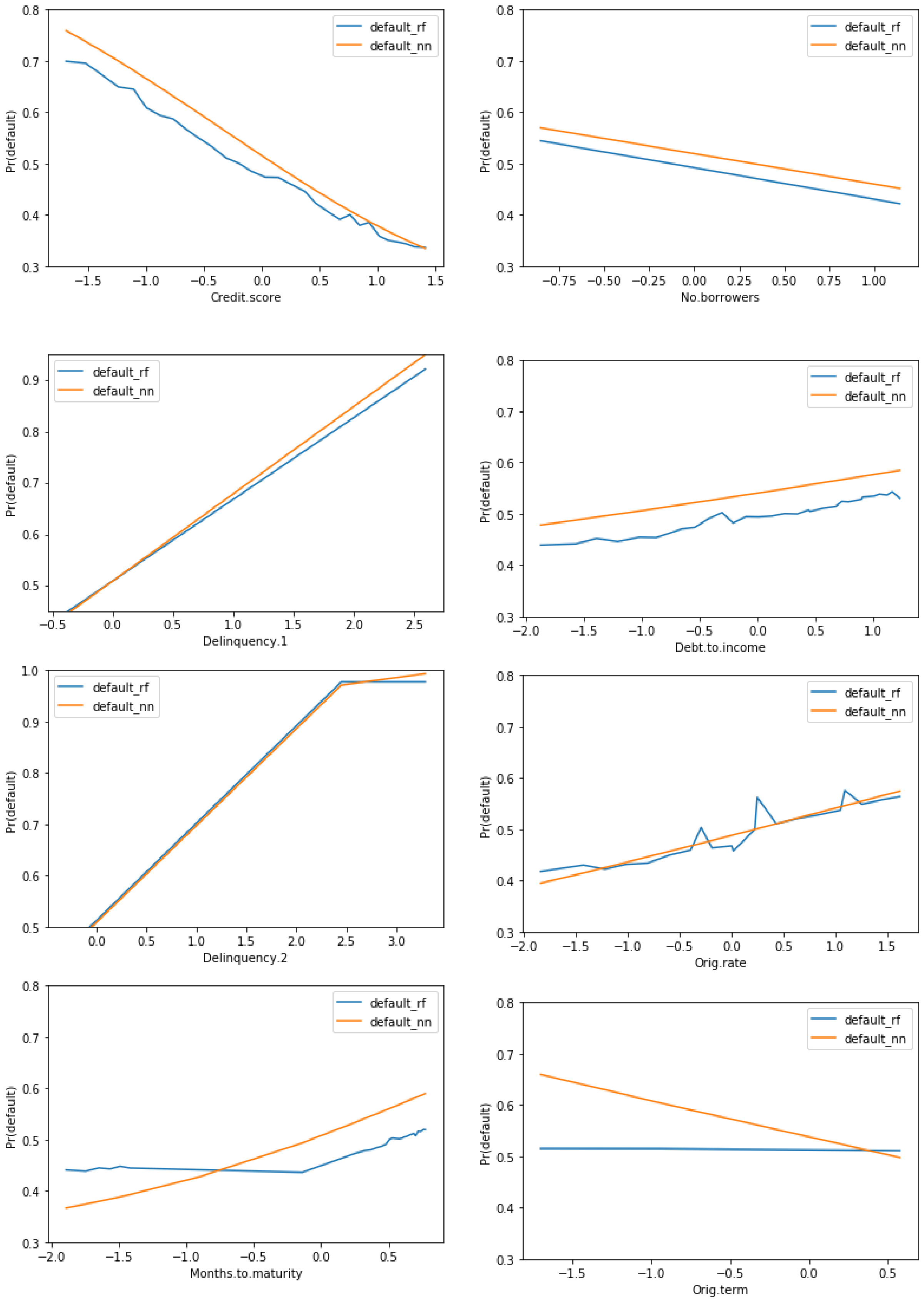


Figure 11: Partial dependence of the probability of default on the most important variables random forests and neural networks

the random forest classifier shows some non-linearities in the relation between default and the original interest rate and the amount of months left to maturity. This could provide us with an explanation on why random forests and neural networks did not substantially outperform the simple logistic regression. In terms of interpretability, the partial dependence plots provide us with a tool for global interpretation. The graphs present which relation the features have with default. We are therefore given information on the sign on the variable contributions. To assess more in depth why the inclusion of some variables leads to a particular prediction, the next section discusses how the model function can be locally approximated, making it possible to explain the prediction of every instance.

6.5 Local interpretation

Here, we present the results from the application of the LIME algorithm. We use the previously presented random forest model, trained on the balanced sample. Predictions are made on the non-balanced hold-out sample, which are explained by the LIME algorithm at a local level. By doing so, the prediction of each individual instance can be explained. As an illustration, the results of 4 non-default and 4 default instances are illustrated in Figures 12 and 13, respectively.

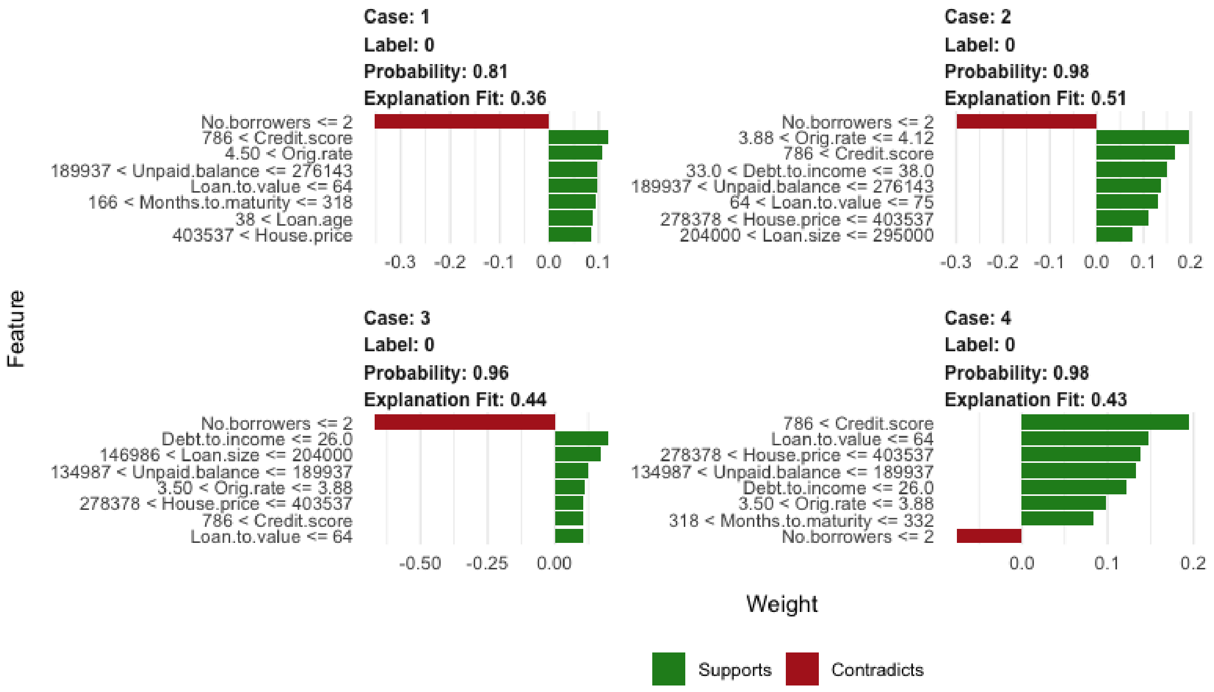


Figure 12: LIME for loans classified as non-default by random forests

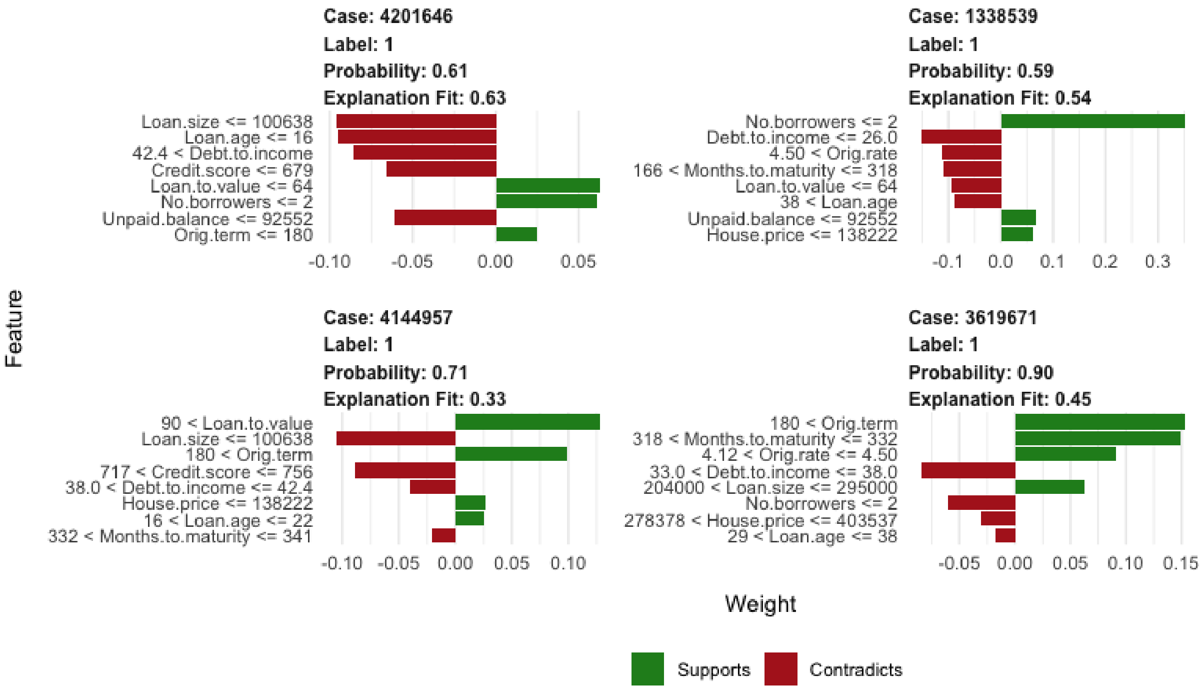


Figure 13: LIME for loans classified as default by random forests

For each instance, the eight most important features are selected. For these features it is determined how strongly they support or contradict the statement on a variable. The label 1 indicates default, whereas 0 indicates non-default. Furthermore, the probability indicates how likely the observation belongs to the assigned class and the explanation fit is a measure of how close the data are to the fitted linear regression line.

We start by outlining the results for loans classified as non-default. In Figure 12 we see that for the first instance the number of borrowers negatively influences the class label with around 0.35, whereas all the other variables listed have a positive effect of around 0.1 on the probability of non-default, which is 0.81. This means that, for instance, if the credit score would not be higher than 786 (as is the case for this instance), the probability of non-default would decline with around 0.12, and if the original interest rate would not be higher than 4.55 (as holds for this instance), the probability would decline with around 0.11. A similar way of interpreting of the other instances can be performed. All presented instances have a high probability of belonging to the non-default class and the local linear fit seems to indicate that these explanations are to be trusted, as pointed out by Ribeiro et al. (2016).

Besides inspecting each instance separately, we can also try to find some patterns in the presented examples. For instance, if the number of borrowers is more than 2, this could support default within 12 months; we see that all instances, having less or equal to 2

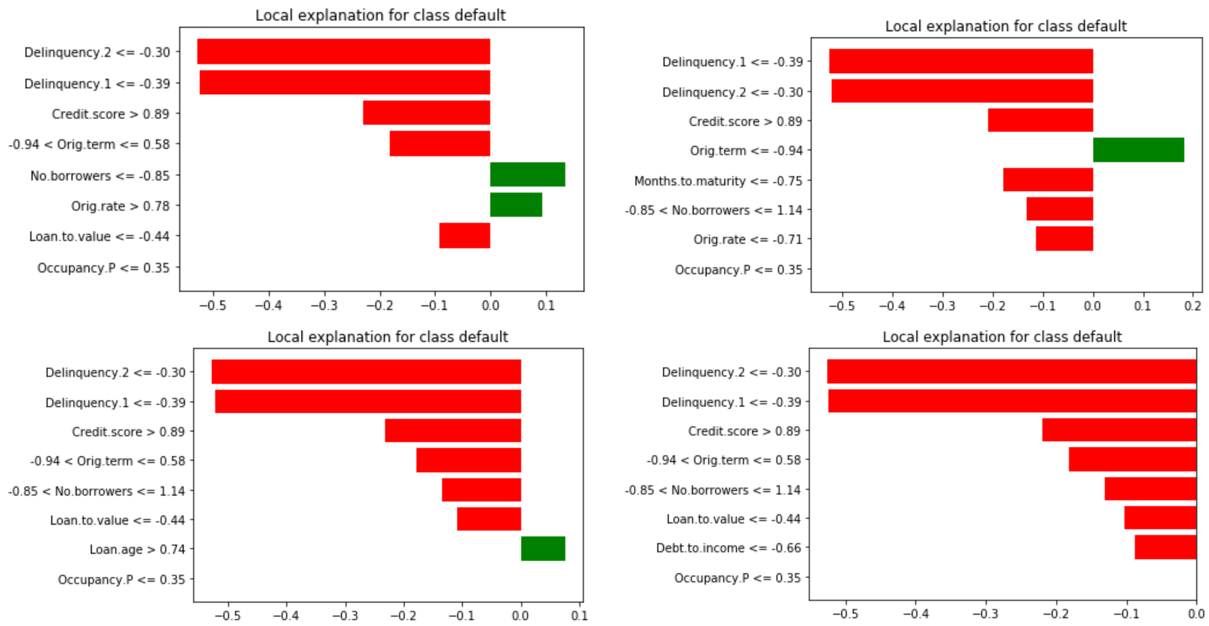


Figure 14: LIME for loans classified as non-default by neural networks

borrowers, contradict the label 0. Besides, a credit score of more than 786 (all instances) could be an indicator for non-defaulting loans, as well as a low loan-to-value score (less than 75 - all instances) and low debt-to-income ratio (less than 38 - instance 2, less than 26 - instances 3 and 4). However, it should be noted that this interpretation cannot be interpreted globally, since only four instances are considered to find the patterns. For a global interpretatation, more instances should be considered. This, in turn, makes interpretation more cumbersome.

Overall, the most important features coincide with the results of conditional variable importance and feature contributions. It can be seen that the credit score, original interest rate and number of borrowers are important in the prediction of default. Also, the least important variables coincide, since in the presented eight most important variables the variables on the loan purpose and occupancy and first time home buyer are lacking.

Some results corresponding to the logistic regression presented in Table 3 can also be found. Firstly, the relationship with the number of borrowers and default is negative; once the number of borrowers is less than 2, this contradicts the probability of non-default, meaning supporting the probability of default. Secondly, credit score is negatively related to default, since a high credit score (i.e., more than 786) shows a lower probability of default. Lastly, the loan-to-value ratio has a positive relation with default, since low loan-to-value ratio's support the non-default class label, which implies that a high ratio means a higher probability of default.

We now turn to the discussion of the loans classified as default. In Figure 13 we see that the fourth case has the highest probability of 0.90; this can be explained by an original loan term of more than 180, between 318 and 332 months to maturity, an interest rate of between 4.12 and 4.5 and a loan size between 204,000 and 295,000 dollars. The results provide us with detailed explanations on the prediction of each instance, however it is harder to discover a clear pattern for defaulting loans across the presented instances. The instances differ concerning the importance of the features. Furthermore, some features are within different ranges, which complicates the comparison of results.

Figure 14 presents the local approximation of four instances, which are classified as non-default loans by the neural network. The explanations are executed for the default class, therefore almost all features contradict the value label. The interpretation is similar to that of the LIME explanations of random forests. From the graphs we see that the most important variables are the delinquency statuses, the credit score, the original term and the number of borrowers. This coincides with our previous findings.

Concerning interpretability, the results of the LIME algorithm provide a tool for local explanation. The result gives insights into both the magnitude and sign of the effect of explanatory variables on the probability of default fore each instance. However, since some of the features are within different ranges, interpretation and comparison with other instances remains hard. Also, no inference can be made on a global level, which is resolved in the next section, where an approximation of the model is given by building a single tree.

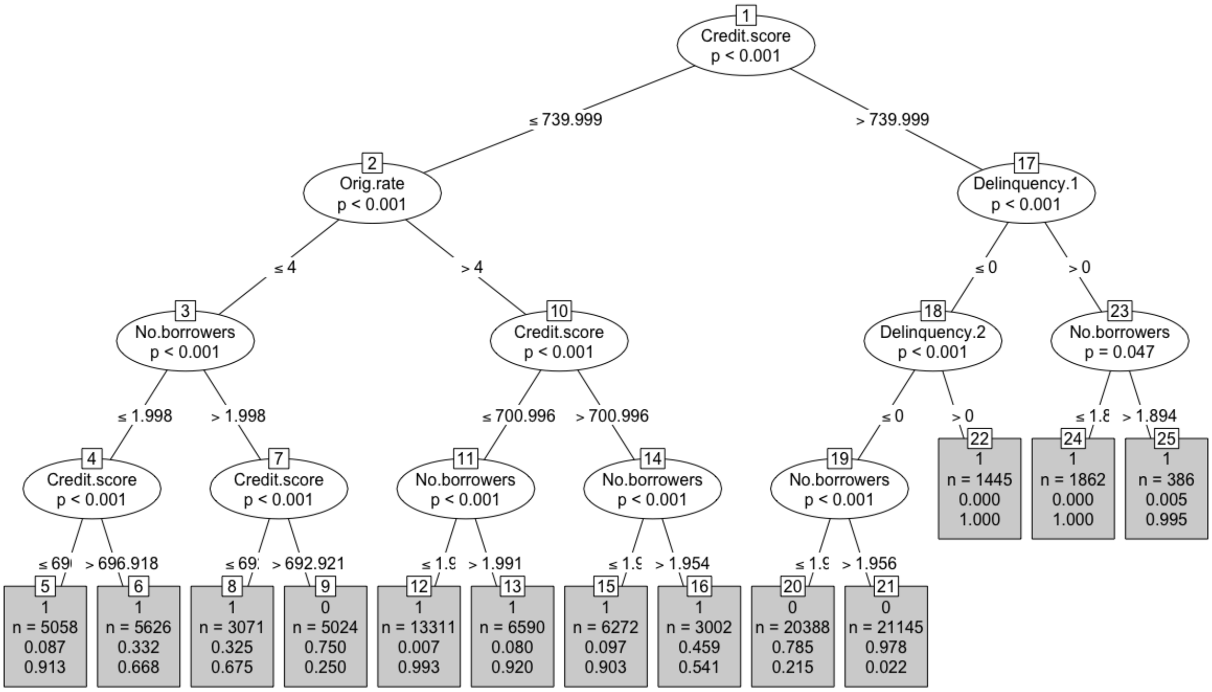


Figure 15: Single tree approximation of random forests

6.6 Graphical representation

To give global interpretation to the random forests model, a single tree approximation is performed, in line with the work of Martens et al. (2007). We decide to build a conditional inference tree, as the conditional variable importance based on this tree does not show the bias present in the Gini or permutation measures. The tree is cut to a tree depth of 4. While this enhances the comprehensibility, it provides us with less detailed information and less accurate predictions than a larger tree.

The accuracy of the constructed decision trees is measured by the fidelity, which is the percentage of the predictions of the tree that coincide with the prediction of the black box model. For random forests the fidelity score of the decision tree is 87.1% and for neural networks fidelity is equal to 89.2%.

From the tree in Figure 15 we see that the most important variables in the random forests approximation are the credit score, the original rate, the number of borrowers and the delinquency statuses. As can be expected, this coincides with the variables identified by conditional variable importance. However, this tree also provides us with information on the splits; we see at which split points of each variable the tree is divided, resulting in the prediction of the classes across the instances.

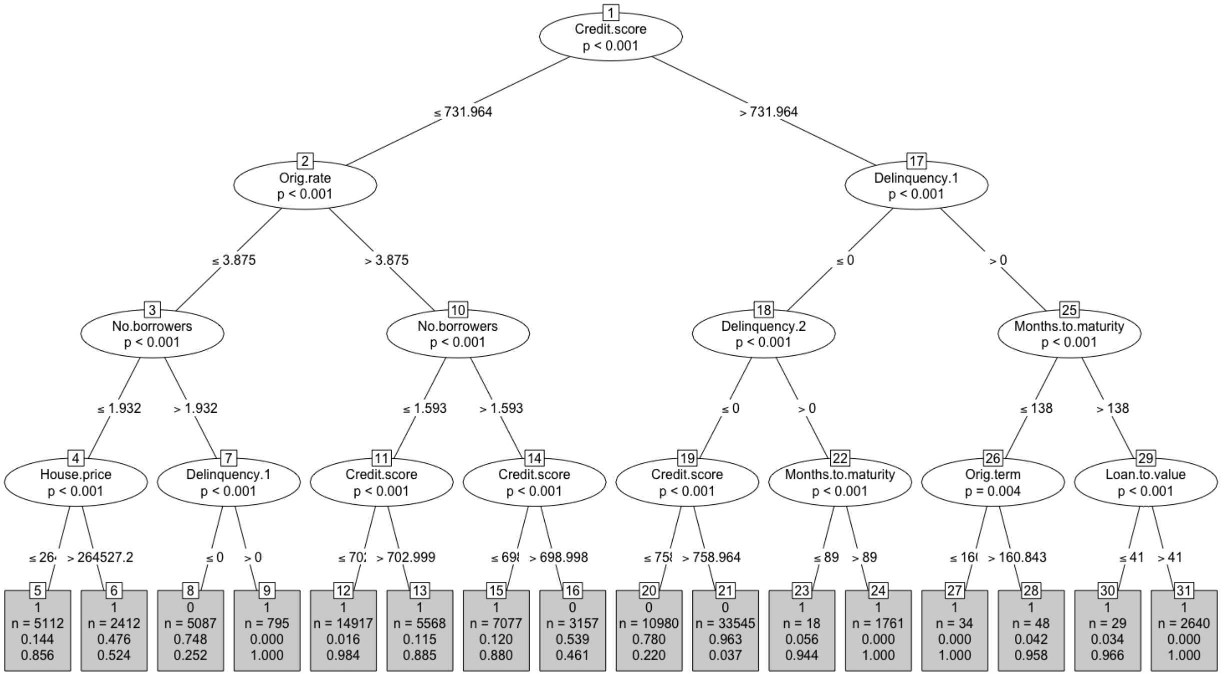


Figure 16: Single tree approximation of neural networks

In addition, insights are given on the sign of the effect of each feature on the outcome. For instance, a high credit score (more than 740) leads to a high amount of instances classified as non-default (node 20 and 21), whereas a lower credit score (less than 740) leads to all instances classified as default. This can be determined for each of the attributes displayed in the tree. Some attributes are listed double in the tree (see, e.g., credit score in node 4, 7, 10), which creates intervals of values of these attributes, providing an even more specific interpretation.

Figure 16 presents the conditional inference tree built on the predictions of the neural network. Credit score, the original rate, delinquency statuses, number of borrowers, months to maturity and loan-to-value seem to be the most important variables, coinciding with the results of the variable importance measure of neural networks. Again, insights into the relation between an explanatory variable and the outcome can be derived by following different paths in the tree.

In terms of interpretability, this decision trees provide global interpretation into the models. The graphical depiction helps in understanding how the models work and how decisions are made. The most important features can be identified by the hierarchical structure of the tree. Also, local patterns can be detected by following each path from the root to the leaf node, which also provides insights into the sign of the effect of each variable on the outcome.

6.7 Overview of results

This section provides an overview of the results. In Table 4 the discussed explanation techniques for random forests and neural networks are presented.

Variable importance of random forests provides global interpretation on the relative importance of each feature on the outcome. Feature contributions can be constructed per instance, which provides a tool for local interpretation and gives insights into the magnitude and sign of the contribution of each feature. Furthermore, by constructing the median of all feature contributions, global insights can be gained, which provide the relative magnitude of each variable contribution. The variable importance measure for neural networks provides global insights into which variables are important in determining the model outcome, and by which magnitude and sign. Variable interactions demonstrate which interactions are used to form the random forests predictions, which provides global understanding of how the model works.

Partial dependence plots graphically represent the relationship between the explanatory and response variable, which indicates the sign the variable contribution and provide interpretation on a global scale. Local interpretation allows for detailed analysis of the magnitude and sign of each variable contribution on the predicted class label on a local scale. Lastly,

classifier*	explanation	global	local	magnitude variable contribution	sign variable contribution	graphical representation
RF	variable importance	✓		✓		
RF	feature contribution	✓	✓	✓	✓	
NN	variable importance	✓		✓	✓	
RF	variable interactions	✓				
RF/NN	partial dependence	✓			✓	✓
RF/NN	local interpretation		✓	✓	✓	
RF/NN	tree approximation	✓	✓	✓	✓	✓

* RF: Random forests, NN: Neural networks.

Table 4: Overview of used explanations in terms of interpretability

single tree approximation provides a graphical representation of how decisions are made in the model, which can be interpreted on both a global and local scale. In addition, the hierarchical structure of the tree provides a means for assessing the relative magnitude and sign of the variable contributions.

Based on these results it is possible to answer the second and third subquestion. Variable importance measures, feature contributions, partial dependence plots and local interpretation explain how observations can be linked to the model outcomes. Variable interactions present how the random forest classifier exploits the structure in the data, which provides insights into how this technique works. Furthermore, partial dependence plots demonstrate which relationship between explanatory and response variables is detected by the model and single tree approximation illustrates the overall (simplified) logic of black box models. The next section presents the conclusion.

7 Conclusion

In this thesis it is researched how machine learning techniques can be made more interpretable. To answer this research question, it is investigated how to define interpretability and several measures are outlined that assess interpretability. Next, various techniques that provide insights into black box models are applied in an empirical study and each technique is assessed in terms of interpretability. The interpretability of an explanation is evaluated by verifying whether it provides global or local understanding, whether the magnitude and sign of the contribution of each variable can be determined and whether a simplified graphical depiction of the model can be made, giving insights into how the model works.

This paper demonstrates that global insight into the relationship between the explanatory variables and a particular outcome on a global level can be attained by constructing variable importance measures or partial dependence plots. On a local level, feature contributions can be constructed, or a local linear approximation of the model can be made to give a detailed explanation on the prediction outcome of single instances. In the case of random forests, some global understanding into how the model reaches its classification results is given by inspecting the variable interactions that the classifier exploits. Also, single tree approximation provides an approximating graphical representation of the black box model, which allows for understanding on a both global and local scale how decisions in the model are reached.

Some limitations of this thesis make further research expedient. The data used in this thesis did not show many non-linear patterns between the explanatory and response variables. Therefore, there are in principle no reasons to deviate from using logistic regression to predict default if interpretability is of main concern. More specifically, since recall is a relevant measure for accuracy in our empirical example, the performance of random forests is relatively poor compared to logistic regression, therefore it cannot be seen as an alternative to logistic regression in this example. Hence, using data with more non-linear relations could be more useful for a further application of machine learning techniques such as random forests and neural networks.

Furthermore, the data did not contain any sensitive information, which makes the ethical aspect of the need for an explanation not applicable in the context of this research. An interesting angle for further research would be using data that do contain information that could raise racial or discrimination issues. The performed empirical study provides insights into which factors affect the probability of default and tries to illustrate how the black box methods work. However, these results cannot be generalised, since using another data set could lead to different outcomes. Therefore, it could be considered to perform a simulation study to demonstrate the validity of the results.

In this thesis explanations are compared in terms of interpretability. Question remains how accurate these explanations are, since this has only been assessed for single tree approximation. In further research it could therefore be tried to derive statistical significance for each explanation, which could add trust to the outcomes. In addition, for each presented explanation method it has only been assessed whether a criterion of an interpretable model is met or not. However, more extensive research is needed on how to quantify the degree of comprehensibility of an explanation, which allows for a more elaborate comparison across methods. Yet, it has to be noted that a study of measures which are able to quantify comprehensibility can be challenging, since interpretability remains a subjective matter. It highly depends on the user knowledge, the user experience level, the time available to understand the explanation and the context of the research.

References

Andrews, R., Diederich, J., & Tickle, A. B. (1995). Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-based systems*, *8*(6), 373–389.

Biau, G. (2012). Analysis of a random forests model. *Journal of Machine Learning Research*, *13*(Apr), 1063–1095.

Breiman, L. (1996). Bagging predictors. *Machine learning*, *24*(2), 123–140.

Breiman, L. (2000). *Some infinity theory for predictor ensembles* (Tech. Rep.). Technical Report 579, Statistics Dept. UCB.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5–32.

Breiman, L. (2004). Consistency for a simple model of random forests.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). Classification and decision trees. *Wadsworth, Belmont*, *378*.

Breiman, L., et al. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, *16*(3), 199–231.

Caliskan-Islam, A., Bryson, J. J., & Narayanan, A. (2016). Semantics derived automatically from language corpora necessarily contain human biases. *arXiv preprint arXiv:1608.07187*, 1–14.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321–357.

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

Fitzpatrick, T., & Mues, C. (2016). An empirical comparison of classification algorithms for mortgage default prediction: evidence from a distressed mortgage market. *European Journal of Operational Research*, *249*(2), 427–439.

Freitas, A. A. (2014). Comprehensible classification models: a position paper. *ACM SIGKDD explorations newsletter*, *15*(1), 1–10.

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The elements of statistical learning* (Vol. 1). Springer series in statistics New York.

Garson, G. D. (1991). Interpreting neural-network connection weights. *AI expert*, *6*(4), 46–51.

Gibbons, R. D., Hooker, G., Finkelman, M. D., Weiss, D. J., Pilkonis, P. A., Frank, E., ... Kupfer, D. J. (2013). The cad-mdd: a computerized adaptive diagnostic screening tool for depression. *The Journal of clinical psychiatry*, *74*(7), 669.

Guidotti, R., Monreale, A., Turini, F., Pedreschi, D., & Giannotti, F. (2018). A survey of methods for explaining black box models. *arXiv preprint arXiv:1802.01933*.

Henelius, A., Puolamäki, K., Boström, H., Asker, L., & Papapetrou, P. (2014). A peek into the black box: exploring classifiers by randomization. *Data mining and knowledge discovery*, *28*(5-6), 1503–1529.

Henelius, A., Ukkonen, A., & Puolamäki, K. (2016). Finding statistically significant attribute interactions. *stat*, *1050*, 22.

Hothorn, T., Hornik, K., & Zeileis, A. (2006). Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical statistics*, *15*(3), 651–674.

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255–260.

Lek, S., & Guégan, J.-F. (1999). Artificial neural networks as a tool in ecological modelling, an introduction. *Ecological modelling*, *120*(2-3), 65–73.

Lowry, S., & Macpherson, G. (1988). A blot on the profession. *British medical journal (Clinical research ed.)*, *296*(6623), 657.

Martens, D., Baesens, B., Van Gestel, T., & Vanthienen, J. (2007). Comprehensible credit scoring models using rule extraction from support vector machines. *European journal of operational research*, *183*(3), 1466–1476.

Menze, B. H., Kelm, B. M., Masuch, R., Himmelreich, U., Bachert, P., Petrich, W., & Hamprecht, F. A. (2009). A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data. *BMC bioinformatics*, *10*(1), 213.

Olden, J. D., & Jackson, D. A. (2002). Illuminating the “black box”: a randomization approach for understanding variable contributions in artificial neural networks. *Ecological modelling*, *154*(1), 135–150.

Olden, J. D., Joy, M. K., & Death, R. G. (2004). An accurate comparison of methods for quantifying variable importance in artificial neural networks using simulated data. *Ecological Modelling*, *178*(3-4), 389–397.

Özesmi, S. L., & Özesmi, U. (1999). An artificial neural network approach to spatial habitat modelling with interspecific interaction. *Ecological modelling*, *116*(1), 15–31.

Palczewska, A., Palczewski, J., Robinson, R. M., & Neagu, D. (2014). Interpreting random forest classification models using a feature contribution method. In *Integration of reusable systems* (pp. 193–218). Springer.

Quinlan, J. R. (1993). *C4.5: programs for machine learning.* Elsevier.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you? explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135–1144).

Sarda-Espinosa, A., Subbiah, S., & Bartz-Beielstein, T. (2017). Conditional inference trees for knowledge extraction from motor health condition data. *Engineering Applications of Artificial Intelligence*, *62*, 26–37.

Smith, S. J., Ellis, N., & Pitcher, C. R. (2011). Conditional variable importance in R package extendedforest. *R vignette¡ http://gradientforestr-forger-projectorg/Conditional-importancepdf*.

Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T., & Zeileis, A. (2008). Conditional variable importance for random forests. *BMC bioinformatics*, *9*(1), 307.

Strobl, C., Boulesteix, A.-L., Zeileis, A., & Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC bioinformatics*, *8*(1), 25.

Thrun, S. (1993). *Extracting provably correct rules from artificial neural networks.* Sekretariat für Forschungsberichte, Inst. für Informatik III.

Zhou, Y., & Hooker, G. (2016). Interpreting models via single tree approximation. *arXiv preprint arXiv:1610.09036*.