Master Thesis

Econometrics and Management Science

Business Analytics and Quantitative Marketing

Increasing Summons Efficiency with Predictive Modelling

Youri de Koomen

ERASMUS UNIVERSITY ROTTERDAM

Erasmus School of Economics

Supervisor: Prof. dr. S. Ilker Birbil

Co-reader: dr. Andrea A. Naghi

November 6, 2018

Abstract

Invoicing is a useful tool that facilitates large parts of our economic system, but can also be of great risk to corporations when debts are left unpaid. Summoning someone to court is a costly procedure to force them to pay after all. However, when this person is simply unable to pay, this forms a significant loss for business and society. This thesis explores the decision making process behind summoning someone, looking to predict whether it is favourable to start these proceedings.

We train a wide range of models from a statistics and machine learning background to perform this analysis. Various model evaluation measures show that a Random Forest is of best use, providing a higher increase in performance than is typically seen in closely related fields like credit scoring.

The results obtained from the model, albeit low according to conventional standards, allow us to identify a sizeable share of potential cases that are better pursued outside of a courtroom. Implementation of the model could reduce investments into the summons process by 10 to 20 percent, proving to be of great business value.

Keywords: Credit Management; Summons Process; Machine Learning; Predictive Modelling; Random Forest.

Acknowledgements

I would like to express my gratitude to the many people who have provided support during my time of writing this thesis. I would like to start with thanking my colleagues at DirectPay for the knowledge and insights they have shared over the past two years, especially my thesis supervisor, Colin Nugteren, who has guided me on this project from the start. I would also like to thank my thesis supervisor at the Erasmus University Rotterdam, Prof. dr. S. Ilker Birbil, who has helped me see this process through and shape the thesis into its current form. Next I would like to thank my fellow students whom I have shared this journey with over the past four years. And finally, a special thanks to my parents, sister and family, who have supported and inspired me throughout my education and life in general.

Contents

1 Introduction

Selling an item or service on invoice creates the inherent risk of the customer failing to pay the amount due. When this happens, the company can choose to attempt to retrieve the amount from the customer themselves, or let an external party take over. One such party is DirectPay, a part of the Credit Exchange Group, which provides credit management solutions across several industries.

These services range from initialising communication to the customer to completely taking over the accounts receivables process. In each case, collecting a debt can be divided into three phases: debt management, summons and confiscation. In the debt management phase, the customer is approached with a friendly attitude in order to collect the invoice without having to take severe measures. If this fails, the customer can be summoned to appear in front of the district court in order to obtain a legal order to pay the invoice. When successful, a whole new range of confiscation options are available in order to collect the amount.

Besides the negative effects on customer relations, measures taken in the summons and confiscation phase are very costly. Costs are made for the bailiff, the court and the lawyers. These costs are initially paid for by DirectPay and can be seen as an investment to collect a claim. After obtaining a positive verdict from the court an attempt can be made to recover the costs from the customer. This makes efficiency in the summons process not only a financial goal, but also of societal importance. When there is no realistic expectation that the customer will be able to pay, these costs are a complete loss for DirectPay and the capacity of the judicial system.

The incurring of extra costs also has the effect of creating *problematic debt*, a term used for situations where people cannot oversee and overcome their debt situation independently. The costs made in the debt collection process after someone loses track of their outstanding debts have an accelerating effect. This becomes clear when looking at the following example; a relatively small debt of €50 can incur a large amount of fees: €40 for collections, €80 for a bailiff to serve the summons, €80 for the court and then again €80 for a bailiff to serve the judgement or verdict. In this case, the debt itself is not so much the problem as the eventual consequences of not having a sufficient overview. It is a sensible idea to investigate whether we can minimise the occurrence of problematic debt as much as possible.

The goal of this thesis is to develop and implement a *summons score*, derived from a model for the decision to start summons proceedings. This brings us to the following questions:

1. What characteristics can be used to predict whether a summons case is likely to succeed or fail?

2. How can we use predictive modelling to lower costs and increase efficiency in the summons process?

To answers these questions we compute an estimation of the costs that DirectPay makes to start legal proceedings and subtract these from the payments that a debtor makes following this action. We mark the outcome as a *success* when this figure is above 0 and a *failure* if it is not. This variable is then used to train and compare a wide range of models from a statistics and machine learning background.

The proposed research sheds a light on an application of these classification algorithms that has received little attention in the literature. We can achieve great cost reduction for stakeholders in both business and societal context.

1.1 Debt and Credit Management in The Netherlands

In 2016 between one and one-and-a-half million Dutch households were in a situation of risky or problematic debt, which amounts to over 15 percent of households in the Netherlands (Hoff et al., 2016; Westhof et al., 2015). This is the case when someone satisfies one of the following conditions:

- at least three types of defaulted invoices;
- late payments in excess of €500;
- late payment on rent, mortgage, utility or health insurance;
- overdrawn in excess of €500 at least 5 times a year;
- credit card debt in excess of €500.

We can conclude that a large share of the population is in a long term situation where a single event can create large financial distress. In these cases a single summons with corresponding incurred costs may be the beginning of a chain reaction, which makes it in the interest of credit management companies and society in general to prevent this. Figure 1 shows a map on the severity of problematic debt, aggregated on Dutch postal code level.

Figure 1: Map showing the severity of debt on Dutch postal code level, on a scale from yellow to deep red (DirectPay, 2014).

It would be more appropriate to do recommendations for a debt counselling programme, providing services ranging from budgeting advice to strict restructuring procedures. According to Hoff et al. (2016) Dutch debt counselling organisation already have over 200,000 households registered, with other sources such as van Putten and Schoot Uiterkamp (2017) estimating that this could range well into the 300,000's. This does not include households that apply for advice but do not enter a prolonging trajectory.

The debt collection industry in The Netherlands is estimated to consist of over 600 companies, from around 500 ten years earlier. Since it is not regulated by law and the Dutch term for a debt collection agency is not protected, accurate numbers on the size of the market are hardly available (Geurts, 2012; ABN AMRO, 2016). The NVI, a self-regulated union of about 30 agencies, estimates that their members possess a collective market share of 65 to 70 percent. Following their statistics, the amount of cases that are currently active is around 6 million, with an outstanding debt of over €15 billion (NVI, 2018).

The part of the collection process that involves the legal system is highly documented and transparent. The *Royal Professional Organisation of Judicial Officers in The Netherlands* (KBvG) certifies and regulates all bailiffs and provides a great source of information. Since the act of process serving is carried out by bailiffs, we know that in 2016 and 2017 they signed 523,700 and 404,500 summons documents, respectively (KBvG, 2018).

1.2 Phases of Debt Collection in The Netherlands

We now look into the laws and processes that involve collecting a claim in The Netherlands. The chain can be divided into three phases: debt management, summons and confiscation. Figure 2 shows the regular time frames of these phases and highlights the decision making moment that is the subject of this thesis: the summons decision.

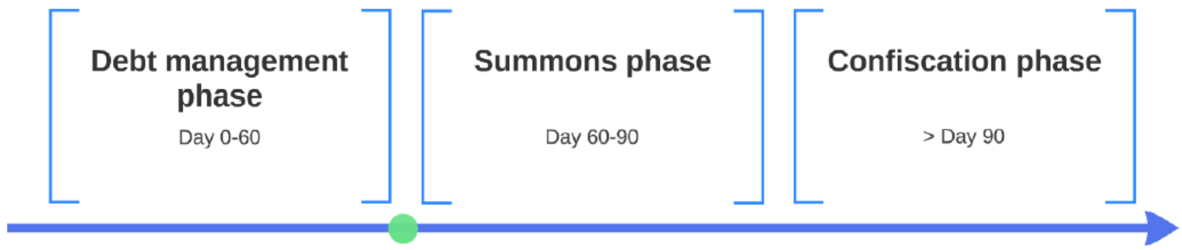

Figure 2: The three main phases of the debt collection system in The Netherlands. The green dot represents the moment the summons decision may be made (DirectPay, 2014).

The debt management phase starts at the day of invoice. Initially this debt is not necessarily a bad thing and if issued and repaid correctly it can be of great use. The terms of payment are usually set at 14 days after which a customer is in default. We speak of a claim when an invoice or instalment is not paid (within these terms); a reminder can be sent at this point. This reminder initiates another 14 day term, during which the debtor can still pay without any additional costs being incurred (Dutch Civil Code, 2012). After this second term expires, another reminder can be sent that incurs the additional costs and suffices as a final warning before legal measures can be taken. Around one and a half month has now passed since the creation of the original claim; three terms of at least 14 days each. In general, 60 days are allocated for the pre-legal debt management phase.

Next is the summons phase, which makes use of the Dutch legal system. If in the debt management phase sufficient measures have been taken to notify the debtor of outstanding claims, the case can be tried in court. A bailiff has to serve a summons document in order to initiate court proceedings; this is done in person. The summons contains all relevant information about the claim, defendant, plaintiff, bailiff and court proceeding.

At the hearing both parties get to present their case to a judge. The judge decides whether the claim, or parts of the claim, is valid and pronounces a verdict. If the defendant does not appear before court, a judgement in absentia is made, which happens in about 75 percent of cases. Usually these types of verdicts order the defendant to pay the full amount with little option to appeal.

The confiscation phase begins after a successful verdict has been obtained. The debtor is given another possibility to fulfil the claim voluntarily, that is, without having to apply legal confiscation measures. A bailiff is allowed to utilise several harsh confiscation options such as seizure of income or assets (bank accounts, furniture, cars et cetera).

A verdict is executable as long as the limitation period of twenty years has not passed. This limitation is interruptable by notifying the debtor of the verdict, which prolongs it with another 5 years at maximum.

1.3 Costs of Collection

The actions above have its costs, which we will go through chronologically. The operating costs of the collection company are incurred as a combined additional cost after the second reminder of the claim. This cost is called the WIK, referring to the law that regulates the incurrence of it (Dutch Civil Code, 2012). It is built up as follows:

- 15% over the first €2,500;
- plus 10% over the next €2,500;
- plus 5% over the next €5,000;
- plus 1% over the next €190,000;
- plus 0.5% over the rest of the claim;

with a minimum of €40 and a maximum of €6,775.

When a bailiff initiates legal proceedings, costs rise quickly. Here a bailiff fulfils the role of process server by bringing out the summons, which costs around €80 and increases every year by about 1.3 percent (Dutch Civil Code, 2018a). Court fees are again dependent on the size of the claim and are determined as follows (Dutch Civil Code, 2018b):

- €119 if the claim is below €500;
- €476 if the claim is over €500 and no more than €12,500;
- €952 if the claim is over €12,500.

A verdict is pronounced as result of the court case, but that is not sufficient to go to the confiscation options directly. First, the verdict has to be served to the debtor by a bailiff. This carries the same cost as serving a summons document, €80 as specified by Dutch Civil Code (2018a). This set of regulations also specifies the costs corresponding to the various confiscation options, which brings upward of €100 in additional costs.

1.4 Summons Decision

Attempting to collect a claim consists of various complex and costly processes. Currently DirectPay only takes legal reasons into account when deciding whether to pursue the case in court, which leads to an opportunity for implementing a statistical model. Compared to the status quo of trying all possible cases, a lot of costs are made and incurred even though statistics might be able to indicate there is little expectation of payment.

The summons score should provide insight into this expectation and the ability to pay, even if the end user has no knowledge of econometric theory. Instead of using model output or summary statistics, its goal is to provide value for those working on operational processes, sales and portfolio management.

1.5 Contributions

An interesting statement of Baesens et al. (2015) is the following: "The most accurate classifier does not necessarily give the most profitable scorecard." In extension of this, the results of this thesis confirm the idea that a generally worse or outright bad performing model can still provide sufficient business value. One of the contributions of this thesis is to show the business implications of machine learning models, beyond purely interpreting model coefficients and statistics.

Another interesting implication is the importance of more complex techniques in comparison to baseline models, such as linear regression. In credit scoring literature, the yield of implementing these techniques is typically a lot lower than what is found in this thesis. A possible explanation is the extent of the information available in the data set and the complexity of the variables due to the laws and regulations that have been explained in this introduction.

1.6 Outline

The rest of this thesis contains a literature review, methodology and description of the data set, followed by a presentation of the results of this research and the conclusions that can be drawn from it. The appendix features the tools that are used in this research and details on the implementation of the analysis, as well as tables, figures and information on sources that are referred to in the main text.

2 Literature Review

The decision to summon a debtor is not largely discussed in the literature as a specific problem, if at all. However, the models that would be appropriate for our analysis have also been used in a similar field. The idea of rating the willingness and ability of an entity to pay, after we have found out they initially did not pay, is not that different from the subject of credit scoring, which does this evaluation before the purchase has even occurred. This area however has been researched extensively.

Most of the techniques that will be used have a background in statistics and machine learning. Hastie et al. (2001) provide a general introduction to these fields and the most popular methods originating from them, many of which are used in this thesis.

In credit scoring, the goal is to assign a value to the credit worthiness of an entity; a person, household, company et cetera. The range of applications is rather wide; from small online retail to large loans like mortgages or even companies and countries. The models we would like to use to analyse summons efficiency have also been used in this context; we take a look at the literature concerning credit scoring based on statistical modelling and other quantitative methods. The most complete overview to date of credit scoring literature is provided in Baesens et al. (2015). Top ranking models are Support Vector Machines, Ensemble methods such as Gradient boosting and Random Forests, and Neural Networks.

Support Vector Machines have found an application in credit scoring modelling. Huang et al. (2007) analyse two popular data sets on credit from the UCI (University of California, Irvine) machine learning repository. They explore the tuning of hyperparameters, as will be explained later on, that leads to the optimisation of the Support Vector Machine. Maldonado et al. (2017) also employ this practice for their research into bank loans to micro-entrepreneurs in Chile. They find Support Vector Machines to have comparable if not superior performance to traditional models, but did so with less variables; as information costs were relatively high, Support Vector Machines provided maximum profit.

Wang et al. (2011) test the performance of ensemble algorithms (stacking, boosting and bagging) on the earlier mentioned UCI credit data sets in addition to that of a Chinese bank. They find stacking all the considered models and bagging decision trees to significantly improve accuracy and type I/II error rate.

An early show of the potential of neural networks in a credit union environment can be found in Desai et al. (1996). However, *generic* models seem to be relatively lacklustre and the sample sizes were very small in terms of modern standards. West (2000) shows two additional, more complex neural network structures that provide improved performance beyond both existing methods and the raw *generic* network setup. A comparison between five neural networks and five more traditional models shows that these new structures are very competitive to methods like logistic regression. Tsai and Wu (2008) feature the usage of ensemble methods compared to using just a single trained network. Using (again) small data sets the authors find insufficient evidence to show that their considered new combinations provided increased accuracy.

Baesens et al. (2015) also provide a useful look on the use of model evaluation measures in the credit scoring literature. Simple measures as the accuracy or true positive rate are used in almost all of the papers they reviewed. Almost half of the reviewed papers employ statistical hypothesis testing, while a quarter looks at the area under the receiving operating curve (AUC). This last criterion was first introduced by Hanley and McNeil (1982), who show a relationship between being able to distinguish observations belonging to different classes, the Wilcoxon statistic and the AUC. A closely related measure that is not explicitly mentioned is the usage of the lift curve. Mozer et al. (2000) show the potential value of using this characteristic in predicting churn in the telecommunications industry.

3 Methodology

To predict whether a summons is successful, we first have to define what *success* means in this context. We can attempt to predict the actual, monetary, profit of a summons and estimate parameters based on the overall profit of our data set. In this case, we get a continuous target variable. An issue that can arise when using this approach is having a large bump in frequency where cases brought in no money but only incurred the standard (almost fixed) costs. These bumps can also be observed when looking at cases where the incurred costs are retrieved from the debtor, leading to a large positive constant.

This has the characteristics of a binary target variable, where the outcomes are denoted as failure (0) or success (1). A classification approach might be more suitable. The models we intend to use in this case can be divided into the following classes: logistic regression, tree based algorithms and other machine learning methods.

The last class contains other models that belongs to the field of machine learning, namely support vector machines and neural networks. Tuning the characteristics or hyperparameters of these methods allows it to pick up more complex relationships in the data than with more traditional methods, but at the cost of interpretability.

To improve the interpretability of the results for business users, we convert the probability of success to a *summons score*. We explore several methods of converting the probabilities from the model to a scale that is more accessible.

3.1 Logistic Regression and Binary Target Variables

Logistic regression (LR) is generally regarded as a method providing a good baseline with relatively little effort. The logistic function (1) maps the real numbers to the interval $(0, 1)$ and is used as the link function required to accommodate for probabilities of a binary variable,

$$f(x) = \frac{1}{1 + e^{-x}}. \tag{1}$$

As input for this link function, that gives us the probability of an observation belonging to a class, we use a linear model. The transformed model belong to the class of generalised linear models. We assume the data generating process for latent variable y_i^* as

$$y_i^* = x_i\boldsymbol{\beta} + \epsilon_i$$

for observations $i = 1, \ldots, n$. The model is written as $x_i\boldsymbol{\beta} = \beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{ij}\beta_j$, which is linear in the explanatory variables $j = 1, \ldots, k$. The error term ϵ has a standard logistic distribution. We translate this to our observed binary variable y as follows:

$$y_i = \begin{cases} 1, & \text{if } y_i^* > 0; \\ 0, & \text{otherwise.} \end{cases}$$

The resulting coefficients are easily interpretable after we dive into the implications of this transformation. The equations below illustrate these relations; the probability of the outcome being one depends solely on a transformation of our linear model. Also, the log odds, the natural logarithm of the probability of the outcome being one with respect to the probability of the outcome being zero is equal to our model equation. That is

$$P(Y_i = 1|\mathbf{X}_i) = f(\mathbf{X}_i\boldsymbol{\beta}) = p_i,$$
$$\text{such that } f^{-1}(P(Y_i = 1|\mathbf{X}_i)) = \ln\left(\frac{p_i}{1-p_i}\right) = \mathbf{X}_i\boldsymbol{\beta}.$$

The interpretability of our estimates is as follows: a unit increase in a variable, say x_1, leads to an increase of the log odds ratio by the coefficient β_1. Equivalently, the odds ratio increases by the exponent of this coefficients e^{β_1}.

3.2 Classification Trees

Another class of models we are interested in are tree based algorithms. In general, decision trees try to assign a value to an observation by applying some decision rules on its characteristics. In the context of binary target variables, we call this a classification tree.

A classification tree starts with a node containing all observations (the root). It then tries to find the best separation of the data, according to the true value of the target, by splitting this node into two new nodes. This decision is made by looking at a single rule based on one of the explanatory variables. By repeating this procedure for the resulting nodes until some stopping condition is reached, we end up with a set of easily interpretable rules from which the predicted class is derived. Afterwards, there is the option to remove splits in order to reduce the risk of overfitting, this is called pruning.

First we have to define an objective in order to grow our tree: the splitting criteria. We assign an impurity score based on Gini impurity and Entropy measures. These measures assign a high impurity score to balanced nodes, say proportions equal to 0.5, and a low impurity score when the node is homogeneous in the target variable, with as extreme case proportions equal to 0 and 1. The scores of a node are given by the following formulas:

$$\text{Entropy: } i(\tau) = -\sum_{j=1}^{J} p_j \log_2 p_j;$$
$$\text{Gini: } i(\tau) = 1 - \sum_{j=1}^{J} p_j^2,$$

where τ is a node and p_j is the proportion of observations belonging to class j in that node, $j = 1, \ldots, J$. Observe that p_j at τ is equal to $\frac{N_\tau^j}{N_\tau}$, if N_τ^j is equal to the amount of observations belonging to class j in τ, and N_τ for the total amount of observations. We can then use these measures to compute which split s creates the maximum impurity reduction:

$$\Delta i(s,\tau) = i(\tau) - \sum_{b=1}^{B} \frac{N_{\tau_b}}{N_\tau} i(\tau_b),$$

where $\tau_1, \ldots, \tau_B$ are the nodes (branches) as a result of split s.
Then we have to create a stopping criteria, a couple of examples:

- Amount of observations in a node is below some threshold (a constant or percentage of the total amount of observations)
- Impurity of the node is below some threshold
- Depth of the node is over some threshold (complexity)
- Additional splits create no a gain in accuracy

As mentioned before, these rules prevent the tree from accommodating to a small amount of specific cases in the training sample. This could result in a set of rules that do not generalise to cases outside of the training sample, which is called overfitting.

It is possible that the stopping criterion can not sufficiently prevent overfitting, therefore we may decide to employ a pruning procedure after the tree has stopped growing. Depending on the change to the total impurity of the tree, we may decide remove some splits. We use a pruning rule derived from Hastie et al. (2001). Define the per-leaf change in impurity when pruning the complete tree (tree 0) to the root node (result is a *stump*, s'):

$$\left.\frac{\Delta i}{\Delta N}\right|_0 = \frac{i_0 - i_{s'}}{N_0 - 1},$$

and in general, replacing the subtree below node n with a leaf

$$\left.\frac{\Delta i}{\Delta N}\right|_n = \frac{i_n - i_\lambda}{N_n - 1},$$

where i_n is the impurity of node n and N_n the amount of leaves below node n. We stop pruning when the following condition is met

$$\frac{\left.\frac{\Delta i}{\Delta N}\right|_n}{\left.\frac{\Delta i}{\Delta N}\right|_0} \geq \alpha.$$

This corresponds to the per-leaf cost of pruning the next node compared to the per-leaf cost of pruning the complete tree. Constant α can be tuned to in- or decrease the degree of pruning.

One of the main advantages of decision trees lies in the simplicity of the splitting rules, as the data is divided into groups based on if-then-else statements. It also does not rely on assumptions such as the distribution of variables or the error term, or the absence of co-linearity in the explanatory variables. A disadvantage of the relative simplicity is that the performance of a single tree tends to be lower than with other models, in addition to an increased risk of overfitting.

3.3 Ensemble Methods

We may also choose to repeat the tree construction in an attempt to improve the overall analysis and overcome the flaws that exist when using only one tree. The goal is to increase the stability of the final model and to reduce the variance of the predictions. Algorithms of this kind are called ensemble methods, in particular we consider using bagging and boosting trees.

Bagging (bootstrap aggregating) involves repeatedly taking a bootstrap of the data (a random sample with replacement), fitting the model to those bootstrap samples and combining the outcomes of those models. It was developed in Breiman (1996) and can be used on all kinds of models and can be seen as a parallel procedure.

Boosting on the other hand is a sequential procedure, where the sampling in a certain iteration depends on the previous model. The intuition involves building a large sequence of simple *weak* learners, in order to create a combined *strong* learner (Shapire, 1990). As opposed to bagging, we attempt to overcome the failure of previous models by moving away from the fully random sampling method. Instead, observations are weighted, where the weight increases with the error in the previous. Freund and Schapire (1996) and Friedman et al. (2000) provide algorithms that have lead to the implementation of ideas, where the latter provides a focused view on applying boosting to classification trees.

A problem that haunts the use of bagging in a tree setting is the existence of correlation between the trees, as this means that the predictions are correlated as well. If a variable is of high explanatory value over the whole data set, it is likely to be included in most if not all of the bagged trees. That is the characteristic that defines Random Forests (as developed in Breiman, 2001) from regular bagging: selecting a subset of the explanatory variables and creating a tree with those variables. The size of this subset is generally set at the square root of the amount of variables available.

The downside of these ensemble methods is decreased interpretability, since a large amount of models are combined. The individual models provide no interpretation on their own, as these algorithms assign different weights each iteration or alternate between subsets of variables. Because of that we need to look at measures that indicate variable importance. For boosting we use a relative importance measure as shown in Friedman and Meulman (2003), which allows us to rank the variables on their explanatory value. To provide a ranking for Random Forests we average the decrease in Gini impurity for each variable, as explained in Breiman and Cutler (2003).

3.4 Support Vector Machines

Support vector machines (SVM) are a class of models developed in the 1970s, with the modern version that is viewed as the current standard developed by Cortes and Vapnik (1995). The optimisation function attempts to find the hyperplane that separates the two classes by the largest margin. This hyperplane is then defined by the observations of both classes that are closest to the plane, which are called Support Vectors. These observations are assigned strictly positive weights, while the inactive observations have weight zero. The formalisation of SVM's optimisation function is as follows:

$$\begin{aligned} \underset{\mathbf{w},b,\xi}{\text{minimize}} \quad & \frac{1}{2}\mathbf{w}^{\intercal}\mathbf{w} + C\sum_{i=1}^{N}\xi_i \\ \text{subject to} \quad & y_i(\mathbf{w}^{\intercal}\phi(x_i) + b) \geq 1 - \xi_i, \quad i = 1, \ldots, N; \\ & \xi_i \geq 0, \qquad\qquad i = 1, \ldots, N, \end{aligned}$$

where $y_i \in \{-1, 1\}$ is the response variable and x_i are the explanatory variables for observations $i = 1, \ldots, N$. In the SVM context $\mathbf{w}$ denotes the weight vector corresponding to the observations, b the bias and ξ the error. The cost C is the first hyperparameter introduced in SVM; which provides a balance between adding additional support vectors (at the risk of overfitting) and the classification error ξ.

More complex problems feature cases that are not linearly separable, which can become a problem with the standard SVM implementation. Because of that, Boser et al. (1992) suggest applying a non-linear kernel to the explanatory variables. The kernel allows creation of a linear decision boundary in an implicit, high-dimensional feature space. Before getting into the workings of kernels and the different available types, we look at the dual problem, obtained by applying the Lagrangian to the primal problem above. The formulation is as follows:

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{maximise}} \quad & \frac{1}{2}\boldsymbol{\alpha}^{\intercal}Q\boldsymbol{\alpha} - \mathbf{e}^{\intercal}\boldsymbol{\alpha} \\ \text{subject to} \quad & 0 \leq \alpha_i \leq C, \ i = 1, \ldots, N; \\ & \mathbf{y}^{\intercal}\boldsymbol{\alpha} = 0, \end{aligned}$$

where α_i is the Lagrange multiplier and Q a matrix with $Q_{ij} = y_i y_j K(x_i, x_j)$ and kernel function $K(x_i, x_j)$. The final classifier or decision function for a new input x is equal to $f(x) = \text{sgn} \sum_{i}^{N} \alpha_i y_i K(x, x_i) + b$. The suggested kernels can be found in Table 5.

3.5 Neural Networks

Artificial neural networks, or simply neural networks (NN), is a class of models used in machine learning. It can be used to model highly non-linear relationships by using multiple layers of nodes connected by link or activation functions (as seen in the context of the logistic function and SVM kernels). A subclass of NN that is widely used for classification purposes are multilayer perceptron (MLP) networks.

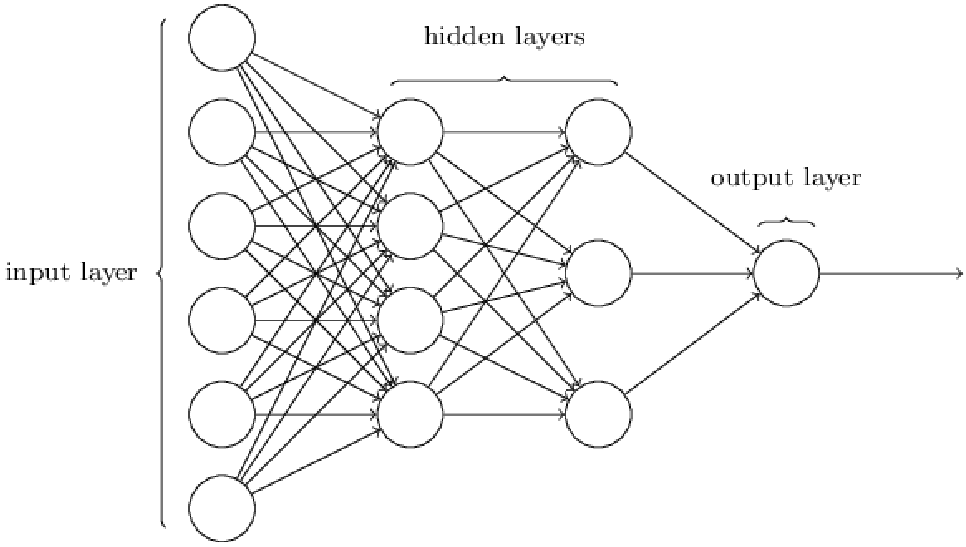

Figure 3: A general example of a four layer MLP, consisting of hidden layers with four and three nodes (Nielsen, 2017)

In Figure 3 a general example is shown. Six input nodes (explanatory variables) are combined in each of the four hidden nodes in the first hidden layer. The combination can differ through the weights and specific activation functions. The combined values of these four hidden nodes are then used as input for the next hidden layer, featuring three nodes. The resulting target variable, our output layer, is specified by yet another combination of our last hidden layer.

To show how this works computationally, we take the first node of the first hidden layer featured in Figure 3. Say we use the logistic function in equation (1) as our activation function, we have six input variables multiplied by their respective weights, plus a bias (constant). Added up, this is our input for the activation function, which requires seven coefficients to be estimated. This amounts to 28 coefficients for the first hidden layer, 15 for the second hidden layer and four for the output layer; 47 coefficients are estimated for this model in total.

This complex structure with a lot of coefficients (in comparison to the amount of input variables) leads to models that can handle very non-linear relationships between the variables.

3.6 Tuning hyperparameters

The machine learning models presented in the methodology each entail a class of models rather than a particular model. The difference can be created by tuning hyperparameters that define the exact model architecture, such as the size of the ensemble or a cost parameter. With models like SVM and neural networks we also have the freedom to choose kernels and activation functions.

To assess which setting of parameters and functions results in the best performance on the test set and future summons cases, we make multiple random divisions of the data set and look at the average generalisation performance.

We use a grid search when multiple parameters need to be optimised. This means a set of allowed values for each parameter is specified, after which the outcomes of all possible combinations are compared. We take the SVM model as an example. The dual formulation allows us to have multiple kernels and up to two hyperparameters for each kernel. Figures 7.3 and 5 show a visualisation of one of the grid searches done on a SVM with the RBF kernel.

3.7 Model Evaluation and the Summons Score

The final part of the methodology is regarding the need to evaluate the performance of our classification models. In order to select a model on which we base the summons score, three measures are considered that approach the comparison in different ways.

The first measure is the accuracy (maximisation) or the misclassification rate (minimisation). It follows directly from the confusion matrix, which shows the frequencies of the classifications compared to the true values. Since we are trying to classify between two classes this is a 2 by 2 table, see Table 1 for an example. For notational purposes we have our classes: Positives (P) and Negatives (N). The entries of the confusion matrix follow from the predicted classes: True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN).

		True value	
		Positive	Negative
Predicted value	Positive	TP	FP
	Negative	FN	TN

Table 1: Example of 2 by 2 confusion matrix.

The accuracy is the amount of correctly classified observations divided by the total amount of observations: $\frac{TP+TN}{P+N}$. By showing the classification results of a single cut-off value, it is a simple, useful tool for model evaluation, but this also presents a drawback.

The accuracy does not show the value of a model by certainty of its predictions and does not allow prioritisation of one of the two decisions.

A way to get insight into these decisions is the lift, which is a derivation of the gain. To obtain these numbers, the charts they produce and corresponding lift score, the probabilities obtained from the model are ranked, after which the proportion of the *covered* target population (True Positive Rate, TPR) is plotted against the proportion of the total population that is targeted; $\frac{TP}{TP+FN}$ plotted against $\frac{TP+FP}{P+N}$. The lift is the ratio of these values, which means that it describes how many more observations of the target are selected through the model compared to a random selection. Examples of the gain and lift charts can be found in Figures 10 and 11.

The final measure we use involves the area under the receiver operating characteristic (ROC) curve, commonly referred to as the AUC. The AUC has a convenient probabilistic meaning: Hanley and McNeil (1982) report there is an equivalence between the AUC and the probability of correctly classifying randomly chosen pair of one negative and one positive sample point. The ROC curve is defined as the TPR against the False Positive Rate (FPR). The TPR, $\frac{TP}{TP+FN}$, tells us what percentage of the positive sample points we classified positively. The FPR, $\frac{FP}{FP+TP}$, tells us what percentage of the negative sample points we classified positively. By plotting these two measures against each other at different cut-off values, thus by allowing cases that the model could not classify accurately, we obtain a curve that represents this trade-off. It is related to the lift curve, but has an extra dimension to it that is useful in the context of our research. An example of an ROC curve can be found in Figure 12.

4 Data

In this section the data set will be discussed. The goal of the business rules behind this data set are to provide a group of cases with their respective information, such that it is representative of summons decisions made in the future and provides sufficient information about their success. First we will talk about the cases (rows) we base our analysis on: taking into account training and validation sets, what cases do we select? Then we will explain the variables (columns) in our data set: what is allowed, available and feasible?

4.1 Case Selection

Since our goal is to score new potential summons cases on their profitability, we select cases that have been summoned in the past. In this selection we do not care if the case has had its day in court, as the decision to summon the case carries the willingness to invest into bailiff and court fees. We do add the condition that the scheduled court date has to be at least two years ago. The justification for this secondary condition lies in specifying the success of the summons decision. For example, if a case has been brought to court yesterday, insufficient time has passed to conclude whether this was a profitable decision.

A final condition is that we do not consider company debtors in our analysis, since rules and regulations are significantly different for this group, we choose to filter these cases out of our sample. Less than three percent of all cases has a company as debtor.

4.2 Subsets of cases

One of the model assumptions is that the out of sample data (new cases that we will score with the model) has the same data generating process. Debtors in new cases should be similar in their decision making process to those belonging to older cases. However, if the conditions concerning unpaid invoices changes over time, we may not be able to build a model that is useful. This is especially worrying as we are already taking a two year grace period in adding observations to our data sample.

Because of this concern we take an extensive approach to show the potential business implications of implementing this model. A standard training, validation and test set approach is used to fit the models, tune hyperparameters and identify the model with the highest performance. The division of cases between these sets is 50, 30 and 20 percent, respectively.

4.3 Variable Selection

The information available to us can be roughly divided into three categories: debtor, address and case level. In order to explain these labels, we look at DirectPay's processes in a broader view, but also what happens before a claim ever arrives at DirectPay. Table 4 lists the explanatory variables that have been collected, including a description.

A client of DirectPay sells products or services, resulting in one or multiple invoices. After sending several reminders, the client sells a claim for collection (this act is called the cession of a claim). These events already generate a lot of relevant data before any operational processes of DirectPay have started. Invoices are created on a certain day with a corresponding amount, which forms the original information and base of the claim. The act of cession is another important event, showing us the outstanding balance at that point in time.

When one or more invoices of a debtor have been purchased from a client, we can start looking into the information we have received. The basic information required to contact a debtor and initiate collection procedures are usually required by contractual agreement with the supplier. This entails data like residential addresses and names, while sometimes details like phone numbers or emails are available. From this starting point we can search for a link to several sources of context information.

DirectPay itself is one of these sources, as we can investigate whether this person has a history with DirectPay. The credit history of a person can provide valuable information concerning what would be the most effective approach to the current situation. We also have internal and external sources that provide information on address level, which is the combination of postal code plus house number and extensions.

The internal source provides a score of credit worthiness based on the household, the latter consists of two external sources: the BAG and Cendris databases. The BAG is the Dutch *base registration of addresses and buildings*; it contains information on all addresses and building in The Netherlands, such as the intended use, size and (municipal) valuation of the property. Cendris is part of The Netherlands' largest postal company PostNL. The data set they supply to DirectPay provides context information, such as approximated income and household composition.

Finally, we have information resulting from the summons level aggregation of a case, which also relates to assessing the success of a summons. A lot of core information can be found in the summons document served by the bailiff such as amounts and dates. These can be related to past and current data; generating differences in claim amounts and the duration of business processes. Figures 6, 7 and 8 show the distribution of some of the basic variables surrounding a summons: the open amount at time of summons, the age of the claim at time of summons and the age of the debtor.

4.4 Target Variable

The goal of this thesis is to improve the *summons efficiency*, the efficiency of the summons process. To have an indication of this on a case level we sum the payments made by the debtor after the summons was delivered and subtract the costs. The payment part of this equation is relatively easy, since they are made on the case level basis and are identifiable through a case reference number. This is also the case for the bailiff and court fees.

Internal costs made by DirectPay in the form of overhead are a lot harder to compute, as labour costs are currently not traceable to a specific case. Financing costs also play a very considerable role, as interest rates are high due to the risk associated with debt collection and the time it can take for payments to start. This means our cost estimation is definitely an under-calculation, which needs to be kept in mind when looking at the results from the model.

In 44 percent of the cases the payments do not cover the costs made following the summons decision and 35 percent of the cases sees no payment at all. A histogram showing the distribution of the result is shown in Figure 9.

4.5 Data Quality

There is a big difference in quality and reliability of the various columns in the data set. Since the variables originate from several source systems, we can discuss these independently.

Data on address level is both of great and of terrible quality in The Netherlands. The great part comes from the data sets that we get from external parties. The address level context information available we receive is very reliable. The terrible part finds its origins in the structure of addresses in The Netherlands. In general a combination of ZIP code and house number should identify a single front door. An addition to this system is the extension and *house letter* that can be added to the address.

However, this creates a complexity that most accounting systems in The Netherlands do not cover. Most of the times these fields are merged or omitted. When a client of DirectPay delivers data on claims or debtors, we end up with more than one potential address in our external data sources.

We also have to deal with several suppliers, which means that parts of our data set might have different data registration standards and formats. Another kind of noise comes from free text fields, making it hard to incorporate into a model without performing additional operations. For example, phone numbers and emails are grouped into categories.

4.6 Data Transformations

Creating new variables from interactions can provide better model accuracy and more interpretable results. The dates of certain important events such as the invoice, session or summons date may be unfit to be used directly as input for a model. By taking differences and creating a variable that contains the time between the original invoice and the cession (or the cession and the summons) interpretability of the coefficients increases. This can also be done for the open amounts measured at these core dates, indicating what happens to a case before it is brought to court.

Transforming variables can also have a positive effect on the correlation with the target. Some options are standardisation to a zero mean unit variance variable, taking the logarithm or censoring extreme values. Another possibility is transforming an interval variable into a nominal (or ordinal) variable by grouping or binning certain ranges of values, as described above.

5 Results

In this section the results of our analysis are presented. A full description of how the analysis was implemented can be found in the Appendix under Tools and Implementation. The final data set we base our models on contains 34 explanatory variables on around 60,000 cases, which will be divided into train, validation and test sets containing 50, 30 and 20 percent of the cases, respectively. Variable names and descriptions (as used in this section) can be found in Table 4.

5.1 Variable Importance

To answer the first research question *What characteristics can be used to predict whether a summons case is likely to succeed or fail?* we make use of the importance of variables in the different models. For models such as Logistic Regression and a single Classification Tree, this amounts to manually looking at the output. Since we perform a stepwise variable selection method for the Logistic Regression model, we can use this selection order to identify some important variables.

Historic variables such as the amount of previous cases of the same debtor at DirectPay, whether they were paid or got to the confiscation phase were selected early in the algorithm. Some of the variables core to the current case were also chosen, such as the time between the original invoices and cession, the amount of payments done to DirectPay between the moment of cession and the initial summons decision and the total amount of the original invoices. The same was observed in the Classification Tree, where the first splits were made based on these variables and even repeated splits on the same variable can be observed.

Random Forest		Boosting	
Variable	Importance	Variable	Importance
FIN_Diff_Summons_Nominal	0.0106	FIN_Diff_Summons_Nominal	1
Credit Score	0.0055	Credit Score	0.753
FIN_Invoices	0.0050	FIN_Invoices	0.707
DATE_Diff_Cession_Invoice	0.0046	DebtorAge	0.688
DebtorAge	0.0045	DATE_Diff_Cession_Invoice	0.671

Table 2: Table showing the top five variables used by the Random Forest and Boosting Classification Trees implementations in SAS Enterprise Miner, including the values for the importance measures derived from the models. For the Random Forest this is the average decrease in Gini Impurity, thus not normalised like the importance measure for boosting.

The SAS Enterprise Miner implementation of Random Forests and Boosting Classification Trees contains output regarding variable importance, which gives us the ability to provide a ranking of the variables based on statistics. An adaptation of this is shown in Table 2. It shows us that both models identify the same 5 variables as having the most explanatory power. We observe some of the core information to the cases that were selected in the earlier models as well, in addition to the Credit Score based on the address that DirectPay supplies and the age of the debtor.

5.2 Model Comparison

The different models all identify roughly the same group of variables of having the most influence on the target. In this part we compare the outcome of the models to evaluate which one is the best candidate to base our summons score on. Table 3 shows statistics used for model evaluation and selection, after performing the analysis over 10 random training, validation and test set samples. Figures 11 and 12 show the ROC curve and a graph of the lift of the models in the Model Comparison node in SAS Enterprise Miner.

Hyper parameters were tuned by optimisation over 10 random samples, in an attempt to further improve generalisation to the test set and future summons cases. We impose a limit of 75 trees to be used in the Random Forest and Boosting models. The SVM model was also looked at, where we have the option to tune the cost parameter and try out several kernels, a grid search shows no alternative set of parameters that yields higher performance than a linear kernel with cost parameter 4. Finally, the neural network model was automatically tuned by SAS Enterprise Miner.

	Accuracy		AUC		Lift	
	Valid	Test	Valid	Test	Valid	Test
Random Forest	0.600	0.599	0.641	0.641	1.477	1.501
Boosting	0.583	0.582	0.612	0.614	1.380	1.382
Tree	0.580	0.582	0.610	0.612	1.356	1.352
Neural Network	0.524	0.526	0.581	0.582	1.279	1.288
SVM	0.517	0.518	0.586	0.585	1.293	1.276
Logistic Regression	0.503	0.504	0.522	0.522	1.225	1.223

Table 3: Table containing the accuracy, AUC and the lift for all the models implemented in SAS Enterprise Miner. For the lift statistic, a depth of 10% is used. The values reported are averages over 10 random training, validation and test set samples.

The Random Forest model is chosen with the highest performance across all these measures. However, compared to results found in the literature, the models all score quite low, which suggests there is little explanatory power in the variables that we used.

Another noticeable result is that boosting renders a practically equivalent performance compared to using just a single tree, with only a slight advantage in the lift statistic. The neural network and SVM also have low performance. This could be explained by the relatively small data set, with a training set of roughly 30,000 observations.

5.3 Summons Score and Average Profit

This final part of the results aims to answer the question: *How can we use predictive modelling to lower costs and increase efficiency in the summons process?*

In order to translate the Random Forest model into something that is graspable by the business user, we explore two options. The first option involves simply showing the probability of success as summons score; the results can be seen in Table 6. There are a couple of downsides to this way of presenting the results.

First of all, since the model has a hard time distinguishing summons cases with a low expectation of payment from those with a high expectation, the average probability we obtain is a bit over 0.5, while the rate of failure is around 40 percent. Another effect of this uncertainty is the large density of observations around the centre.

The second option is dividing the cases into 10 bins of equal sizes. Each bin contains 10 percent of the test set, which corresponds to two percent of the complete data set. The interpretability of the score increases by making each bin of equal importance, as the amount of workload and investment by summoning, say, bin 2 is roughly equal to that of bin 6 or 7. This makes it easier to see the impact of choosing not to summon a certain group of cases.

By using this binning strategy, we obtain the numbers in Table 7. We find that the lowest ranked 10 percent of summons cases yield about €13 more in payments than what was invested directly into bailiff and court fees, which corresponds to just 1.3 percent of the total result. In contrast, the best 10 percent have a positive result of almost €200, which leads to a share of over 19 percent.

6 Conclusion and Discussion

The objective of this thesis is to research the potential of predictive modelling in the context of the summons process.

Using various models from a statistical and machine learning background, the characteristics with the highest explanatory and predictive power were identified. The most influential were basic variables describing the payment history of the debtor, lead-times between key events of the case and simple demographics.

While even the best model provided relatively lacklustre performance statistics, the efficiency gain presented through the monetary outcome is very promising. The model can be used prioritise cases of debtor that have the ability to pay their debt, but are not willing to, and on the other hand identify debtor for whom the summons process is not the appropriate path.

The obtained model and results are a useful addition to predictive analytics in credit management processes, but there are many aspects that could be extended.

On the subject of available and influential there data seems to be potential for gain. The variables used in this research were mostly focused on the key financial aspects of a case, while there is a pile of unstructured data to be found in the original invoices and correspondence with a debtor. Clustering and text mining models could be used to create extra information to base classification models on.

Another topic of future research is the extension of modelling over the complete credit management chain, as there are many similar decisions to be optimised. Being able to identify the ability and willingness of a debtor to pay off their debts is information that is relevant to all phases of the collection process. In the end, models can be combined to improve the accuracy of financial evaluations and decrease risks for consumers, clients and credit management companies.

7 Appendix

The appendix contains information that is deemed to take up too much space in the main text. It features the tools that are used in this research and details on the implementation of the analysis, as well as tables, figures and information on sources that are referred to in the main text.

7.1 Tools and Implementation

The technical implementation of this research was done in various SAS programs built for data management and analysis. SAS Data Integration Studio was used to create the data set, while the models were trained and initially evaluated using SAS Enterprise Miner. Figures and post mining evaluations were created using SAS Visual Analytics on SAS Viya.

Figure 4 shows the process flow implemented in SAS Enterprise Miner, that is used to create the model behind the summons score. It contains the following steps:

1. Load the data;
2. Drop columns unnecessary for data mining, but are useful for identification;
3. Create a training, validation and test set from the raw data set;
4. Create additional variables to treat outliers, standardise data and group rare categories;
5. Model training;
6. Model comparison;
7. Score test set and export data.

The ordering of the models corresponds to the ordering in this section, where Gradient Boosting is a SAS Enterprise Miner implementation of Boosting Classification Trees. The acronym *HP* stands for High Performance; these nodes were created to handle large amounts of data through parallellisation.

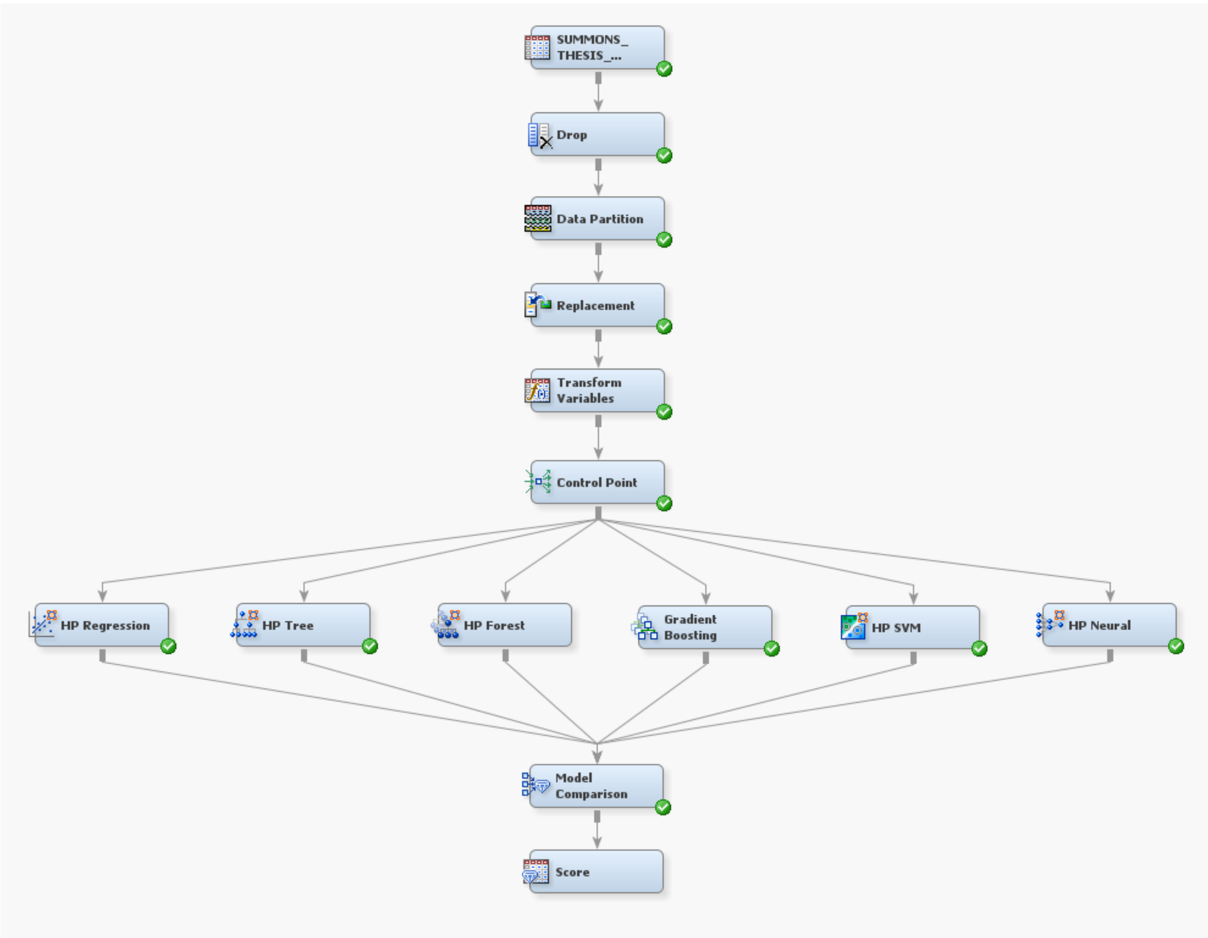

Figure 4: Figure containing the graphical representation of the process flow in SAS Enterprise Miner.

7.2 Tables

Column name	Description
Client_Type	Type of arrangement with DirectPay
Credit Score	Credit score on address level.
DebtorAge	Age of the Debtor (in years)
Market	Industry of client: Energy, Telecom, Retail et cetera.
CAT_BankAccountNumber	Category of bankaccount number based on availability of IBAN / pre-IBAN reference.
CAT_Email	Category of email extension.
CAT_PhoneNumber	Category of phone number.
DATE_Diff_Cession_Invoice	Amount of days between cession date and the oldest invoice date.
DATE_Diff_Summons_Cession	Amount of days between summons date and cession date.
FIN_Diff_Nominal_Invoices	Difference between remaining nominal amount of the case at time of Summons and the invoice amount.
FIN_Diff_Summons_Nominal	Difference between the amount on the summons and the nominal amount.
FIN_Invoices	(Nominal) Value of the original invoices.
HIST_AmountPaid	Amount paid by this Debtor in previous cases.
HIST_Cases	Amount of cases of this Debtor with Directpay.
HIST_CasesConfiscation	Amount of cases of this Debtor that got to the Confiscation phase.
HIST_CasesCounseling	Amount of cases of this Debtor where Debt Counseling was involved.
HIST_CasesFraud	Amount of cases of this Debtor that involved fraud.
HIST_CasesIrrecoverable	Amount of cases of this Debtor that were deemed irrecoverable.
HIST_CasesPaid	Amount of cases of this Debtor that were fully paid.
HIST_CasesWardship	Amount of cases of this Debtor where the Debtor was put under Wardship.
HIST_CasesWSNP	Amount of cases of this Debtor where the Debtor resorted to a Legal Debt Restructuring Program.
HIST_NumberPayments	Number of payments by this Debtor in previous cases.
HOUSE_Gebruiksdoel	Intended use of property.
HOUSE_MXbouwjr	Construction date of the building.
HOUSE_MXCTHT	Consumer type.
HOUSE_MXinhoud	Volume (indoors) of property.
HOUSE_MXkoophuur	Rental or property.
HOUSE_MXOppvl	Categorized surface area of property.
HOUSE_MXperccode	Property type.
HOUSE_MXwoning	Home type.
HOUSE_MXwoz	Estimated worth of property.
HOUSE_Oppervlakte	Surface area of property.
HOUSE_Status	Use status of property.

Table 4: Table containing explanatory variables and their descriptions. CAT standing for categorised variables, DATE for dates, FIN for financial information, Diff for differences, HIST for historical information concerning the debtor and HOUSE for address level information.

Kernel	$K(x_i, x_j)$
Linear	${x_i}^T x_j$
Polynomial	$(\gamma {x_i}^T x_j + r)^d$
Radial Basis Function	$\exp\{-\gamma \|\|{x_i}^T - x_j\|\|^2\}$
Sigmoid	$\tanh\{\gamma {x_i}^T x_j + r\}$

Table 5: Examples of widely used kernels

Bin probabilities	Amount of cases	Average expected value (in €)
0.20	10	-175.00
0.25	16	-123.93
0.30	47	4.15
0.35	336	4.76
0.40	1389	37.82
0.45	2929	70.57
0.50	2831	102.20
0.55	2105	128.95
0.60	1329	158.03
0.65	562	183.71
0.70	192	205.05
0.75	53	223.77
0.80	6	210.65

Table 6: Option one: the binned probabilities obtained from the Random Forest model divided into intervals of 0.05, including amount of cases in the bin and the average expected value. The first bin contains all cases with an estimated probability up to 0.20, the second those between 0.20 and 0.25, et cetera.

Bin	Amount of cases	Average expected value (in €)
1	1180	13.03
2	1181	50.41
3	1180	61.65
4	1181	93.21
5	1180	103.29
6	1181	94.18
7	1180	117.04
8	1181	130.19
9	1180	146.41
10	1181	193.34

Table 7: Option two: the summons score as probabilities obtained from the Random Forest model divided into bins of equal sizes, including amount of cases in the bin and the average expected value.

7.3 Figures

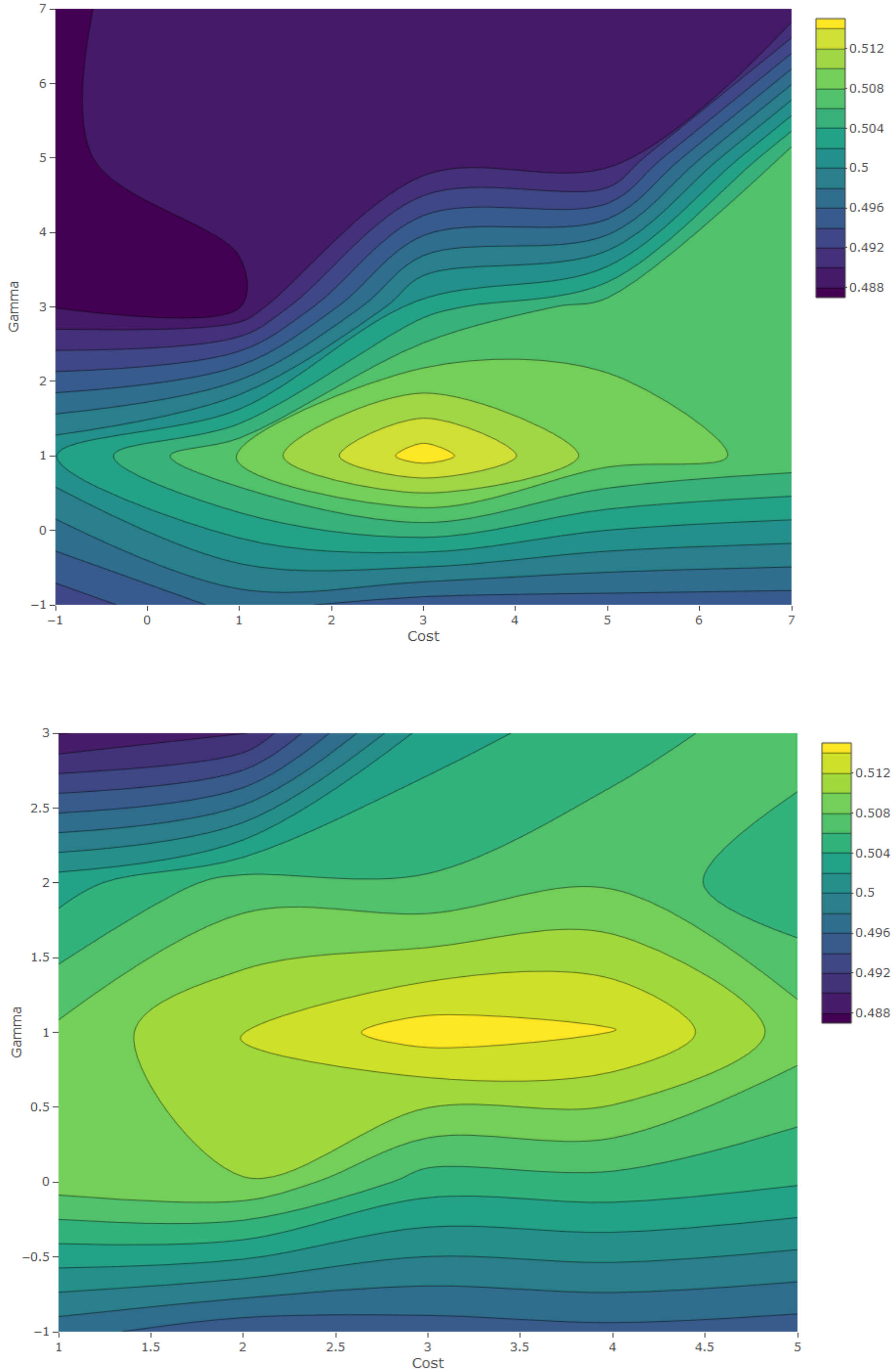

Figure 5: Grid search for the SVM model with RBF kernel on intervals $C = 2^{-1}, 2^{1}, \ldots, 2^{7}$; $\gamma = 2^{-1}, 2^{1}, \ldots, 2^{7}$ and $C = 2^{1}, 2^{2}, \ldots, 2^{5}$; $\gamma = 2^{-1}, 2^{0}, \ldots, 2^{3}$. Displayed values are test set accuracies.

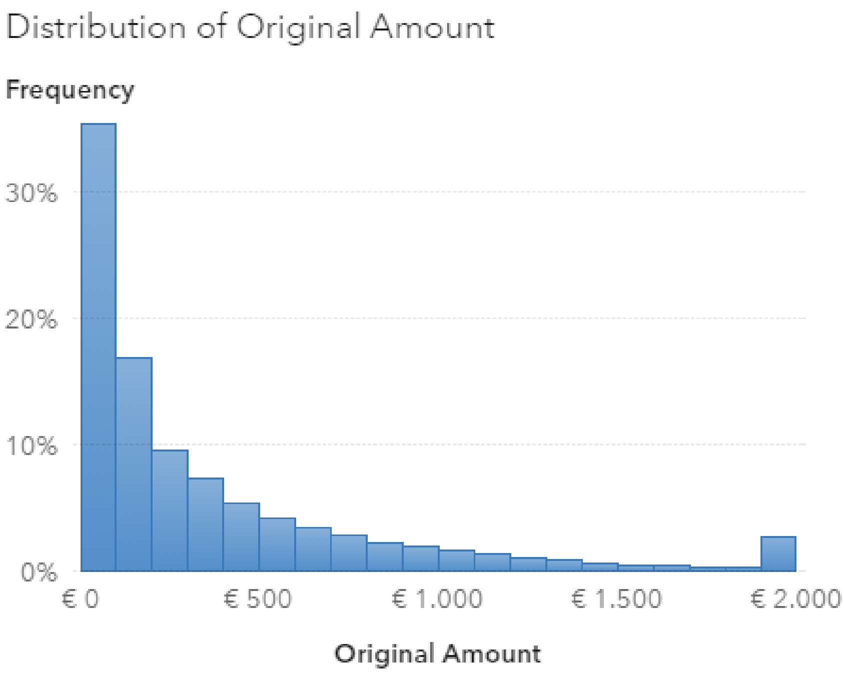

Figure 6: Histogram of the original amount; the open amount at time of summons. Values above €2000 are censored and placed inside the category €1950 - €2000

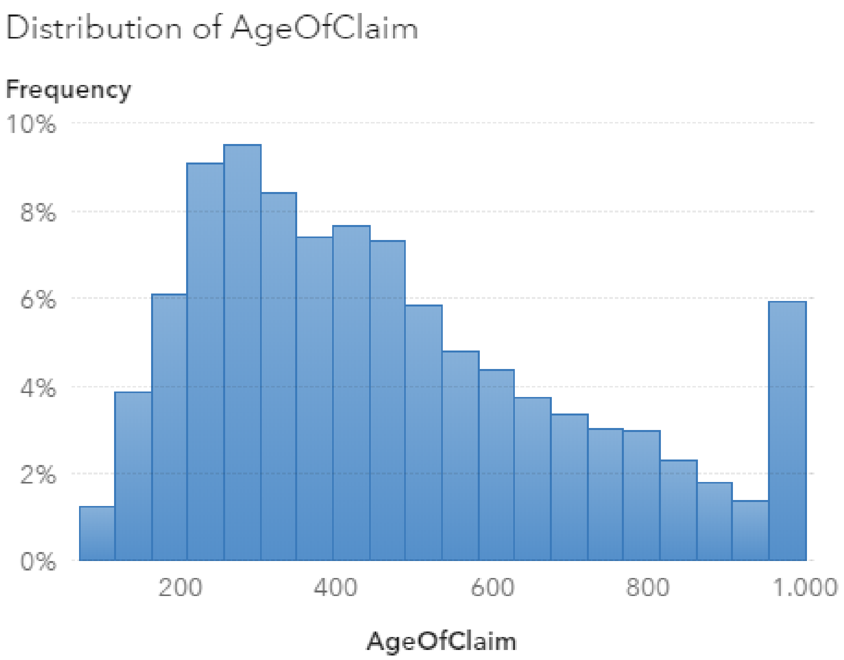

Figure 7: Histogram of the age of the claim at time of summons, in days. Values above 1000 days are censored and placed inside the category 950 - 1000 days

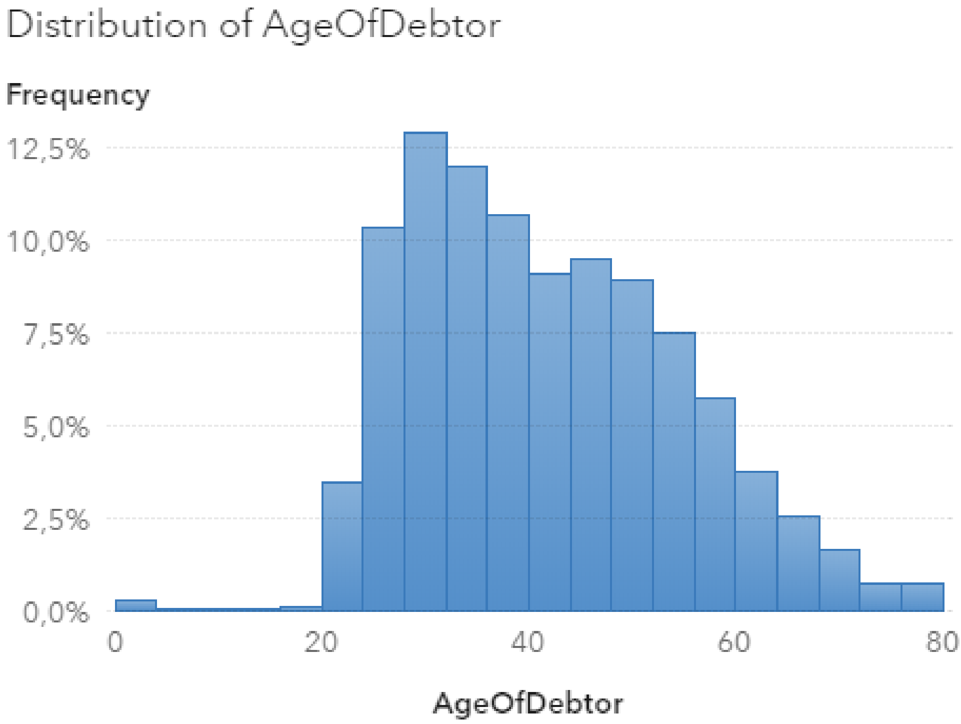

Figure 8: Histogram of the age of the debtor at time of summons, in years. Values above 80 years are censored and placed inside the category 76 - 80 years, while values below 16 are placed inside the category 0 - 4 years. Due to privacy regulations, information of minors under 16 is not allowed to be used for this research.

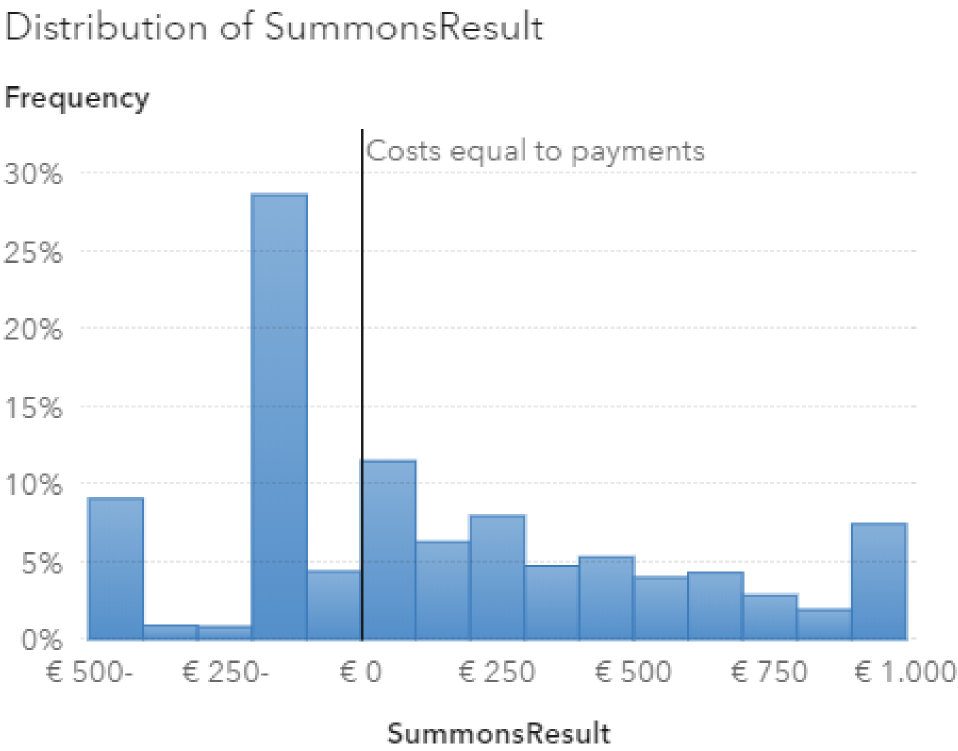

Figure 9: Histogram of the target variable of thesis: the Summons Result. Values above €1000 are censored and placed inside the category €900 - €1000.

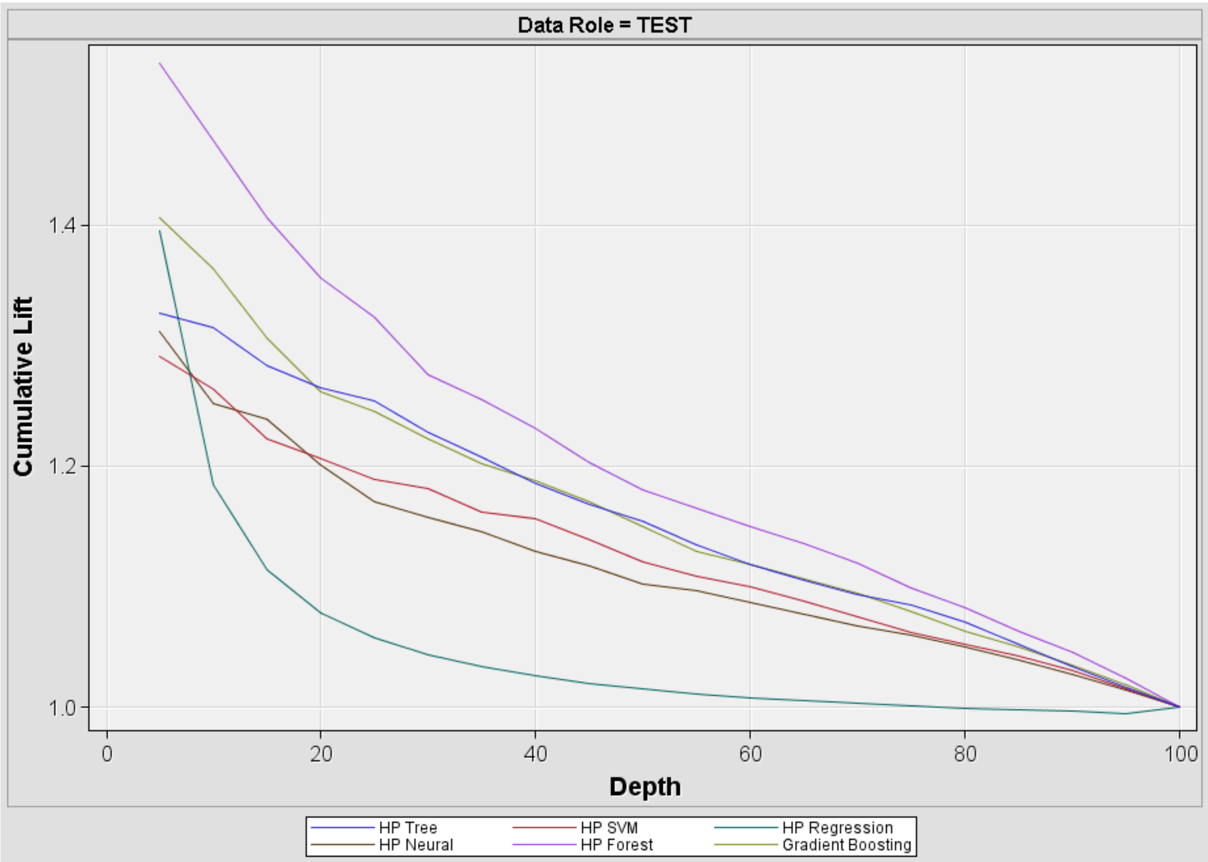

Figure 10: Figure containing the gain graph of the models in the Model Comparison node in SAS Enterprise Miner.

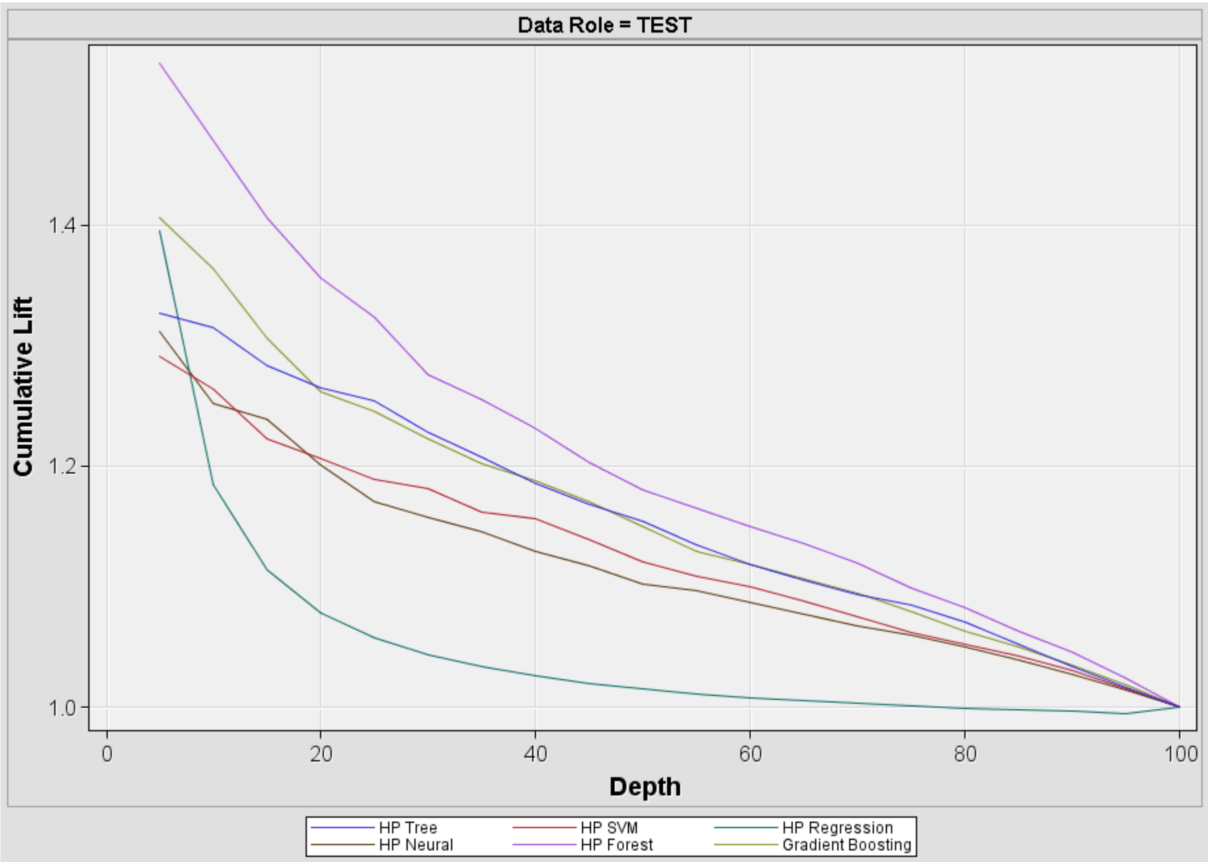

Figure 11: Figure containing the lift graph of the models in the Model Comparison node in SAS Enterprise Miner.

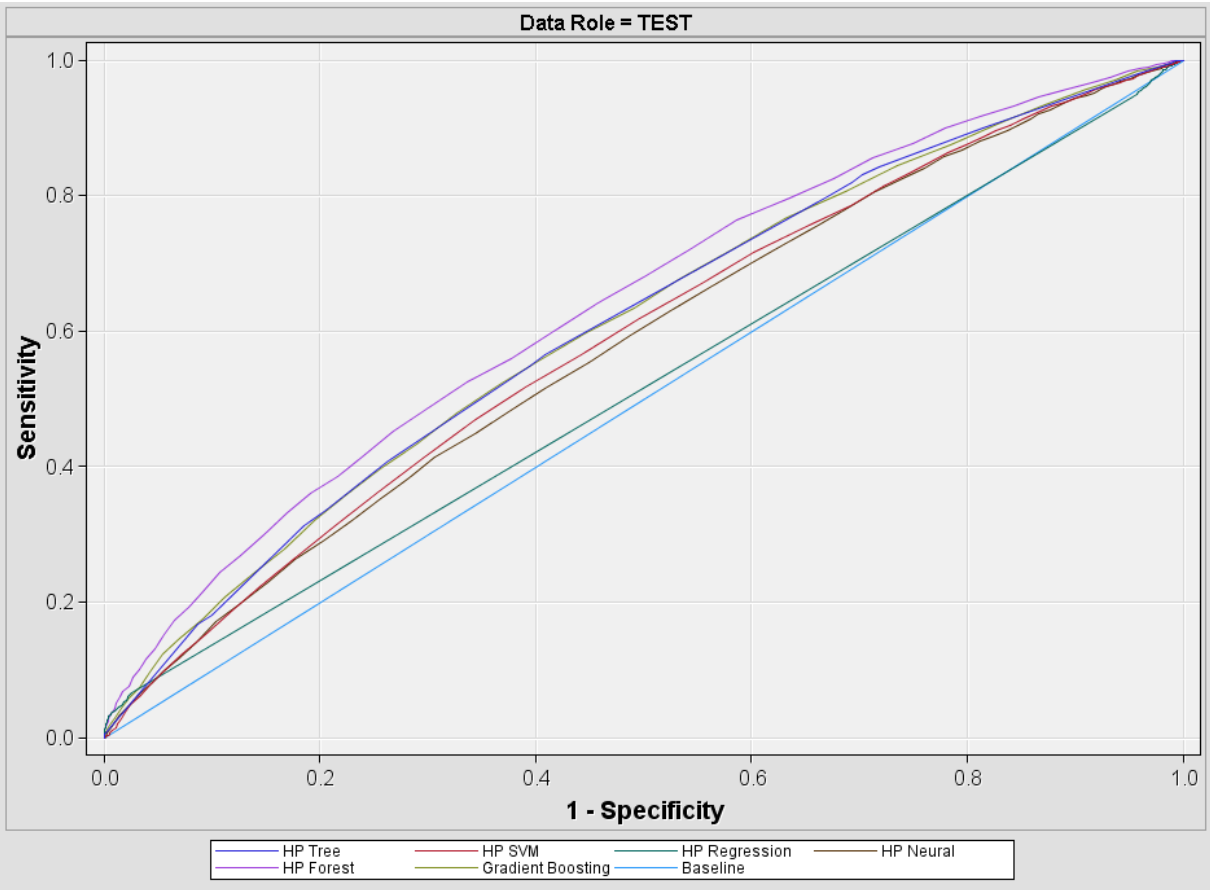

Figure 12: Figure containing the ROC curves of the models in the Model Comparison node in SAS Enterprise Miner.

References

ABN AMRO (2016). Markt van buitengerechtelijke incasso. https://insights.abnamro.nl/2016/07/branche-update-deurwaarders-en-incassobureaus/ Last accessed 2018-08-20.

Baesens, B., Lessman, S., Seow, H.-V., and Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, 247:124–136.

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pages 144–152, New York, NY, USA. ACM Press.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24:123–140.

Breiman, L. (2001). Random forests. *Machine Learning*, 45:5–32.

Breiman, L. and Cutler, A. (2003). Manual—setting up, using, and understanding random forests v4.0.

Cortes, C. and Vapnik, V. N. (1995). Support-vector networks. *Machine Learning*, 20:273–297.

Desai, V. S., Crook, J. N., and Overstreet, G. A. (1996). A comparison of neural networks and linear scoring models in the credit union environment. *European Journal of Operational Research*, 95:24–37.

DirectPay (2014). Unpublished internal document.

Dutch Civil Code (2012). Besluit vergoeding voor buitengerechtelijke incassokosten. http://wetten.overheid.nl/BWBR0031432/2012-07-01 Last accessed 2018-06-14.

Dutch Civil Code (2018a). Besluit tarieven ambtshandelingen gerechtsdeurwaarders. http://wetten.overheid.nl/BWBR0012638/2018-02-17 Last accessed 2018-06-14.

Dutch Civil Code (2018b). Wet griffierechten burgerlijke zaken. http://wetten.overheid.nl/BWBR0028899/2018-01-01 Last accessed 2018-06-14.

Freund, Y. and Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*, ICML'96, pages 148–156, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Friedman, J., Hastie, T., and Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting. *The Annals of Statistics*, 28(2):337–407.

Friedman, J. and Meulman, J. (2003). Multiple additive regression trees with application in epidemiology. *Statistics in Medicine*, 22(9):1365–1381.

Geurts, T. (2012). Markt van buitengerechtelijke incasso. https://www.wodc.nl/onderzoeksdatabase/incassomarkt-in-kaart-brengen.aspx Last accessed 2018-08-20.

Hanley, J. A. and McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA.

Hoff, S., Wildeboer Schut, J. M., Goderis, B., and Vrooman, C. (2016). Armoede in kaart. Technical report, Sociaal en Cultureel Planbureau (SCP), Den Haag, The Netherlands. https://goo.gl/KvM6sJ Last accessed 2018-06-14.

Huang, C.-L., Chen, M.-C., and Wang, C.-J. (2007). Using neural network ensembles for bankruptcy prediction and credit scoring. *Expert Systems with Applications*, 33:847–856.

KBvG (2018). Jaarverslag 2017. https://www.kbvg.nl/2338/jaarverslag under 'Jaarverslag 2018'. Last accessed 2018-08-20.

Maldonado, S., Bravob, C., Lópezc, J., and Péreza, J. (2017). Integrated framework for profit-based feature selection and svm classification in credit scoring. *Decision Support Systems*, 104:113–121.

Mozer, M. C., Wolniewicz, R., Grimes, D. B., Johnson, E., and Kaushansky, H. (2000). Predicting subscriber dissatisfaction and improving retention in the wireless telecommunications industry. *IEEE Transactions on Neural Networks*, 11(3):690–696.

Nielsen, M. (2017). Neural networks and deep learning. http://neuralnetworksanddeeplearning.com/chap1.html Last accessed 2018-06-14.

NVI (2018). Kerncijfers. https://www.nvio.nl/kerncijfers-nvi-2015 Last accessed 2018-08-20.

Shapire, R. E. (1990). The strength of weak learnability. *Machine Learning*, 5:197–227.

Tsai, C.-F. and Wu, J.-W. (2008). Using neural network ensembles for bankruptcy prediction and credit scoring. *Expert Systems with Applications*, 34:2639–2649.

van Putten, B. and Schoot Uiterkamp, T. (2017). Schuldhulpverlening in nederland. Technical report, KWIZ, commissioned by the Dutch Ministry of Social Affairs and Employment, Groningen, The Netherlands. https://goo.gl/nyLcKk Last accessed 2018-08-28.

Wang, G., Hao, J., Ma, J., and Jiang, H. (2011). A comparative assessment of ensemble learning for credit scoring. *Expert Systems with Applications*, 38:223–230.

West, D. (2000). Neural network credit scoring models. *Computers & Operations Research*, 27:1131–1152.

Westhof, F., de Ruig, L., and Kerckhaert, A. (2015). Huishoudens in de rode cijfers. Technical report, Panteia, commissioned by the Dutch Ministry of Social Affairs and Employment, Zoetermeer, The Netherlands. https://goo.gl/rG87v8 Last accessed 2018-06-14.