A model of priority in loan applications based on expected profitability

van der Kroef, M.

Award date:
2020

A Model of Priority in Loan Applications based on Expected Profitability

Master Thesis – **Final Version**

20th of April 2020

Student **M**. (**Maud**) **van der Kroef**, **BSc**

Master Operations Management & Logistics | Department Industrial Engineering & Innovation Sciences | Technische Universiteit Eindhoven

Student number | 0862548

Student mail | m.v.d.kroef@student.tue.nl Private mail | maudvanderkroef@me.com

---

First Supervisor University

Dr. S.S. **Dabadghao**

Mail | s.dabadghao@tue.nl

Department Industrial Engineering & Innovation Sciences | Operations Planning Account & Control

Second Supervisor University

Dr. M. **Slikker**

Mail | m.slikker@tue.nl

Department Industrial Engineering & Innovation Sciences | Operations Planning Account & Control

Third Assessor University

Dr. R.P. Ranjan

Mail | r.p.ranjan@tue.nl

Department Industrial Engineering & Innovation Sciences | Operations Planning Account & Control

Company Supervisor I

H.T.W. (Harm) **Hoebergen**

COO at CompX

Mail | harm.hoebergen@CompY.nl

Company Supervisor II

T.A.A. (Tom) **Jans**

CFRO at CompX

Mail | tom.jans@CompY.nl

Keywords

BigML; Classification Model; Consumer Finance; Credit Scoring; Decision Forest; Decision Tree, Expected Profit Loans; Loan Application; Machine Learning; Personal Loan; Priority Rule; Regression; Repay Behavior

Management Summary

The objective of this research is to improve the application process at CompX by generating more insight into the profitability of loans. The aim is to model this profitability in order to predict it based on characteristics of an application that are known as soon as it arrives. The main research question is:

How to model the expected profitability and how can this knowledge be used at the application process?

The first main finding of this research is the definition of profitability of a loan. There are many costs concerned with approving and maintaining a loan, but it is not easy to assign all these costs to one contract. For the purpose of this analysis, the profit should serve in comparing loans. So, all costs that are the same for all loans are in that sense not relevant. The only costs that do differ per loan are the risk costs and the commission costs. The risk costs are the costs for the expected loss of a loan, also referred to as provision. Having a probability of a loan, some money should be set aside, so-called provision, to protect the potential financial loss. This implies this money cannot be lend and no money can be earned over this money. The average provision of a loan has an opportunity cost of 17% per year (3,17% per month), which are the risk costs. The commission costs are the costs associated with third parties who led the customer to CompX. The revenue is the sum of all interest parts. So, in conclusion, the formula for profit looks like:

$$profit = \sum_{m=1}^{ad} interest\ paid_m - 1{,}317\% * \sum_{m=1}^{ad} provision_m - \sum_{m=1}^{ad} commission_m$$

where:
$ad = the\ actual\ duration\ in\ months$
$m = the\ month\ of\ the\ payment\ where\ m = 1\ at\ the\ month\ of\ the\ first\ repayment$

Two more side notes are important to this formula. Firstly, the interest paid is highly dependent on the interest rate the customer pays. This interest rate has changed over time, but this should not influence the outcome of the model. Therefore, the interest paid is recalculated in order to equal the interest rate. Secondly, though the term profit is used, this cannot be fully interpreted as the profit since there are still other costs such as overhead costs and marketing costs that should be covered; and the margin taking into account the funding costs.

The next step of the research consists of the data preparation in order to find a model that can predict profit. Input variables, that are available when the application arrives, are needed to do so. Many variables were evaluated in order to investigate relations with profit. The objective is to train a model that will predict the profit based on the input variables selected. Multiple machine learning techniques are applied to find a predictive model. Techniques used were decision trees, decision forests and linear or logistics regressions. In addition, multiple output measures were tried: predicting profit as a continuous value; predicting profit in three classes; and predicting profit in five classes. About 300 different models are evaluated. The best model found contains five classes and predicts the profit based on three input variables: the loan volume of the contract, the sum of the interest that will be paid eventually and the monthly

payment that should be executed according to the contract. The method used for this model is the decision forest. The model assigns an application to one of the five classes, which can be seen as priority labels. Label A represents the highest class, label E represents lowest class. The average profit of each label and the ranges associated with each label, is shown in Table MS1.

Table MS1. *5 classes of priority, their ranges and average profit*

Class	y_3/Y3	**Average profit (per bucket) | number of contracts**
A	> €2000	€2982
B	[€1000 , €2000)	€1393
C	[€600 , €1000)	€783
D	[€400 , €600)	€497
E	< € 400	€209

The last step of the analysis is to examine the application of this model into the process of CompX. In collaboration with the MT and another test that ran during this thesis, the preference is to experiment with this rule by adding a new step into the process. Currently, the applicant has to return the offer in 30 days after receival, but is not called if there is no response in these 30 days. The pilot showed that calling applicants that received an offer, improved the return rate from 75,2% to about 79%. As a follow-up on this test, it is suggested to implement the priority label in this step. A calling scheme is suggested to call the applicants after they did not respond in 7 days. If the offer is sent at $t = 0$, the applicant will be called at $t = 7$, but if not responds, also at $t = 10$, $t = 15$, $t = 20$, $t = 25$. At most five calls are executed, but the applicant is only called again when there is no response in the meantime. This process is simulated with an A/B test that consisted of two groups: Group A calls the applications randomly, Group B calls the applications with highest priority. The total workload of applicants that need the first call is equally divided over these two groups. There is a capacity constraint of 6 new applications, i.e. first call, per day per group. From a simulation of 1 year, it appeared Group B's expected profitability was 25% higher than the expected profitability of Group A.

In conclusion of these results, it is advised to try the A/B test in real life for a longer duration, at least 6 months to see the effects of some seasonal trends as well. Unfortunately, time is needed here because the repay behavior cannot be examined before the loan is repaid, thus this outcome is not a result of this research. Other suggestions for implementation are to create a fast lane for A instances at each step of the application process; to implement the priority rule at the phase of the Check by Call (see Figure 4 at page 9); or implement the rule at Risk Assessment only.

To summarize these findings, profitability with the objective of comparison loans at arrival only need inclusion of summed interest paid, risk costs and commission costs. A predictive model is found using input variables loan volume, contractual interest and the monthly payment. A simulation shows the application of the priority rule at a new process step Call after Offer Sent suggests improvement in expected profitability, which shows potential for the priority rule. Therefore, it is advised to execute the A/B test in practice.

Preface

This thesis serves as the last missing puzzle piece of my study at TU Eindhoven. A university I've been studying at for almost 7 years now. On the one hand, I'm looking forward to the challenges and new experiences the working life will give; and will most likely not miss the exams and stressful deadlines. On the other hand, I'll miss the freedom study life gives, the friends I was able to see on a daily basis and the continuous stream of new knowledge.

I'm also very glad looking back at the time I executed this last piece. I've struggled with ups and downs, but luckily I've been blessed with the supportive people around me. First of all, I would like to thank CompX for giving me the opportunity to return to the 'CompX Family', where I already had good working experience as a service agent at the call center. The openness and supportiveness of Harm en Tom have made me feel welcome again, but most importantly gave me the opportunity to develop myself and my thesis research. I would like to thank you both for the fact you were always available for questions, brainstorms and providence of your opinions; and for the provided guidance and control when needed. In addition, I would like to thank my colleagues for the nice lunch breaks, inspirational conversations and pep-talks. And especially, Redmar, for learning me new (and sometimes frustrating) skills, such as working with T-SQL and PowerBI, for brainstorming with me and for giving your insights and opinon.

Besides the company, I would like to thank the guidance of my university. After the unforeseen bumpy start, I'm happy with the fact Shaunak Dabadghao was willing to be my first mentor. Your enthusiasm at the beginning of the project was inspirational and motivational. Also, the feedback and (Skype) meetings were useful for further progress, though it seemed an endless road from time to time. In addition, I would like to thank my second mentor, Marco Slikker, for your critical view on my thesis, that stimulated me to rethink and look at things in multiple perspectives.

Unfortunately, I'll not be able to thank you all in person due to the highly unusual circumstances of the COVID-19 virus regulations. It feels at bit weird to have been working together for over 6 months without a formal closure, but maybe our paths will cross in future.

Before I wish you a joyful time reading my thesis, I would like to thank a few more people: starting with my parents for their support through my whole study ánd my thesis period. And last but not least, my boyfriend, sister and friends for the mental support during this period and their never ending belief in me.

Maud van der Kroef

20 April 2020

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Full meaning
AFM	Authority of Financial Markets (Dutch: 'Autoriteit Financiële Markten')
BKR	Office of Credit Registration (Dutch: 'Bureau Krediet Registratie')
DF	Decision Forest
DK	Revolving Credit (Dutch: 'Doorlopend Krediet')
DT	Decision Tree
DurationA	Actual Duration
DurationC	ContractualDuration
EAD	Exposure at Default
EL	Expected Loss
EoY	End of Year
FCFS	First-Come-First-Serve
FN	False Negative
FP	False Positive
ITO	Internal Take Over (Dutch: 'Interne Oversluiting')
KCC	Klant Contact Center
LGD	Loss Given Default
ML	Machine Learning
MLC	Maximum Loan Capacity
MT	Management Team
NBV	New Business Volume
(D)NN	(Deep) Neural Network
PD	Probability of Default
PL	Personal Loan
TN	True Negative
TP	True Positive

Glossary

Term	Explanation
Applicant	The person who applied for a loan. The applicant is linked to an application. As soon as the application is accepted and the applicant receives the loan, the applicant is not an applicant anymore but a customer.
Application	Any filled-in form by a customer that is received by CompX. As long as it is in the application process flow and is not (yet) an activated contract, it is referred to as an application, owned by an applicant or two applicants.
Application Process	The process that an applicant has to follow in order to receive a loan. Figure 4 is a visualization of this process
Approval Conversion Rate	Refers to the percentage that is accepted, after passing the system check
Contract/Loan	An activated and approved application. It implies an outstanding debt to CompX.
Customer	A customer is a person having a loan at CompX. Thus, if a person does not have a contract yet, but only an application, it is not referred to as a customer.
Lead	A lead is a digital 'path' that leads to an application. For example, a lead is generated by a comparison site that brings an applicant to CompX.
Loan Volume	This is the original net debt of the customer; the money the customer receives when the contract is paid out.
Marketing Conversion Rate	Refers to the percentage that applies for a loan from the people who clicked on an online advertisement of CompX

1. Introduction

CompX is an online financial institution operating on the private consumer market, since the year of 2006. It provides financial products via internet or phone and limits these to only two type of products: a personal loan (‘*Persoonlijke Lening*’; PL) and a revolving credit (‘*Doorlopend Krediet’;* DK). The fact the company is an online institution implies there is no physical office where customers can go to, but all online services are provided from one office located in Eindhoven. This also reveals a part of the strategy of CompX: it wants to offer a competitive price by claiming a place in the top 3 on the market of each product type. This is achieved by having low costs, such as a low operational costs with just one digital office, and low risk costs by assessing applications in a strict manner with high cut-off levels on potential risk of default.

The portfolio of CompX increased rapidly over the last 7 years, as visualized in Figure 1. The portfolio refers to the sum of all outstanding debts of customers, measured at the end of that year (EoY). The line of active contracts refers to the count of all outstanding contracts at that time.

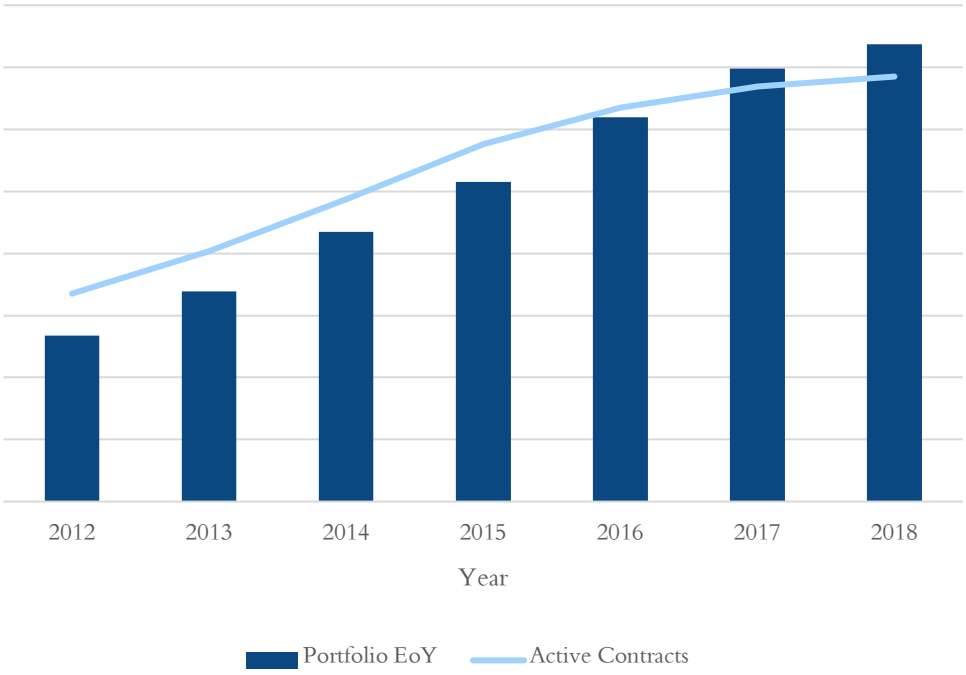


Figure 1. Growth in portfolio of CompX between 2012 and 2018.

As the figure also shows, the size of the growth per year decreased the last few years. The portfolio consists of loans with an original loan volume varying between €5.000 and €75.000. One of the key characteristics of loans at CompX is the flexibility in repaying. A customer can repay the loan, the way he or she wants with the benefit interest future lapses and without paying any fee. As a consequence, the revenue of a loan becomes unknown beforehand. Now, taken notice of this decreasing growth and the uncertainty of the repay schedule, the next Section will dive deeper into the problem CompX is facing.

1.1 Problem Definition

The total portfolio of CompX generates profit year by year. However, there is no insight into how profitable each loan in specific is, but there is a gut sense some loans might not be profitable at all. So, first of all, the management team (MT) would like to have more insight into how profitable each contract has been or, in other words, what the contribution of that loan is to the company's profit and how this can be determined. This knowledge can be useful to the marketing strategy and product range strategy in later stages.

Furthermore, the problem origins from the capacity of the application process. CompX has reached the maximum capacity for departments that assess an application in terms of risk and decide whether an application becomes a contract. At peak times, throughput times rise and workload exceeds the maximum work capacity. In addition, the reachability of an applicant is influencing the workload highly. Both facts urge for more efficiency in terms of revenue. Therefore, CompX strives to have a new guideline, (priority) rule or cut-off level, or new process step to decide which application to put in most effort and how; and in which not, or less. The objective of the company is to gain most profit from the capacity and the demand it currently has, without direct exclusion of certain loan application types.

However, the underlying problem is that the profitability of a loan is not known beforehand. Thus, a decision about effort based on profitability is not made easily. This has to do with one of the most important product conditions of a loan at CompX: a customer can repay the loan any way he or she wants. Irrespective of the agreed contract. The interest is paid over the net debt, which might differ from contract in case of extra repayments. This leads to an unpredictable revenue from contracts. The MT would like to use some model of expected profitability to assist the decision for applications at the application process and prior applications differently than the current application handling strategy: First-Come-First-Serve (FCFS).

More explanation about the product type and the process flow of the operation will be given in Chapter 2.

1.2 Research Objective and Research Questions

Based on the above description concerning the problem CompX is facing, the objective is to deliver deeper understanding of an application and the repayment behavior. A model of expected profitability that determines a ranking or priority in applications' profit contribution; and an extensive analysis on how to implement this knowledge in the application process should serve in this. Therefore, the main research question of this thesis is:

How to model the expected profitability and how can this knowledge be used at the application process?

To reach this goal step-by-step, some sub-research questions have been made.

SQ1. How to determine the profitability of an application?

SQ1.1 What are the different cost aspects and how to assign these to a contract?

SQ1.2 How to calculate the revenue of a contract?

SQ1.3 Which characteristics does an application have?

SQ1.4 Which characteristics does a contract have?

SQ2. What is the as–is situation?

SQ2.1 How does the portfolio look like in terms of profitability?

SQ2.2 Which characteristics of an application can be predictive for the profitability?

SQ2.3 Which product groups can be formed to analyze differences?

SQ2.4 How do these predictive characteristics perform on the portfolio?

SQ3. How to improve application handling by changing priority or implementing new guidelines?

SQ3.1 Which type(s) of applications are less profitable?

SQ3.2 With what rules can the application process be improved using insights of expected profitability and how do these perform?

SQ3.3 How can CompX use an expected profitability model to improve its application process?

SQ3.4 What experiment should CompX do to test the proposed improvements?

1.2.1 Scope

This research focusses on PLs only, DKs are excluded. There are three main reasons for this choice. Firstly, due to time restriction it is not possible to take both type of loans into consideration. Because of the difference in characteristics, the same method for data interpretation and preparation cannot be applied to both products, as where the method for DKs is less straightforward. Secondly, the product conditions of DKs have recently changed which has large impact on repay possibilities of customers and analyzing the profit of old DKs is not comparable to the expected profitability of new DKs. The differences will be elaborated on in Section 2.1.1 and 2.1.2, but shortly said new DKs will be more comparable to PLs. Thirdly, DKs become a less popular product as where the majority of the contracts is a PL; and this share of PLs has been increased over the past. This trend is visualized in Figure 2. Due to regulation and restrictions from supervisory institutions, MT even expect DKs might disappear on the long term.

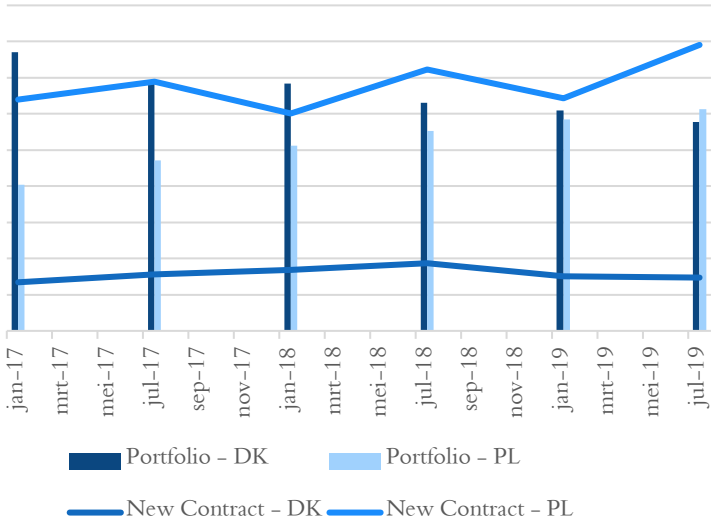


Figure 2. Amount of DK versus PL contracts in portfolio and new contracts.

Another important scope restriction is based on the origin of the loan. There has to be an application of the loan to analyze and therefore taken over portfolios are not taken into account for this analysis.

This research started in September 2019. Only data until the end of September 2019 is included in this research. Thus, new contracts after that period are not included, neither transactions nor events for running contracts.

Also, out of scope are all aspects related to funding. Funding is not fully in hands of CompX, but is done in cooperation with CompY. Next, funding is done based on portfolio level, not on contract level. The interest paid over funding is the same for each contract, which is close to 0 currently[1]. Large changes in these funding costs affect the whole market, and will increase the interest rate for the customer on the whole market. Since the effect will be the same for the whole portfolio, it is assumed not to be relevant when comparing loans.

1.3 Research Methodology

The model of Mitroff (Mitroff & Sagasti, 1973) is the guideline for the research methodology. Figure 3 represents the visualization of the method.

[1] In Q3 of 2019, the interest rate for funding was even negative. In Q4, it was very close to zero.

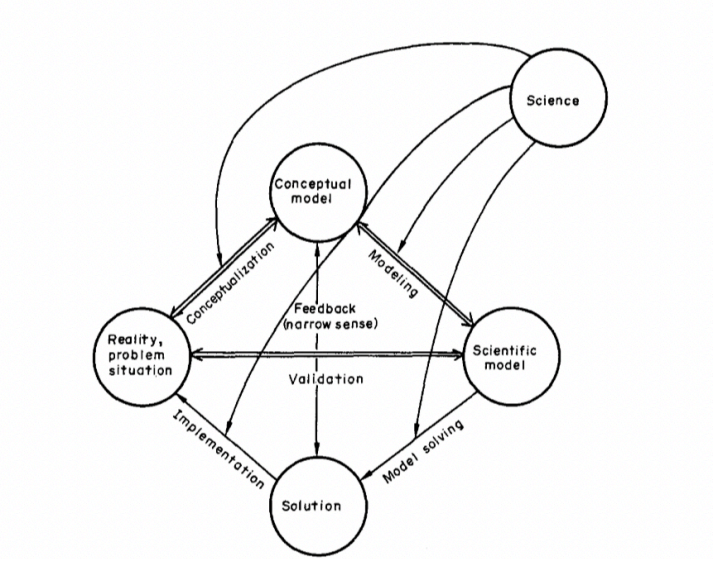


Figure 3. System view to operations research process (Mitroff & Sagasti, 1973).

The model is usually applied to researches in operations, though the company does not handle physical products, it is still applicable due to the operational process of CompX. The figure represents four circles in the shape of a diamond where each circle represent one of the four subsystems. The Reality, Problem Situation refers to the phase CompX currently is facing, and the phase where the problem CompX is facing has been determined. This phase involves learning and experiencing without direct result, thus will not be directly visible in this thesis. According to Mitroff & Sagasti (1973) the most important phase is the conceptualization between the Reality subsystem and the Conceptual model subsystem. It evolves a mental image of the situation. The formalized representation is part of the subsystem Scientific Model. It enables the analysis of the problem, the creation of a solution and testing it. Of course, as in every scientific research, other scientific research provides input for all the phases of this research. However, an extensive literature review has been done separately from this thesis research, from which parts are used. The literature review was more solely focused on market development and credit scoring models and is summarized in Section 2.5.

As shown by the arrows within the diamond cycle, the process is iterative. In this research the Conceptual subsystem and Scientific Model subsystem can take place in the data preparation phase and all choices that come with data selection. The analysis of this data and the reflection to or validation on the process of CompX can be seen as the steps around subsystem Solution. The Conceptual subsystem and the steps connected it, turned out to be critical, and very time-consuming during this research. This subsystem determines the scope and the level of applicability of the outcome of the research, therefore it is worth the time. In short, this model determines the input and framework of the research and assists scoping while the company is a broad concept with many factors involved.

1.4 Software Tools

Besides this general methodology, some software tools have been used. To access the data in the data warehouse, the software of Microsoft Server Management Data Studio is used. All the tables are stored in this software and contain information about applications and contracts. The data is called using Transact-SQL . For further data calculations and aggregations, Microsoft Excel and Anaconda Spyder with Python are used. For the basic analysis and descriptive statistics, Microsoft Excel and Microsoft Power BI were sufficient software applications. For the creation of a model and validation, BigML is used, which is an online platform for machine learning (ML) using multiple techniques, that is licensed by CompY. For validation of the predictive model and the application in the process, also Excel appeared to be sufficient.

1.5 Thesis Outline

Chapter 1 aimed to give an explanation of what the aim of this research is, and where and how it is conducted. Next, Chapter 2 aims to give a deeper insight into product types, process flows, the definition of profit and cost allocation, and some literature that has been analyzed before and during this research. Chapter 3 will give some details about the data preparation, the input and output variables of the model and decisions made to come up with the used data set. Chapter 4 will follow with the results of descriptive statistics and the results of construction of the predictive model. Chapter 5 will reflect on the application of the model and practice, and lastly, Chapter 6 will summarize, reflect and recommend to conclude the insights from the research.

2. Background and Understanding of the Market

This Chapter aims to give more details about the types of product CompX provides, the process flow that is key to their operation, the revenue and the costs associated to their business and some background information about the environment and market their operating in. It will end with a short literature analysis that gives a scientific base for the research.

2.1 Product Types and Repayment

As described earlier, there are only two types of products, namely PLs and DKs. The differences between these two are essential to understand the revenue model of CompX and will be explained in Section 2.1.1 and 2.1.2. However, both products follow the same principle: a certain amount of money is lent to the customer and has to be repaid by a minimum monthly fee. This fee consists of interest and repayment. Besides this obliged fee, a customer can always repay more. If the customer repays more than agreed by contract, it will result in a financial advantage for the customer. Namely, interest is calculated monthly over the net debt of that month. An extra payment creates a lower net debt for the next month. A lower net debt implies a lower interest paid eventually and a shorter duration.

2.1.1 Personal Loan

With a personal loan, the loan volume of the provided loan will be paid out directly to the customer at moment of activating the contract. The contractual duration is chosen beforehand and varies between 6 and 120[2] months. However, as described before, the customer can repay earlier than the contract agreed on. The interest rate depends on the loan volume, is not risk-based, and will be fixed for that contract as long as it runs. There are five different ranges for this interest rate based on the loan volume, also referred to as buckets. The current[3] interest rates, set since May 2018, are shown in Table 1. The interest rate decreased over time, as can be seen in Appendix A.

Table 1. *Current interest rates of personal loans*

Loan Volume Bucket	**Yearly Effective Interest Rate (PL)**
€5.000 - €10.000	6,5%
€10.000 - €15.000	5,2%
€15.000 - €25.000	4,4%
€25.000 - €75.000	3,9%

Interest rates are always mentioned as a yearly effective interest rate. However, they are passed on to the customer on monthly base. The monthly interest rate can be calculated from the yearly rate by using the equation:

[2] In the past, some loans were contracted for a duration up till 180 months. This only happened under special circumstances where the mortgage debt was higher than the resale value of the house and this difference had to be financed. This condition does not exist anymore, however a duration longer than 120 months will appear in the data.

[3] The interest rate changed after September 2019, however this time period is out of scope for the research.

$$i_m = \left(1 + i_y\right)^{\frac{1}{12}} - 1$$

where:

$i_m = monthly\ interest\ rate$

$i_y = yearly\ interest\ rate$

The monthly interest rate is calculated over the net debt at the beginning of the month and this is the amount of interest paid per month.

It can occur a customer wants to borrow more money during the contract. This is a loan increase. The customer has to follow the application process again. If the new application is accepted, the new loan will have a new contract number. Administratively, the old loan is paid off with the new loan and now the customer has a new 'increased' loan with an interest rate, adjusted to the new loan volume bucket and the time. For further use in this report this type of action will be referred to as an 'internal take-over' (ITO). This is an important understanding to the analysis because looking solely at all contracts, cannot be equally linked to the duration of the customer at CompX. The previous contract seems to have a short duration, however the customer stays at CompX by increasing their loan.

2.1.2 Revolving Credit

For a revolving credit, the loan volume of the provided loan is fixed by the contract. However, this is not fully paid out by definition. The loan volume represents an upper bound of the customer's loan. As soon as the customer requests a payout, the monthly fee has to be paid independent from the level of the net debt. A customer can do multiple payouts as long as the total net debt does not exceed the original loan volume. After repayments, the unused loan volume can be paid out again. Therefore, this product is more flexible than a PL. The interest rate is not fixed, not risk-based and is also based on the loan volume and the corresponding bucket interest rate. There exists an old version and a new version of a DK. With the old version, the customer could use the loan volume until the age of 70 years, after that the credit limit decreased rapidly, such that the contract was ended by the age of about 75 years. With the new version, the credit limit will decrease after two years, such that the contract is ended in about 14 years at maximum.

However, as mentioned before, this type of loan is out of scope, because of the (1) other repayment possibilities and debt schedule; and (2) the fact there are multiple applications for one contract which makes the model for input characteristics too complex for this research' time span; and (3) the product is provided way less than PLs or is even expected to disappear by regulation of AFM (Authority Financial Markets-*Dutch: 'Autoriteit Financiele Markten'*) on the long term; and (4) with the new product conditions a DK becomes more comparable to PLs. Therefore, further details about this product type will not be given.

2.2 Process Flow of an Application

An application does not become a contract by definition, especially since CompX is quite strict in accepting loans. For the rest of this report it is crucial to state the difference between an applicant and a customer: an applicant is a person who applied for a loan but has not been accepted yet; a customer is a person who has received a loan and repays the loan to CompX. The applicant must go through several steps to be assessed and become an accepted customer. Figure 4 shows the process for an applicant to get an application approved.

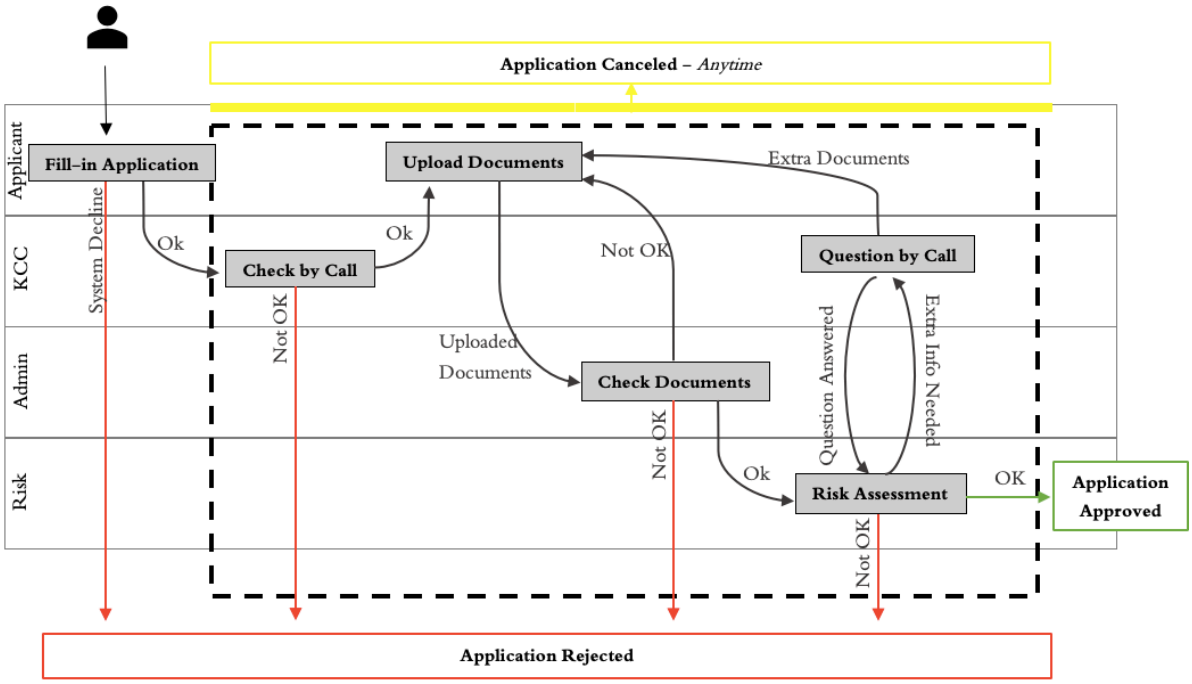


Figure 4. Workflow of the application process.

However, an applicant will not always go through the whole process. There are four check points identified where CompX can reject the application. These are represented by the red arrows downwards. Check 1 is after the fill-in, where the system can reject the application because it appears not sufficient enough in terms of accepted risk. In 2019 between January and September, on average, 33,6% of the applications was rejected directly at this Check 1. This share will not influence the workload of the employees of the approval process. If the application goes through, the applicant is called by the Klant Contact Center (KCC) to discuss and complement the filled-in forms. The objective and promiss to the applicant is to carry out this call within 24 hours, excluding Sundays as working day. This is an important key performance indicator to CompX with high priority. With this call, the possibilities of the applicant are discussed and if possible, an offer is sent. However, at this Check 2, only 49,0% receives an offer, thus 51,0% is rejected. Besides rejection, an applicant can always cancel the application on its own initiative. On average, 24,8% does not return the offer within 30 days. If the applicant does return, Check 3 is done at the administrative department, and Check 4 is done by the employees of the risk department who check

the documents and the application on trustworthiness and responsibility of lending. This step might involve some iterations in case of insufficient information or extra information needed. In this phase, 38,2% of the remaining applicants becomes a customer with a contract. Looking at this funnel of rejection and acceptation, the average acceptance percentage from applicants was 16,2% of filled-in forms, measured from January 2019 until the end of September 2019.

However, there are some issues with this process flow, caused by iterations of the same process step. The first side note should be stated at the 'Check by Call'. CompX states to call the applicant to pass this check, but many times the customer does not answer the phone. This implies the customer will call back CompX, or CompX will do another attempt 1 or 2 working days later, depending on the workload. Another issue arises at the document delivery phase. In 70-75% of the cases an applicant forgets an item, hands in a wrong file or uploaded a wrong format. This takes extra effort for the administrative department and the employees of KCC who has to contact the applicant again. Comparable to this, the iteration of the risk assessment occurs as well. During this assessment, there might rise some new questions for the applicant. Again, the KCC is responsible for this information gathering, either by text or by requesting a new file, which has to be checked by the administrative department again as well. As with the 'Check by Call', the applicant might not be reachable by phone immediately, thus the KCC needs to contact multiple times at this step as well.

In summary, it can be stated the process is sensitive to iteration of steps in which multiple departments are involved and that is highly dependent of behavior of applicants. Because of this, the effort put into an application can differ, but the only accepted applications can generate revenue to CompX and have to cover all costs.

2.3 Profit

There are many ways to express the performance of a company. For CompX, an important measures is their new business volume (NBV). There is a target line for NBV on a monthly base and NBV represents all new outstanding debts compared to the previous month, thus new PL and DK contracts and new payouts from DKs. The level of the portfolio is another target of CompX. It represents all outstanding debts at a certain time, thus indirectly includes new contracts and payouts; and adjusts for all repayments.

The profit on company level depends on the net debts per month on which all interest is paid minus all the costs the company makes. The objective of this research is to generate a definition of profit on contract level with the purpose of making a comparison between two loans. So, generally, on the one hand there is the revenue of a product and on the other hand there are product costs. In the case of a loan, all costs and profit can only be fully known if the contract has finished. Therefore, only finished contracts are taken into account for the model. The revenue of a contract is defined as the sum of all interest a customer has paid. This is a monetary value in euros. However, as interest changed over time, some adjustments are needed for a better comparison. This will be discussed in Section 2.3.1. 'Costs of a company' is a wide

concept, and for most purposes it is not needed to take into account all costs when allocating them. This will be discussed in Section 2.3.2.

2.3.1 Revenue and Correction

As described in Section 2.1, interest rates have changed over time. Thus, for example, comparing a loan of €25.000 started in 2011 with a loan of €25.000 that has started in 2014, having all other characteristics the same, will have totally different interest revenue. Underlying assumption here is that the repay behavior is not influenced by other factors that are related to the time the loan takes place, such as general economic state. Thus, the circumstances of two year repay are the same for a loan between 2011 and 2013 as for 2016 and 2018. It assumed economic state has not been changed dramatically such that it has influenced the repay circumstances of a customer between 2011 and now. No big changes such as the economic crisis of 2009 (see Section 2.5 for more details) occurred in the time scope of this research.

The model should predict the profitability of a loan that arrives now or in the future, based on characteristics of an application. The relevance of this correction will be supported with an example. The objective is to create a model that predicts the profit, which highly depends on the revenue. The prediction is made based on a set of characteristics (further elaboration on these characteristics in Chapter 3). The model is trained on data with these characteristics and the output which is the profit. For example, one loan had the input of (loan volume: €25.000, age: 48 years, 2 applicants, contractual duration: 120 months) in 2011 and paid 7,2% interest rate, but repaid the whole loan in 36 months. Another loan had the input of (loan volume: €25.000, age: 48 years, 2 applicants, contractual duration: 120 months) in 2016 and paid 4,2% interest, but repaid the whole loan in 36 months. For this example, assume the profit of the first loan was €2500, the profit of the second loan was €1500. These applications are exactly the same, and behave exactly the same so the model should recognize them into one category, in order to predict an application in future. However, this is not the case, because the profit is way different due to the interest rate. That is why a correction is needed: the model should see these loans similar in order to predict likewise loans in future.

The correction is made, towards the current levels of interest rates. It is considered too time-consuming to correct each loan exactly and therefore an approximation method is applied to each loan revenue. Each interest period gets a code. There are eight interest periods, Roman numbered according to Table *2*.

Table 2. *Periods of interest rate*

Period	Start Date ; End Date	Period	Start Date ; End Date
I	< 01-05-2012	V	11-11-2015 ; 01-02-2016
II	01-05-2012 ; 01-05-2014	VI	01-02-2016 ; 01-06-2016
III	01-05-2014 ; 01-04-2015	VII	01-06-2016 ; 01-06-2018
IV	01-04-2015 ; 11-11-2015	VIII	01-06-2018 >

The interest rate of period VIII represents the current interest rate. Next, the interest in euros for interest period VIII is compared to the interest of another period on a monthly base starting at month 1 up until month 120. The percentual difference, between the sum of interest paid, is calculated in each month which represents the interest difference between the 'old' and current interest rate. This percentual difference represent the correction and changes over time. The decision is made to merge the periods into three different durations: 1- 36 months, 37-60 months, 61-120 months, to overcome too time-consuming correction. For each duration the correction is averaged. All these correction percentages for each category are listed in Table 3. So for example, if a loan of €7.500 started in period II and took 42 months, the interest paid is corrected with a multiplication of 0,8996 (see second row, seventh column in Table 3).

Table 3. *Correction of interest rates*

Bucket	**Period**	**Interest Rate**	**Max. correction**	**Min. correction**	**1–36 months**	**37–60 months**	**61–120 months**
€5.000–€10.000	*VI, VII, VIII*	**6,50%**	1	1	1	1	1
	I, II, II	7,20%	0,9055	0,8797	0,9021	0,8996	0,8931
	IV, V	6,90%	0,9437	0,9282	0,9416	0,9401	0,9362
€10.000–€15.000	*VII, VIII*	**5,20%**	1	1	1	1	1
	I	6,80%	0,7701	0,7198	0,7631	0,7582	0,7455
	II	6,60%	0,7927	0,7471	0,7864	0,7820	0,7705
	III	6,50%	0,8046	0,7614	0,7986	0,7944	0,7835
	IV, V	6,10%	0,8558	0,8236	0,8514	0,8483	0,8401
	VI	5,70%	0,9143	0,8949	0,9116	0,9098	0,9049
€15.000–€25.000	*VII, VIII*	**4,40%**	1	1	1	1	1
	I, II	5,90%	0,7507	0,7036	0,7441	0,7395	0,7276
	III	5,70%	0,7764	0,7339	0,7704	0,7662	0,7555
	IV	5,30%	0,8335	0,8015	0,8290	0,8259	0,8178
	V	5,10%	0,8654	0,8394	0,8618	0,8592	0,8527
	VI	4,90%	0,8999	0,8805	0,8972	0,8953	0,8904
€25.000–€75.000	*VIII*	**3,90%**	1	1	1	1	1
	I, II	5,80%	0,6781	0,6242	0,6704	0,6651	0,6515
	III	5,30%	0,7404	0,6963	0,7342	0,7299	0,7187
	IV	5,10%	0,7688	0,7293	0,7632	0,7593	0,7494
	V, VI	4,80%	0,8157	0,7840	0,8113	0,8082	0,8002
	VII	4,20%	0,9298	0,9175	0,9281	0,9269	0,9238

The percentage for each duration is an average of the correction percentage for that period. So, the average correction percentage over 36 months, over 60 months and 120 months. One exact example is given in Appendix B. The accuracy of this correction is calculated by comparing the corrected interest to the exact interest of the current period. One example will may clear this: a loan of €65.000 with interest rate 3,9% that took 72 months had a summed interest of €11023,28; a loan of €65.000 with interest rate 5,8% that took 72 months had a summed interest of €16593,70. So, according to Table 3 the second should be corrected by a factor of 0,6515 which brings the summed interest to €10810,80. This close to the €11023,28, but not exact: the accuracy is: $\frac{10810,80}{11023,28} = 98,07\%$. This accuracy measure is executed on a sample of different loan volumes, durations and interest rate. The results of these tests are shown in an table at Appendix B2. The correction accuracy is for almost all cases above 98,00%.

2.3.2 Costs

The objective is to compare loans based on differences in expected profitability. So, only costs that do differ per contract are relevant within the perspective of this purpose. This is comparable to the approach of contribution margin, widely used in manufacturing. According to the theory of contribution margin, only variable costs are considered to be relevant. The definition of contribution margin by Wouters, Selto, Hilton, & Maher (p. 840, 2012): "the difference between sales revenue and variable cost of sales that measures the incremental profit earned toward covering fixed costs and desired profits", confirms this inclusion of variable costs only.

There are two types of costs that do differ per contract: risk costs and commission costs. All other costs are not assignable on contract level in a way that these will differ per contract. A description of these other costs will be discussed in Section 2.4 to argue why these are not taken into account for this model; a description of the costs included will follow below.

Firstly, risk costs are the costs associated with the provisions of a bank and are interpreted as an opportunity cost. This is a cost that is specified on contract level. A provision is an amount of money set aside, to create less risk in case the loan is not repaid. In literature, this is also referred to as the expected loss (EL) of a loan. In formula this looks like: $EL = PD * EAD * LGD$ (Kim & Kim, 2007). Where PD represents the probability of default, EAD represents the exposure at default, LGD represents the loss given default. PD and LGD are rates, EAD is a monetary value. This amount of money, EL, is set aside which is called (loss) provision (Hlawatsch & Ostrowski, 2010). This is an adjustment to the book value and it cannot be used for other investing purposes. The LGD is a standard rate at CompX and has not been changed since the existence of CompX; the rate is 0,4. The EAD and PD are adjusted each month. EAD is adjusted based on the net debt at that time. For example, a loan of €10.000 in the first month will have a EAD of €10.000, after the first payment of €250 (€75 of interest and €175 repayment), EAD will be €9.825. The PD is calculated with more complex models taking into account the state of the current economy and the repay behavior of the customer. A loan started in September 2019, had a PD of 0,01. A delay in repayment,

immediately leads to a higher PD. The higher the default probability of a loan, the higher the level of provision. In this example, the EL would be €40 in the first month. Practically, the provision, i.e. EL, does not imply a loss. If the customers repays the loan without defaulting, the money of provision decreases over time and in the end can be fully lend again to a new customer. However, the money is reserved for a part of time. Therefore, the costs for provision are interpreted as an opportunity cost, using the profit that should have been generated by the money at provision. An opportunity cost is a cost for 'not doing something'. CompY has set a target profit to CompX that states that each euro lend should generate 17 cents of revenue. Thus, each euro of provision implies a loss of 17% on annual base and 1,317% on monthly base. The $average\ provision\ per\ month\ *\ 1{,}317\% * actual\ duration$ is the cost of expected loss, in other words risk cost. CompX is not responsible for economic capital because this is regulated by CompY, and is therefore out of scope. This formula:

$$ad * 1{,}317\% * \frac{\sum_{m=1}^{ad} provision_m}{ad}$$

where:
$ad = the\ actual\ duration\ in\ months$
$m = the\ month\ of\ the\ payment\ where\ m = 1\ at\ the\ month\ of\ the\ first\ repayment$

can be simplified to: $1{,}317\% * \sum_{m=1}^{ad} provision_m$

Secondly, commission costs are not applicable to all contracts, but are related to the way a customer entered CompX. There are three comparison websites, Independer, Pricewise and Geld.nl that generate leads for applications. If the customer applied for a loan via a fill-in form of one of these parties, they receive a share of the revenue if the application becomes a contract. This is a percentage paid over the net debt, calculated each month. This is referred to as the commission or the commission costs For the profit model, these costs are only allocated to contracts that came in via such a website. The costs are represented by the sum of the amount of money that is paid monthly to a third party. These costs are available per contract in the data base.

In summary to the start of this Section 2.3, the 'profit' of an application is calculated by summing the corrected interest paid, minus the risk costs and the commission costs:

$$profit = \sum_{m=1}^{ad} interest\ paid_m - 1{,}317\% * \sum_{m=1}^{ad} provision_m - \sum_{m=1}^{ad} commission_m$$

where:
$ad = the\ actual\ duration\ in\ months$
$m = the\ month\ of\ the\ payment\ where\ m = 1\ at\ the\ month\ of\ the\ first\ repayment$

2.4 Other Costs[4]

Practically, CompX is a subsidiary of CompY, which implies that the funding of money to be lend is not regulated by CompX. It is a company financing their own operations based on budgets set in cooperation with CompY. This influences the budget strategy and decision-making within this 'given' financial capacity. The total operational costs and overhead costs, i.e. 'the costs to keep the business running', are budgeted on an annual spend that is quite constant over time. This was true for the year of 2017, 2018 and 2019. The years 2017, 2018 and 2019 are taken into account only, because the portfolio increased less explosively since 2017 (Figure 1) and suggest some stability. In addition, the cost structure changed since 2017, due to a relocation in company structure from CompZ towards CompY. All relevant costs are the costs for the labor at the application process; costs for register checks at the application process; marketing costs to attract web-users to CompX and all remaining costs related to back-office processes referred to as overhead costs.

2.4.1 Application Process Labor Costs

The application process labor are the costs that are associated with the process flow as described in Section 2.2 and the personnel hired to execute this process. These costs are caused by handling all applications, from which one part remains and finally becomes a contract, i.e. a customer paying revenue. This percentage is referred to as the approval conversion rate. The outcome of an application, passed the system decline, is not known beforehand. However, at arrival there is no reason to assume the application will not succeed and thus no reason to reject the application. Thus, it is assumed there is a need to handle these applications in order to achieve the approval conversion rate. In other words, to find one acceptable contract, more than just one application is needed. All costs associated with this process are allocated to the contracts that are approved. Assuming these costs will not decrease or increase next year, it can be assumed the cost of labor at the application process is approximately €230 per contract, based on the trend over the past three years.

2.4.2 Register Check Costs

Register checks costs are the costs associated with doing three tests for an application. Information about an applicant, or in case of two applicants, is requested from registers outside of CompX. These are registers on national level and concern morality checks on applicants. One system is called BKR, in Dutch *'Bureau Krediet Registratie'*: an institution that keeps track on consumer credits, i.e. other debts, and takes notes of defaulting credits. Two other checks refer to whether the applicant provides a valid identification document or has committed fraud at other financial institutions: VIS and EVA[5]. These costs are associated

[4] The establishment of these costs is not supported by showing real values, but are based on real budgets and portfolio sizes. These details are considered as non-public, confidential information.

[5] VIS: Verificatie Informatie Systeem and EVA: Externe Verwijzings Applicatie

with new contracts. Assuming these costs will not decrease or increase next year, it can be assumed the cost of these checks will be approximately €26.

2.4.3 Marketing Costs

The marketing costs are all costs associated with creating leads. A lead is a path that leads to interaction with the website of CompX. This can either be just a website visit or an application. The major part of these costs are originating from Google Advertisements (GoogleAds/Ads). The price of these Ads is fluctuating based on market forces, however details about this bidding strategy is out of scope for this research. A 'click' refers to an internet user who clicks on the link of CompX, Ads, on the internet, and this immediately implies CompX has to pay for that lead. The marketing conversion is the rate of clicks that lead to an application, i.e. a web-user that fills in the application form. Besides, these direct costs of a click, there are other smaller marketing campaigns and marketing consultancy costs. The direct effects of marketing are hard to measure, since the exposure of CompX via marketing, is not traceable. If a web-user decides to apply for a loan aside from that click, these activities cannot be linked to that contract. Therefore, it cannot be stated, 'if a click does not generate an application directly, the click was useless'. For example, if that web-user visit CompX via an ad, reads information, leaves the website and calls CompX the next day for an application, that lead is not traceable to the application anymore. However, all marketing costs are associated with the number of new activated contracts. Looking at the last three years, the costs increased per contract year by year with about €15 per contract.

What these costs will be next year is not specifically known, neither the average cost per contract. This depends on marketing strategy, state of market saturation and customer approach, which is not in the scope of this research. However, the increasing costs can be explained by the law of diminishing returns (Ahmed Ali, 2016; Armstrong, Adam, Denize, & Kotler, 2015). This theory yields reaching the 'first' customer on the market does need a relative low level of effort and/or money. The more customers reached, the more effort and/or money is needed to reach a new customer. Thus, reaching the 'last' customer is more expensive which may explain the increasing marketing costs. Assuming these costs will not decrease or increase next year compared to 2019, it can be assumed the cost of these checks will be per contract is approximately €180.

2.4.4 Overhead Costs

The overhead costs are associated with all activities and personnel back office, such as the MT, Human Resources, portfolio analysts and other supportive services. The major part of these costs are fixed and not depending on the number of new contracts or the contract base: if CompX would stop providing loans, it still needs these services. The same accounts for a large part of the overhead costs. The last three years, these costs decreased slightly year by year.

Assigning these costs to a contract could be done by multiplying the average cost per month by the actual duration of a contract. The average costs in that way is €6,63 per month per contract, measured over the

past three years. Though, within the objective of this research, these costs are not allocated to one contract, the total costs for overhead that can be allocated to a contract, are $€6{,}63 * actual\ duration$. Within the overhead costs, IT and building costs are not included. The building is rented and the office is located in the building of CompZ. IT personnel is mostly hired from third parties and is partially shared with CompZ. Since overhead costs are not necessary for cost allocation in this model, these costs are not further elaborated on.

These costs together make the total cost, besides the costs from the model, of a contract to:

$$costs = €230 + €26 + €180 + €6{,}63 * duration = €436 + €6{,}63 * duration$$

From this, a conclusion could be that a contract should at least yield more than €436, in order to contribute to the profit. However, exclusion of a group can influence other costs which may affect another product choice etcetera. This is not the objective of this research and will therefore not be discussed further.

2.5 Literature Review: Scientific Input

Apart from this thesis, a literature review has been conducted. A summary of this literature review and other insights gained from scientific research will be given in this Section. It provides scientific background of the research area; insights for methodologies of analysis; and already investigated fields.

From the review appeared, predictive models in consumer finance or loans in general, tend to focus on predicting risk. Mostly in terms of probability of default. Risk is defined by Dukić, Dukić, & Kvesić (2011, p.391) as "the probability that a debtor will be unable to pay interest or the principal according to contract terms". Risk of loans can be assessed from two different points of view. On the one hand, risk can be seen from the perspective of consumers. This partly origins in the concept of financial capability (Xaio, 2016) which relies on knowledge and understanding;, skills; and confidence and attitudes which influences the consumers' behavior. The governmental policy is to protect consumers from irresponsible debts, which regulation is executed by AFM. This institution controls banks on this responsibility (AFM, 2018). On the other hand, there is risk for a bank. In terms of loans and credits, this risk is related to the probability a customer will not repay the loan, neither the interest, which is a loss to the bank. Accurate classification of potential risk in financing is in benefit to the lender, by creating more profit or reducing loss; and to the consumer by avoiding overcommitment and unnecessary rejection (Hand & Henley, 1997).

The applicant can be assessed manually by an employee of the bank to decide whether the risk is acceptable or not; thus the loan will be provided or not. Besides this manual assessment, there are also other methods to determine the risk of borrowing money to customer. In some countries a score that determines the level of risk of a person, is generated by national institutions. This implies a customer will always have a score that is measured likewise for everybody and is accessible by the bank. A well-known example is de FICO-score used in the United States, implemented since 1989 (Leong, 2016; Sengupta & Bhardwaj,

2015). Besides, there are many methods to create model that predicts the risk level of a person. However, there is an underlying assumption in all these researches about decision-making in lending: the customer is not allowed to repay the loan earlier without any fee. This makes the situation of CompX unique to the existing scientific field. Still, the approaches to create a predictive model can be applicable to this situation.

The applicability of a method depends on the type of variables. Some cannot handle categorical values or numeric values. One method found, is the discriminant analysis. This method assumes normal distributed input variables (Mester, 1997), which is not the case for loan applications and therefore eliminated for this research. Another method suggested by Mester (1997) is logistic regression, which assumes the probability of risk is logistically distributed. The output variable has to be categorical. Linear regression assumes a linear relation between the probability of default and the input variables. The output variable has to be continuous. Both techniques are more 'classic' statistical methods and follow the same input structure: the input variables are assigned a weight which lead to a linear outcome with a certain variance or, in the logistic case, a probability per class (Tu, 1996). Logistic regression with more than two output categories it is often referred to as multinomial regression (Starkweather & Moske, 2011). The use of Bayesian networks is a relatively new method. As Chernyak & Pavlenko (2010, p.327) describe, it "provides a framework for representing, quantifying and managing the uncertain knowledge in concentration of credits risk exposures."

Another widely-used method for predictive models is the decision tree (DT). A decision tree can handle classification problems and regression problems. One popular method is also referred to as 'CART' according to the book of Breiman (1984) who set a standard in decision trees. A decision tree consists of nodes and edges where questions are hierarchically ordered and a set of sequential questions, i.e. decisions, leads to a region of decision space (Criminisi, Shotton, & Konukoglu, 2011). Decision trees have been applied extensively in discriminant and predictive modeling and the method is able to find exact patterns and discover features. Besides, an advantage of decision trees is that it is easy to interpret and it works intuitively (Myles, Feudale, Liu, Woody, & Brown, 2004). The method requires very little data preparation, can easily deal with categorical variables and numerical values, and can have multi-categorical outputs (Gupta, Uttarakhand, Rawat, Arora, & Dhami, 2017).

Very closely related to this decision trees method are ensembles. An ensemble is a set of classifiers, usually decision trees, that are combined using weighted or unweighted voting to create a predictive model (Dietterich, 2000). One common type of ensembles is called random decision forests, which is an ensemble of randomly trained decision trees, also founded by Breiman (Gupta et al., 2017). Each tree is randomly different from another one (Myles et al., 2004). The accuracy and generalization of ensembles is argued to be better than the accuracy and generalization of decision trees. In addition, according to Gupta et al. (2017), an ensemble is better in detecting outliers and estimates the importance of variables.

However, interpretation by humans is less easy and has a higher potential in overfitting for the case of noisy data.

Even less interpretable are Neural Network (NN) or deep nets. A NN tries to replicate the functioning of the human brain and generates thousands of edges and nodes in which it tries to seek for patterns. Based on the iteration, it adjusts weights of these nodes and the influence of a variable (Gurney, 1997). A deep net, or Deep Neural Network (DNN), is a type of NN that requires more than one hidden layer between the input and output variables. A NN is easily applicable to any type of problem, but works best with normalized variables (sometimes automatically done by the NN software) (Tu, 1996). In addition, according to Tu (1996), NNs are good in finding less straight forward, non-linear relations; and are able to detect interactions between input variables. However, one issue with NN and DNN is that it is hard to depict what exactly happens in the algorithm, also referred to as the black-box problem. A NN does not store what is has learned, but diffuses the knowledge in way that it is hard to decipher (Castelvecchi, 2016). In addition, the performance of such a model is not always better than other, more straightforward, methods. For example, West (2000) compared the performance of a NN model to a logistic regression model, where logistic regression was found to be a better method. Also, Tu (1996), states logistics regression is better in finding causal relationships and NNs are prone to overfitting.

Though existing research focuses mostly on predicting risk in terms of default, this is not the objective for this research. In comparison, it is more related to the risk in terms repaying 'too good', because if the customer repays too good the loan might become unprofitable to the lender. This short overview confirms the newness of the model to be created and underlines the added value to existing research in this scientific area.

In summary, there are multiple methods to construct a model. The advantages and disadvantages of each as described above, are used to make a decision about which method to apply and which not. This scientific overview of possibilities are input to Section 4.3 and 4.4.

2.5.1 Performance Measures

There are multiple ways to interpret the performance of a model. Different type of measures are applicable to linear problems and classification problems. Usually, this performance is measured by a split in the data set before training the model: the model is trained with 80% of the instances and tested with 20% of the instances (Sanchez-Barrios, Andreeva, & Ansell, 2016). Starting with linear problems, R-squared and Mean Absolute Error (MAE) are common measures. The model predicts a value and with the real values of the testing dataset the prediction is evaluated. R-squared, the ratio of sum of squares, is usually judging the adequacy of a regression model (Montgomery & Runger, 2011). MAE measures the average difference between each prediction and real value.

For the performance measures of a classifier model, an outline of Parker (2011) provides an overview of different measures and their quality. Evaluation is based on the confusion matrix which distinguishes true positives (TP); the positive instances that are predicted false positives (FP), the negative instances that are predicted positive; true negatives (TN), the negative instances that are predicted negative; and false negatives (FN), the positive instances that are predicted negative.

First, Parker (2011) starts with mentioning the *accuracy* measure which simply counts the number of good predictions and divides by the total of the predictions. However, this method is riddled with problems and therefore not considered accurate at all by the author. Next, he distinguishes two types of measures: point measures where the outcome of the performance depends on the threshold of the model and integrated measures that quality over multiple thresholds. *Precision* is measuring the TP over all actual positive instances. *Recall* measures the TP over the TP and the FP. *F1-measure* represents the harmonic mean between precision and recall. The *phi-coefficient*, or Matthews Correlation Coefficient, is another measure and considers the full confusion matrix. Both, phi-coefficient and F1-measure, are point measures and are designed to apply to data where distribution is skewed. *R-precision*, or precision at k, measures the precision where threshold is exactly set to an objective amount of k labeled positively instances. The Area Under the ROC Curve (*AUC*) is not a point measure and plots a curve, the ROC curve, for the cumulative distribution functions of both instances one each axis and AUC represents the area under this curve. A random classifier would have AUC of 0,5 and a perfect classifier has an AUC of 1. The *average precision* averages the precision at all possible levels of recall. It is comparable to the AUC measure, but does take into account the class distribution. Cohen's k measures the Area under the Cohen's K (*AUK*) where the curve represents the cumulative distribution of positives by a function that compares the model prediction. AUK and AUC integrate over levels of specificity, but AUK does consider the class distribution of the test set. *H-measure* is another measure opt to overcome problems with earlier mentioned measures. It does not integrate over specificity but over possible costs of misclassification and thus implies a distribution of costs (symmetric beta distribution assumed). The author concluded there is a general issue with the fact each method indirectly or directly implies a determination of loss or costs. Therefore, H-measure is opted to be a good measure for comparing the phi-coefficient as an optimized value. AUC assumes all costs and all thresholds would be equally likely, so is better to use in such cases.

In summary, there is not one single answer to what the best performance measure is. For the evaluation of the models constructed in this research, not all will and can be used. Based on the outline here, a decision can me on which measure is best applicable in this situation. This will be further elaborated on at Chapter 4.

3. Data Preparation

3.1 Data warehouse of CompX

The data warehouse of CompX is owned by themselves and accessible via Microsoft Server Management SQL. The data is hold in tables that are directly linked to the fill-in forms, textual and numerical boxes and drop-down menus. These tables contain large data sets, which yield all information from the software tools CompX is using. These tables will also be referred to as the 'raw' tables. To serve customers and applicants, two software applications are used. Lara and CKS. Lara is more focused on the application process. All departments, KCC, the administrative department and the risk assessors work with Lara to serve the application process. It contains queues and workload for each department and all details about the application and applicant contact. CKS only serves customers, thus those who have a loan. This system keeps track of all payments and information related to that. Lara does contain some information about an activated loan but this is very scarce.

In addition to these raw tables, CompX implemented a star model to aggregate this data to a smaller set with most important data and modified some data for easier interpretation with logic rules. The star model describes a pattern for a data warehouse with facts and dimensions where facts describe records of the data warehouse and dimensions contain descriptions about these facts (Grefen, 2016). These dimensions and facts are used to analyze for businesses purposes and to overcome the cumbersomeness of the large original tables.

As input for the model, decisions have to be made about which tables and variables to include and which not. Including all information about an application makes the dataset too large to handle and this is not recommended to analyze (Mester, 1997). For this research both table types were available: facts and dimensions; and the large original tables. Also, for analyzing purposes, applications with too many missing values are excluded.

The objective requires the availability of application information and payment information of the contract. The application information is referred to as the input of the model and can be a numerical variable or a categorical variable. The payment information is referred to as the output of the model and can also be either numerical or categorical. All considerations about the collection of variables are described next.

3.2 Input Variables

The input variables are characteristics of a loan application that are available at the time the application arrives. These are biased to what is available in the data warehouse systems and are based on a brainstorm session with a part of MT to what might influence the profitability of loan. Notes of that brainstorm are available in Appendix C1. Within these restrictions, 28 input variables are listed in columns per contract. Some of these take numerical values, discussed in Section 3.2.1 and some of these are categorical, discussed

in Section 3.2.2. In addition, some variables were used to gather or check data but cannot be used for the model. These are shortly mentioned in Table 4.

Table 4. *Variables for data gathering without influence on the model*

Variable Name	**Definition**
Contract Number	A unique number for each contract. For each new loan, a new contract number is generated.
Product Code	A code that marks the product type, whether it is a PL or DK.
Client number	A unique number for each applicant/customer. The number remains and will never change for that person.
Start Date / End Date	The date a contract started of officially ended.

3.2.1 Numerical Input Variables

Loan Volume Demanded – this is the loan volume an applicant originally applied for, in euros. This value might be missing if the application came in via phone. However, during the process, the loan volume might increase, because of new insights into the applicants' financials or take-overs from competitors; or decrease, because of new insights by risk assessors that decrease the maximum loan capacity (MLC).

Loan Volume Contract – this is the loan volume that is actually paid out, in euros. It is also referred to as the original loan volume and the first net debt of customer.

Contractual Duration (**DurationC**) **–** this is the duration of the loan according to the contract in months.

Age – for a loan of one applicant, this is the age of that applicant at the time the loan was approved. In case of two applicants, the highest age is assigned to the contract. This, because the oldest applicant can influence the product conditions. For example, the contractual end date may not be later than the 75th birthday of the oldest customer.

Contractual Interest (**InterestC**) **–** is the sum of all interest paid, if the loan is repaid according schedule. So, without extra repayments or defaults. It represents the revenue according contractual conditions and depends on the loan volume and the duration. It is recalculated to one general interest rate as described in Section 2.3.

Monthly Payment – refers to the monthly payment as agreed to in the contract. It depends on the loan volume and the contractual duration. This variable does not exist in data but is calculated using the duration, interest rate and loan volume.

Open at BKR (**OpenBKR**) – refers to the amount of other debts the applicant(s) has or have, and are registered at the BKR system. This is a sum in monetary value of euros. If the value of this variable is higher than 0, this implies the applicant is accounted for other debts. BKR denotes only original loan volumes and does not adjust for repayments in time. In case of two applicants, it takes into account the sum of all unique loans. In addition, BKR denotes bad payments but this is not included in this variable. However, if this occurs the applicant will never receive a loan at CompX thus this actually is not possible for a contract.

MLC adjusted for BKR (**MLC–BKR**) – refers to the maximum loan capacity (MLC). This is the maximum loan volume the applicant may get at CompX, minus the other debts from BKR (OpenBKR). The MLC is calculated based on income, expenses, marital status and family situation. It takes into account expenses for living such as insurances and groceries. This calculation is regulated nationally by VFN, based on Nibud (*Nederlands Instituut voor Budgetvoorlichting)*, who sets norms for these types of expenses and the responsible amount to be lend.

prcMLC – refers to the percentage lend of the MLCBKR. It is the loan volume divided by the MLCBKR.

Income – refers to the net income per month taken into account for the application and the calculation of MLC. In the case of two applicants, or more income types, the income is summed to one value.

ExpenseSum – refers to the expenses related to the mortgage or rent; and in the case of a divorce to alimentation. Mortgage is denoted before taxes. These are the only types of costs denoted in the application form.

Creditscore - refers to the creditscore of an application. The creditscore aims to give a prediction about creditworthiness and is related to the probability of default. The score is based on about 10 factors, such as income, age and BKR experience. There is a *scenario* creditscore which is given to an application by arrival and an *offer* creditscore which is assigned to the applicant at the moment the offer is sent.. These can differ because the first check, with the call agent, can lead to changes in values which impact the creditscore. For this analysis, the scenario creditscore is used if available, otherwise the offer creditscore will be used. Operationally, the score is not used as a replacement of manual judgement, but as a cut-off line to start this judgement or not.

Ratio EX–IN – refers to the ratio between the ExpenseSum and the Income of an application. It is simply calculated by dividing the ExpenseSum by the Income.

IN–EX – refers to the difference between the Income and the ExpenseSum. This is the disposable income without taking into account other costs than housing.

prcPay – refers to a percentage the customer is paying relatively to what it has to spend. How much the cutomer is paying for the loan compared to the disposable income,. It is calculated by the monthly payment divided by the IN-EX.

3.2.2 Categorical Input Variables

The categorical input variables can have two values, i.e. binary, or more. All categorical variables are listed below.

LoanGoalCategory - refers to the overarching category of the purpose of a loan. The different application goals are clustered into more broaden categories. There are about 10 categories in a drop down menu. Only one can be selected. If a loan is used for multiple categories, the category with the biggest share of the loan volume, will be set as the loan goal. The choices of categories (a dropdown menu) are adjusted and added over time which implies applications before an addition cannot have that category, since it did not exist before. It can happen that the application goal is not specified, but these contracts are not skipped because this is an administrative error. Other variables of this loan might still be valuable.

HousingType - refers to the living situation, such as a rental house or mortgage living, of the applicant(s). If there are two applicants, they have to live at the same address by product condition. The same housing type is than assumed for the second applicant. Therefore, the housing type of the first applicant[6] is representative.

City - refers to the city or village the applicant(s) lives in, which has to be in the Netherlands by product condition.

ZIP – refers to the first two digits of the ZIP-code of the customers. Originally a ZIP-code consist of four numbers and two letter. There are only two first digits available in the data warehouse because of privacy issues where showing more than two is too specific. Thus the ZIP contains geographical information about where the customer lives but is more high-over and less specified than the City. There is a map in Appendix C2 which shows to which area each ZIP code belongs in the Netherlands is shown in.

ProspectType – refers to the relation between CompX and the application. Four types are distinguished. A *Prospect* is new and has never had an application or contract before at CompX. An *Ex-Prospect* has never been a customer before but has applied for a loan in the past which was cancelled or rejected. A *Customer* already has a contract at CompX which is not ended yet. Lastly, an *Ex-Customer* has had a contract at CompX before but this loan has already repaid fully at the time of the new application. This

[6] Technically, each application has 1 applicant which automatically is the first applicant (*'Hoofdcontractant'*). In case of two applicants, there is a second applicant as well, the (*'Medecontractant'*).

information is extracted from the profile of the first applicant only. In case a the first applicant becomes a second applicant with a new application or the other way around, this will be assumed to be neglected.

Gender - refers to the gender of the first applicant. If there are two applicants from the same gender, this is not visible in the data extraction. It is assumed that all contracts with two applicants consist of a male applicant and a female applicant.

IND_Children (Indicator of Children) – refers to whether applicant(s) do(es) have children under the age of 21. No matter if these children still live in the same household. By work instruction, this is binary variable: 0 for no children, 1 for one or more children. However, the field in the form has a wider range than 0 and 1 which led to some variation in the data. This is solved, by adjusting all values above 1 to 1.

Source - refers to where the application is coming from; how did the customer find CompX and fill in the form? This is a constructed variable from Account and Application Source. The Account contains information only about digital channels and refers to which strategy or channel has been used. The Application Source is more high-over and refers to web, telephone or revision. So, web application can be specified with the Account. Revision are special applications on initiative of CompX to assure the fit of an existing loan. This is not considered as relevant to this research so these kind of applications are eliminated from the selection. The new categories made are listed in Table 5.

Table 5. *The categories of the variable Sources*

Source Category	**Explanation**
Telephone	This application came in by a call
WebDirectFree	This application came in via internet, without CompX directly paying for visibility
WebAds	This application came in via internet. The customer clicked on advertisements of a search engine (Google or Bing)
WebComparisonSite	This application came in via internet. The customer used a website (Independer, Priceswise or Geld.nl) that compares loan providers and filled-in the application here. CompX pays provision on a monthly base when such an application is activated into a contract.
WebPaidOther	CompX tried some other ways to advertise online. The initiatives had a relatively small share in the portfolio. All these try-outs are in this category.

The Source represents only the last medium the applicant used before filling in the form at CompX.

MaritalStatus - refers to the relational situation of the applicant(s). In case of *geregistreerd partnerschap* or marriage, the loan has to be taken by both. If a person is single or cohabiting, the number of applicants can be 1. This variable also influences the MLC. The variable is extracted from the first applicant. In the case of two applicants it is assumed, the second applicant has the same marital status.

ApplicantsNo (Number of Applicants) – refers to the number of applicants: 1 or 2.

IncomeType – is the type of income of the application. In case of one applicant it is based on that applicant. In the case of two applicants, both types are taken into account. Appendix C3 contains a table with all combinations of income types and how it is converted into a new type used for the analysis.

LVBucket – is an aggregation of the loan volume into 5 categories.

DBucket – is an aggregation of the loan duration into 7 categories.

3.3 Output Variable

The output variables relate to the repay behavior of the customer(s). There are many elements that contain information about this repayment. Default prediction is not the objective of this research and thus measures of default are excluded as a solely output variable. The wanted output is already described as the profitability in Chapter 2, where default is included as a risk cost. This approach will be used for the model.

The output is a continuous value originally. Predicting this profit as a continuous value will imply a regression model. Modeling with this output, will be referred to as Y1 or y_1. Next, the output can be predicted in ranges. These are referred to as classes, which will imply a classification problem. The advantages might be, a class can easily be interpreted as a priority label, which makes it easier to implement the model in a process. Two different classifications are constructed where one distinguishes three classes (Y2 or y_2) and one distinguishes five classes(Y3 or y_3). The ranges are shown in Table 6.

Table 6. *Profit buckets, ranges and average profit per bucket*

Category	y_2/Y2	**Average profit (per bucket) | number of contracts**	y_3/Y3	**Average profit (per bucket) | number of contracts**
A	> €900	€1763 | 11319	> €2000	€2982 | 3013
B	[€400 , €900)	€627 | 10038	[€1000 , €2000)	€1393 | 6950
C	< €400	€209 | 9743	[€600 , €1000)	€783 | 6698
D	-		[€400 , €600)	€497 | 4696
E	-		< € 400	€209 | 9743

One distinguishes three categories A, B, C where A represents the category of highest expected profitability and C the lowest (y_2/Y2); and one distinguishes five categories A, B, C, D, E where E represents the lowest class (y_3/Y3). The number of contracts in this table are extracted from the training data set.

There is an infinitive amount of options for the ranges of buckets. Knowing the overhead costs from Section 2.4.3, a profit under €400 will most likely not add value to the revenue of CompX. Therefore, this bucket is created. Other ranges are somewhat intuitively and taking into account one bucket will not be totally underrepresented.

4. Results

The results can be divided into two parts. The first part of the analysis aims to get general insight into divisions in the data, provide descriptive statistics about variables and depict remarkable trends. Firstly, without taking into account the relation to the profit measure, secondly analyzing the relation to profit for each variable. It is also meant to validate the use of each variable. This first part is covered by Section 4.1 and 4.2. The second part of this Chapter covers the search to a model by using various methods for modelling, ending with the best model found that is advised to implement in the process flow. This application of the model will be discussed after, in Chapter 5.

The goal is to find a model to predict the profitability of a contract in a way to compare contracts on expected profit contribution. The definition as established in Section 2.3 will be applied to do so.

4.1 Descriptive Statistics

According to the data preparation as described above, a dataset of 60668 rows, thus 60668 unique personal loan contracts. 51,26% (n = 31102) of these contracts is ended, thus the final profit is known. Taking into account non-ended loans as well, becomes difficult because the revenue, risk costs and commission costs are unknown as they depend on the actual duration of the contract. Predicting this, is actually the objective of the research. Therefore, making a measure of future profit of these in order to take them into account, will make the model very insecure: the model would make a prediction based on a prediction. A solution could be only looking at loans that really should have been ended by now. This leads to a dataset of n=8488. However, CompX is a relatively young company, which makes the number of loans with long duration scarce, since rapid growth exists since 2014.

Thus for analysis from now on, ended loans will be included only. Of this aggregation, 43,07% was ended because the customer got a new loan at CompX. This is an ITO. It can be discussed whether to include these loans into the analysis or not, because the duration of that customer seems short, however the customer remains a customer at CompX. It is decided to include these and make no distinction because these types of loans are contributing to the workload of the application process; take a sufficient share in this workload; and do follow the same cost pattern as any other loan. First, the variables will be discussed solely in the perspective of representation, next the repayment behavior will be analyzed in terms of how a customer repays.

4.1.1 Variables and their Representation

Intuitively, the loan volume would have high influence on the revenue of a loan: a customer with a higher loan volume will pay interest over a higher net debt. However, there are two variables concerning the loan volume, namely the one the applicant applied for and the one the applicant finally receives. The LoanVolumeDemanded is missing for n=2401. The difference between the LoanVolumeDemanded and LoanVolumeContract is small, as Figure 5 represents.

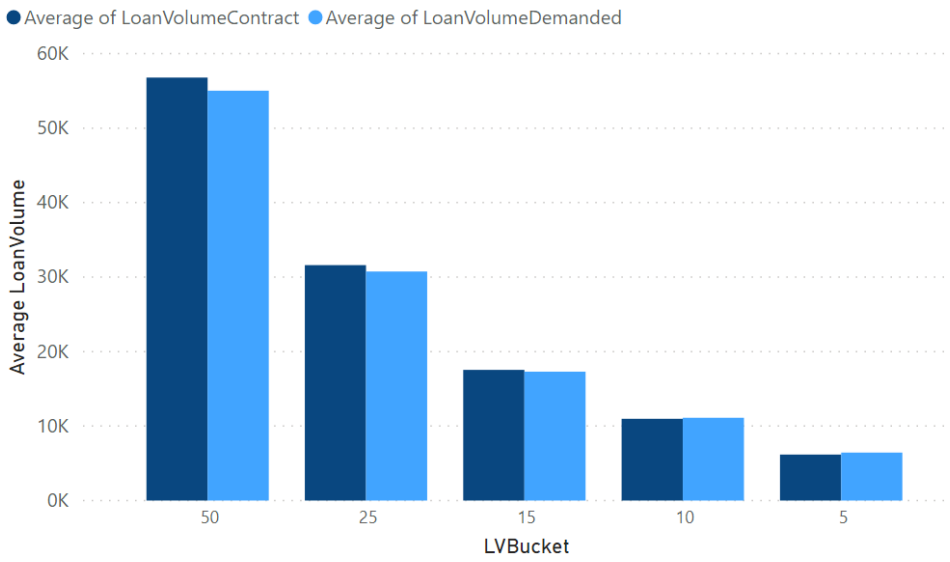


Figure 5. Average loan volume demanded vs. average loan volume of a contract

Since the difference between these two is small, only LoanVolumeContract will be used as an input value of the model, because this information is available for all applicants. The figure below, of LoanVolumeContract, shows some peaks at certain values. Outstanding loan volumes that are provided relatively a lot are €5.000, which makes sense since this is the minimum; €10.000, €15.000 and €25.000 which make sense since these loan volumes represent the start of a new bucket with new interest rates. Rounded loan volumes are also popular, such as €20.000 and €30.000 and €75.000 remarks the maximum amount. This is visualized in Figure 6. A loan amount between €15.000 and €25.000 is most popular. The average loan volume is €17.490.

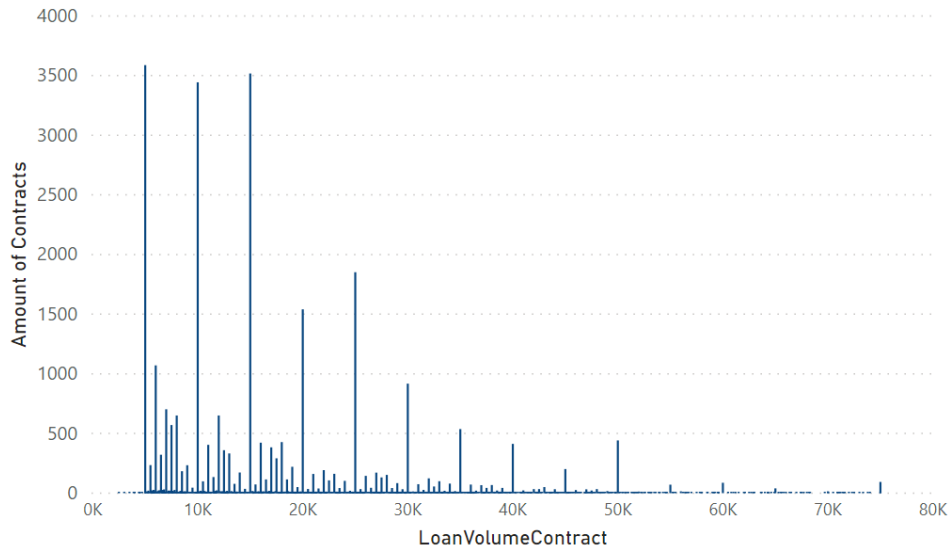


Figure 6. Number of contracts per loan volume

For the duration of contracts, there is a preference for a duration of 60 or 120 months apparently, as Figure 7 shows. Next, Figure 8, suggests the higher this duration according contract, the higher the average loan volume.

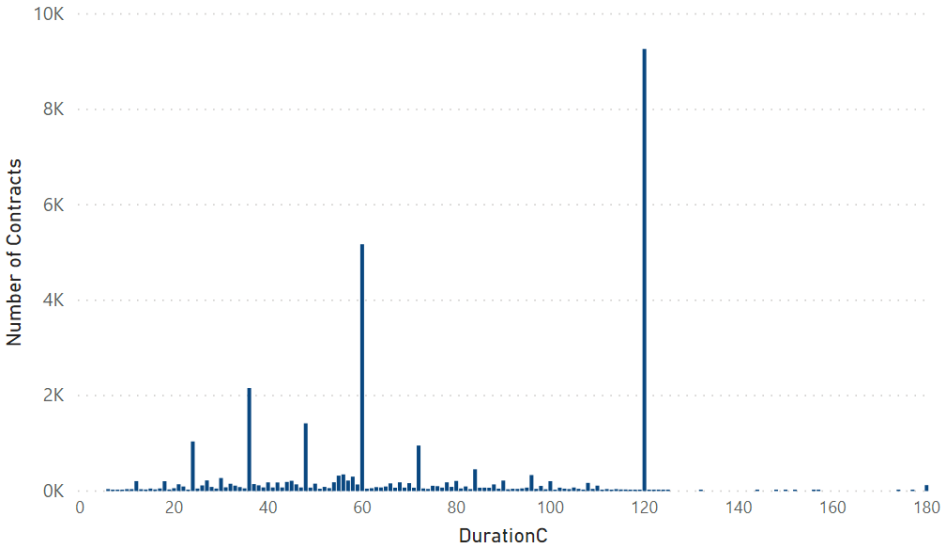


Figure 7. Number of contracts per DurationC

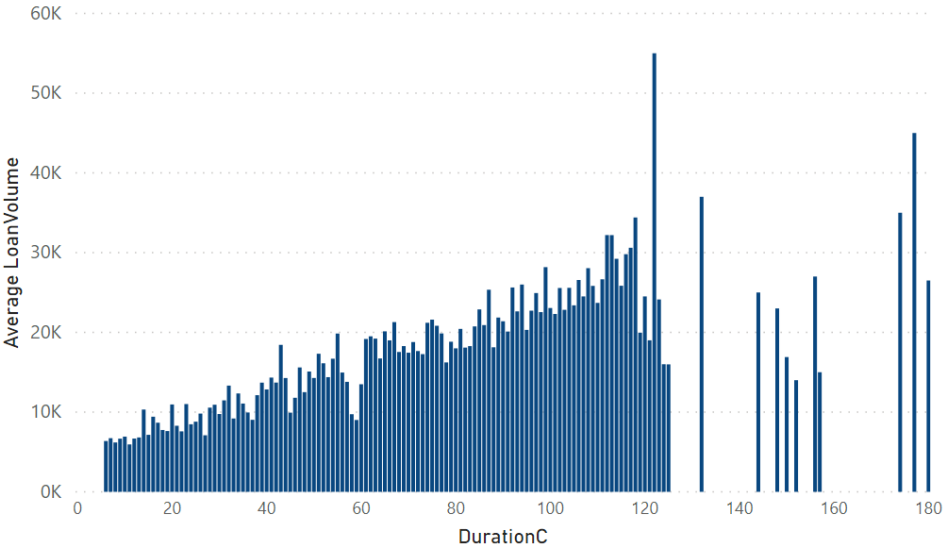


Figure 8. Average loan volume per DurationC

Only around 60 months, and 120 months, there seems to be a drop in this trend. Maybe this is caused by the fact 5 or 10 years are just popular durations to be chosen and are not that much related to the loan volume. Loans with a duration above 120 months are not provided anymore but were in the past for specific purposes only. This specific financing purpose is in case of a mortgage loss selling a house (*Dutch: 'restchuld'*). Therefore, these are very rare, thus the peaks shown are only based on a few contracts and this findings is not considered relevant.

Looking at the actual duration of contract, the trend is totally different. By far, the majority does not fulfill a duration longer than 36 months as represented in Figure 9.

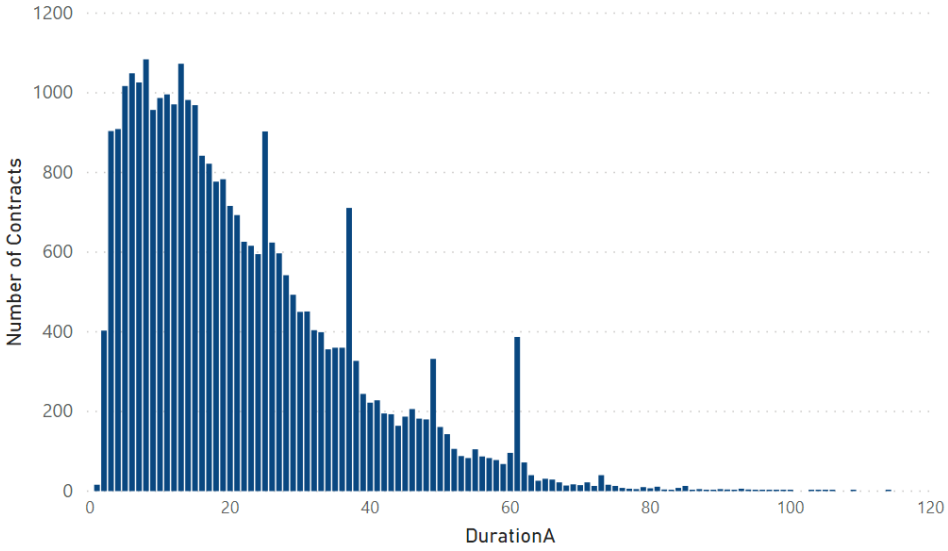


Figure 9. The number of contracts per actual duration

One could argue, this is a biased view on the duration of contracts. The contracts that have not ended yet, may show a totally different patterns but are excluded. Therefore, a fictive duration, DurationF, has been created to gain insight into what the duration of running contracts was at the time of this data extraction. This can help to determine if exclusion of these does leave out a large population that is behaving way differently than the dataset to train. As appears from Figure 10, it follows a similar shape as Figure 9. There is no extreme difference in durations longer than 60 months. Thus, this confirms the model will not totally focus on one group, short-running, and excluding a far majority of long-running contracts.

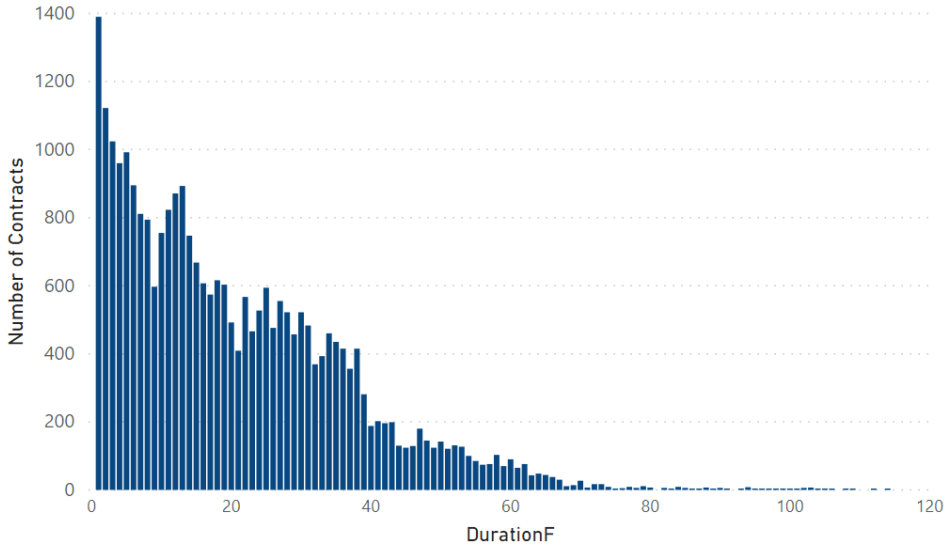


Figure 10. The number of contracts per fictive duration

So, the trends around duration and loan volume, have been depicted lightly. There is way more to discover, however not relevant to this study. More visualizations of trends and relations can be found in Appendix D.

About 63% of the contracts do have two applicants. In the other 37% of the contracts there is one applicant. The majority of these one-applicant loans has the gender of a male: 70%.

Looking at the living situation, about 80% has bought a house, while only 16% is renting a house. The remaining part is living in with someone, without having a fixed paying obligation. As said before, CompX does have strict approval rules. Having a house, makes a profile usually stronger and more reliable for providence of a loan. This might explain the majority has a house. The division between having kids or not is about 50/50. 52% of the applicants is married or has a registered partnership, while 30% is single. The rest is cohabiting without legal dependency and is free to choose to take the loan by themselves or together. The city of applicants is a categorical variable with a wide-spread: it appears there are 2258 different cities. About 14% is from Amsterdam and Rotterdam. Other popular cities are Almere and Den Haag, followed by Eindhoven, Utrecht and Arnhem. Only about 100 cities have more than 100 contracts. The large number of categories and the low representation value on many of them, makes it a weak input variable. Therefore, it is chosen to leave it out for the model creation. However, a more overarching variable is ZIP, while it is still related to the living place of applicant. There are 90 different digits within in this variable, as where 30 and 10, Amsterdam and Rotterdam area, logically are most popular.

Looking at the customer type, 54% of the contracts are a prospect, 26% are a customer, 15% is an ex-prospect and 5% is an ex-customer at the time of their application. However, as Figure 11 shows, on average customers do borrow the highest loan volume and prospects borrow on average the lowest amount of money. This can be declared by the fact, a customer who takes a new loan most of the time wants more money and raises their previous need of money which implies a higher loan volume.

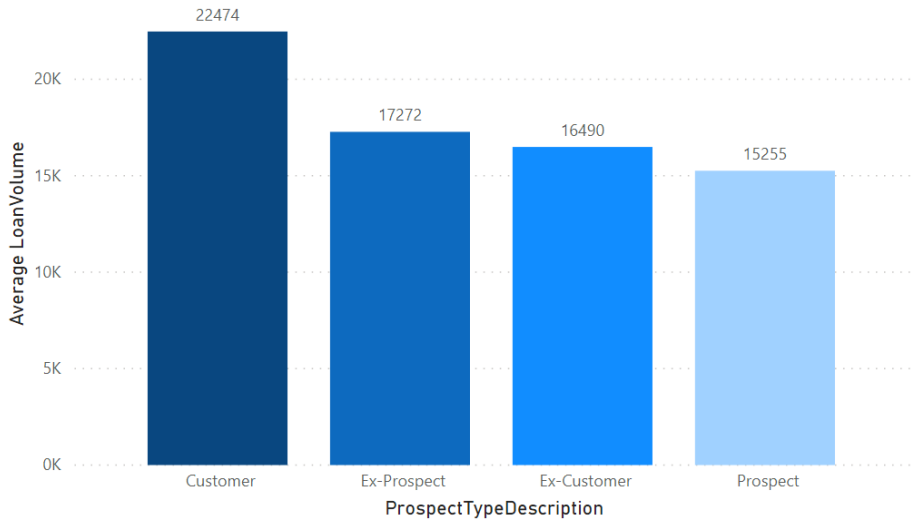


Figure 11. Average loan volume per prospect type

Related to the prospect type, is the Source of an application. Most contracts came in via phone (38%), 26% via WebAds and 7% via other paid web actions; 8% via a comparison site where commission fee is paid; and 21% came in via the website without direct relation to paid marketing activities. In the loan volume of these contracts there are no big differences found.

4.1.2 Repayment

From the actual duration, it becomes clear in most cases, the customer repays before the contractual end date. However, from this data the real repayment behavior cannot yet be extracted: how does the customer repay? Most likely there is at least one larger payment at the end of the period, referred to as the final payment. Next to these, there can also be extra payments. These are extra payments besides the monthly payment during the contract but are not enough to repay the loan fully. After such a payment there still is a net debt left, which will be repaid by monthly payments and optionally extra payments after.

From the dataset, it appears 2953 (9,5% of the total) of the contract ran as long as the contractual duration including the ones that defaulted. So, 91,5% of the contracts repays earlier. 57% of the contracts repaid the loan with one final payment, without any extra payment. Then, there is a group that does 1 extra repayment. However, from experience and sampling, this is not an extra payment during the contract always. It appears to be a correction on the final payment as well. For example, a customer does the final payment, does not pay the monthly payment anymore while the customer actually should. Then, the customer needs to do an extra payment as well, while this actually not really is an extra payment: the customer paid according the contract for the whole period and intended to repay the loan at once. From sampling and experience it is assumed in 50% of the cases the extra payment was not really an extra payment. So, the 57% of final payments is scaled up to 65,6% and 8,2% of the customer does one extra payments during the contract of an average value of €2909. 5,6% of the customers does two extra payments with an average value of €2408; 3,1% of the customers does three extra payments with an average value of €1831 and the rest, 8% does 4 or more extra payments where the average value decreases to €917. How many extra payment are done is visualized in the figure below.

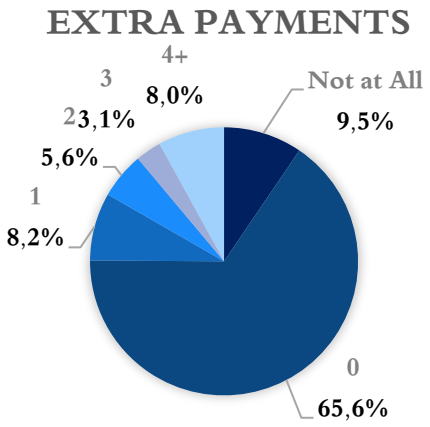


Figure 12. Circle diagram of extra payments excluding final payments

In conclusion, most customers do not keep the loan as long as the contract subscribes. The majority, about 66%, of these customers do repay 'normally' and then, all of a sudden, repay all at once. There is only a small group, 8%, that does extra payments on a more regular base by doing extra payments over three times.

4.2 Analysis of Profit and Data Elimination

Except for the expression of profitability as established in Section 2.3, the output of a loan can be measured in multiple ways. The repay behavior can also be expressed in terms of duration or by a type of comparison. A comparison between the actual duration and the contractual duration; or a comparison between the actual interest paid and the contractual interest. However, these output measures solely cannot distinguish one loan from another in a way one is more profitable to CompX and do not serve the objective. Therefore, only profit is chosen as an output measure y.

4.2.1 Profit and the Variables

This Section will outline some trends of individual variables in the light of the profit they generate. Not all variables will be discussed, only remarkable or relevant ones. The visualizations of all variables can be found at Appendix E.

Looking at the parts profit exists of, it is expected the Source of an application influences the profit. Namely, an application from a comparison site has to share a part of the revenue and thus is in general expected to yield a lower profit. This is confirmed by Figure 13.

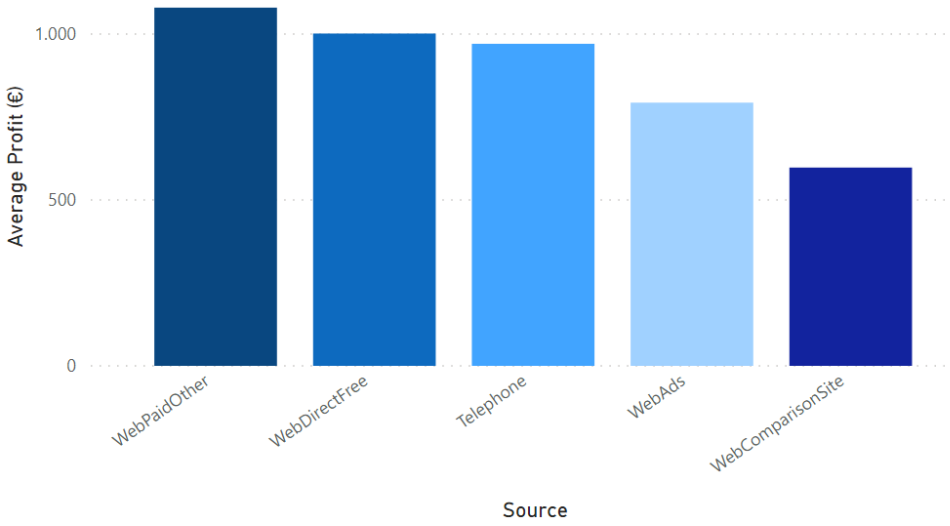


Figure 13. Average profit per source

WebDirectFree and Telephone have a comparable average profit; WebAds have a lower average profit. WebPaidOther seems to be most profitable, however the origin remains vague since it is a collection of many small marketing campaigns and is little underrepresented with 2261 contracts compared to the others. Figure 14 represents the average of profit per prospect type.

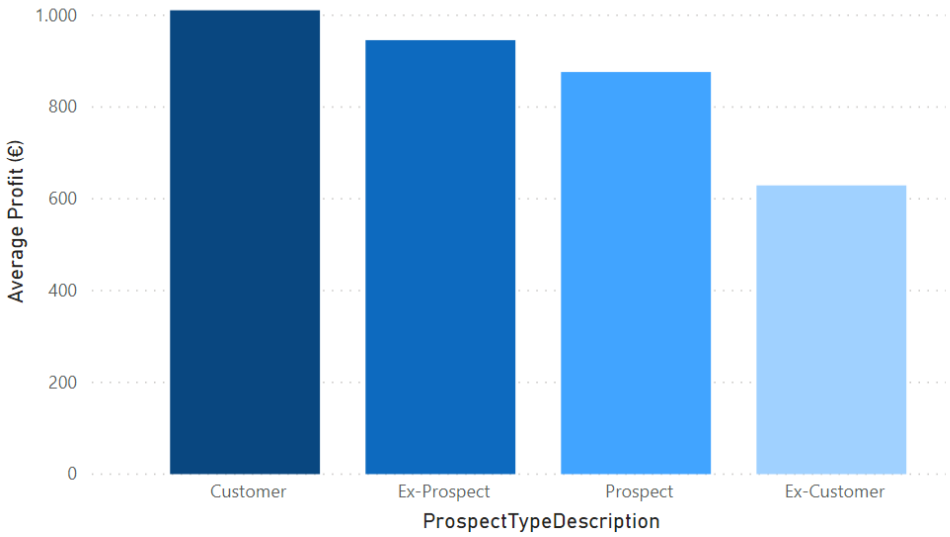


Figure 14. Average profit per prospect type

Remarkably, ex-customers yield less profit than the other prospect types. One reason could be, ex-customers are familiar with the ease of the application process, product conditions and the repayment at CompX. This might declare they lend easier for a short period of time, because of their positive experience. Other differences are not that big. The reason customers yield higher profit can be they usually retake a loan with higher loan volume; thus, borrow on average more money than when they arrived CompX as a prospect. The goal of the loan also shows some differences. However, the highest average profit can be found at the unknowns ('Onbekend') which is hard to deal with in future. Leaving these out of scope, the highest average profit accounts for takeovers (€1060). The lowest for transport purposes ('Vervoer') with an average profit of €790, followed by housing purposes ('Woning') with an average profit of €850. The fact takeovers yield highest average profit might be caused by the fact the applicant has multiple purposes: a takeover and some extra money for another purpose which increases the original loan volume. Only one purpose is technically denoted; the one that covers the biggest share of the loan volume.

The difference between man and women is not that big. If there are two applicants instead of one, the average profit increases roughly from €750 to €1000. The same trend is visible at the marital status. A person with marital status 'Alleenstaand' (Single) yield lower average profit and a legal connection between two applicants yields higher profit. Cohabiting ('Samenwonend') is in between these two groups, which makes sense: these types of contracts can have 1 or 2 applicants which is linked to the finding of the number of applicants. The average profit of an application with children is €100 higher than without children (€850). One reason could be, other costs for families with children are higher which makes them less likely to repay earlier. It can also partly be declared by the fact the average loan volume of families with children is higher (€18.800 vs. €16.160). For the age of applicants, there is roughly said a peak between the age of 40 and 60 that diminishes on both sides of the age range, as visualized in Figure 15.

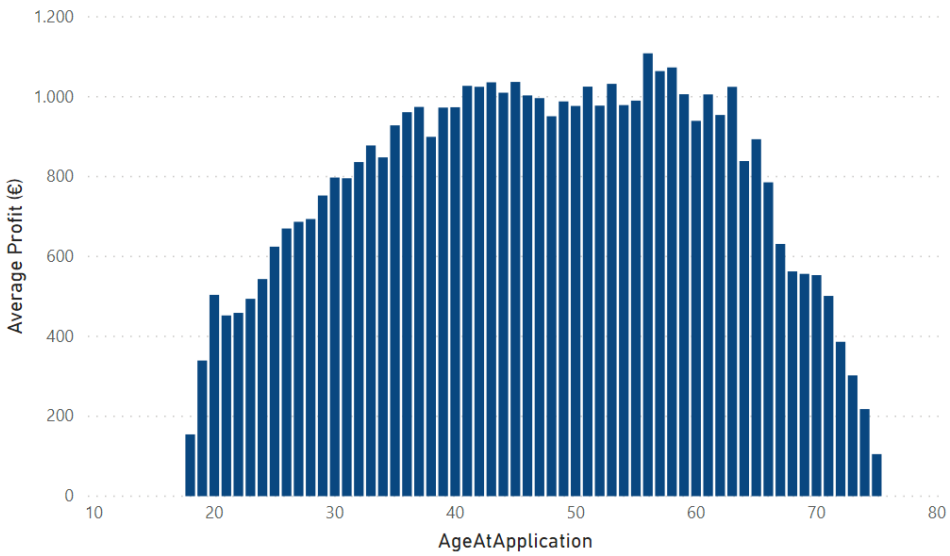


Figure 15. Average profit per age

A small extra peak of higher profit is seen around the age of 60. One reason could be, applicants in these ages have a stable and relative high income and therefore the loan volume might be higher. However, looking at the average loan volume per age in Figure 16, there is no clear indication that convinces this argument.

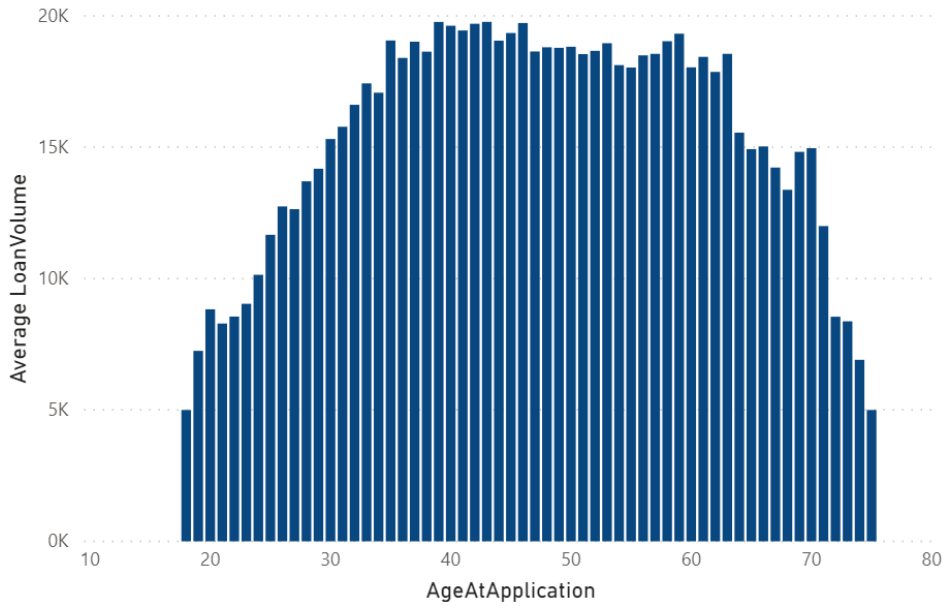


Figure 16. Average loan volume per age

Another explanation could be the income is less fluctuating, because promotion or increase income is less likely, assuming these people are at the end of their career. Therefore, customers at these ages are less likely to repay extra. The edges of the ranges also have lower loan volumes which can be declare the lower average profit.

Logically, there is a pattern between the bucket of the loan volume: the higher the bucket, the higher the average profit. The same accounts for the bucket of the contractual duration: the longer the duration, the higher the average profit. The ZIP code contains about 100 categories in which there are no major differences between the average profit per ZIP code. The income type does show some peaks for some values, but there is an issue with the underrepresentation of some categories. One occurs about 21000 times, and one other occurs about 10 times.

Other numerical variables are evaluated with a scatterplot. For most of these plots it is hard to detect at trend in relation to the profit, but all visualizations are shown in Appendix E. As expected, the relation between InterestC_Corr, DurationC and the LoanVolumeContract seem somewhat linear. The creditscore shows a similar pattern as the age: a peak at the median, as where the profit decreases around the range edges (see Figure 17).

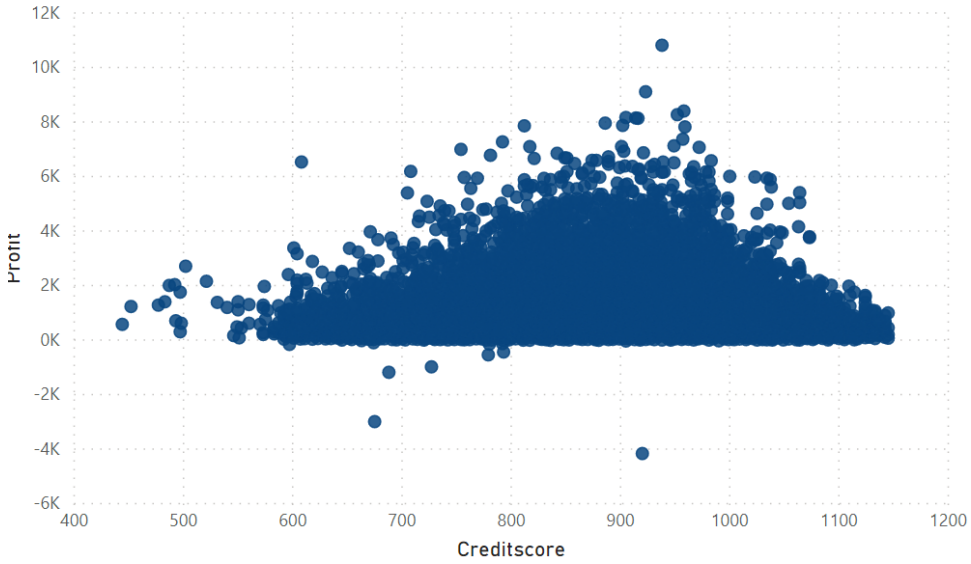


Figure 17. Scatterplot of profit by creditscore

Other relations of other variables are too vague and therefore not discussed further.

4.2.2 Less Profitable Loans

One of the research questions is which type of applications are less profitable. The profitability measure does not determine profit in terms of all revenue minus all costs. Otherwise, less profitable could be defined as a profit lower than 0 which actually is unprofitable. However, to give some insight into an answer on this research question, it is decided to stick to line of €400. The data is researched on characteristics of loans with a profit lower than €400 and if there are major differences on these characteristics with loans having a higher profit.

Overall, there is not one characteristic for less profitable loans that solely count for this group and does not occur at higher profit. One of the most distinguishing characteristics is the prospect type. If this is an ex-customer, about the half of them ends up with a profit less than €400. The same accounts for the

housing type, in about half of the cases the applicant is inhabiting ('Thuiswonend/Inwonend'), the loan is not profitable. Also, as intuitively expected, the loan volume bucket is one of the most distinguishing characteristics. 52% of the loans between €5.000 and €10.000 are less profitable. Under €6.000 this is even 67%. On the other hand, only 7,5% with a loan volume above €50.000 will be less profitable.

These are the most outstanding characteristics that can easily be detected by simple graphs. Out of this analysis, it seems not one characteristic will predict the profit easily, though there are some small differences detected already. The loan volume, source, prospect type for example seem to have influence. A more complex model that combines characteristic values might be better in findings patterns and making an accurate prediction about the profitability. This is what will be done for the remaining part of this Chapter.

4.3 Set–Up for Modeling

This Section will explain which methods for modeling will be used. Based on the findings of Section 4.1 and 4.2, some input variables are not used for the model. To be clear about this, Table 7 represents all variables from where each model evaluation is started. The numbering of the variables may be odd due to some intermediate skipping; however, this number is irrelevant to the research and just used as an identifier during the model testing.

Table 7. *Overview of input variables for the model*

Variable Name	**Variable Name**	**Variable Name**
1. DurationC	11. MaritalStatus	20. prcMLC
2. MLCBKR	12. ApplicantsNo	21. Income
3. LoanGoalCategory	13. AgeAtApplication	22. IncomeType
4. HousingType	14. LoanVolumeContract	24. RatioEX-IN
6. ZIP	15. InterestC_Corr	25. IN-EX
7. ProspectType	16. MonthlyPayment	26. PrcPay
8. Gender	17. OpenBKR	27. LVBucket
9. IND_Children	18. ExpenseSum	28. DBucket
10. Source	19. CreditScore	

The chosen models to test are based on the findings of Section 2.5 and what the software has to offer. The objective is to find the highest performance level by changing variables and tuning parameters, as available by BigML[7]. As mentioned before, a risk of decision trees is overfitting (Mehta, Rissanen, & Agrawal, 1995) and to examine this, testing is required. A model is trained on 80% of the dataset and tested on 20%

[7] All information around BigML is acquired from its manual https://static.bigml.com/pdf/BigML_Classification_and_Regression.pdf?ver=819a33b

of the dataset. This division a widely-used split in the literature reviewed, or similar researches handling machine learning such as the research from (Sarwar, Karypis, Konstan, & Riedl, 2000) and originates from the Pareto Principle (Mike Vladimer, 2018).

The three methods that will be tested are decision trees, decision forests and linear/logistic regression. Each method can be applied under multiple conditions, depending on their tuning parameters. A tuning parameters is a setting of a method. What these parameters are and what they do, will be discussed per method in Section 4.3.1 until 4.3.3. Besides these methods, deepnets, referred to as DNN at Section 2.5, can be an additional method, and there are some automatic optimization functions the machine learning software provides. As discussed before, DNNs are not preferred because of its black-box problem and the fact it is less intuitive. Though, this lack of understanding cannot be the reason to exclude the method by definition. Therefore, the method will be shortly tested as well to check whether the method may outperform other methods If deepnets do not outperform DTs, DFs and regression methods with big difference, deepnets are not preferred because of the black-box problem, the fact that they are hard to interpret. If deepnets do outperform DTs, DFs and regression methods with confidence, it should be considered to prefer deepnets and elaborate on this method nevertheless.

4.3.1 Decision Tree

The first machine learning technique that will be applied, is a decision tree. This section will elaborate on how the method is applied, but a deeper background understanding is provided at Appendix F1. The algorithm of DT applied is based on the CART model of Breiman, as discussed in Section 2.5. It can be used for regression problems and classification problems. For regression models it minimizes the mean squared error; for classification models it maximizes the information gain by each partition in the tree. This is a recursive process. Furthermore, BigML applies an in-memory algorithm (mtree) which is an anytime algorithm[8] with frequent updates. Each split, the algorithm looks for a new split that will improve the reliability of the prediction most. The end nodes of the model represent the final prediction. For classification problems, each end node will represent the class with the highest probability. For regression problems, each end node will represent one exact value. Therefore, the number of end nodes is way less than all possible outcomes, because the actual output is a continuous value. A DT for regression might have high errors, since it practically never will be able to predict each value: not all values are in the range of the predictive values. One variable can be split more than once. Creating a model, the algorithm already set the parameters to some values, that are referred to as the 'standard' settings: these are standard by the software used.

[8] With an anytime algorithm, a before-optimal stop of the model training will still be able to give a result (https://en.wikipedia.org/wiki/Anytime_A*)

Tuning parameters for a Decision Tree are:

- The number of input variables and which to include
- A pruning technique to prevent overfitting
    - No: no pruning will be applied
    - Statistical: considers every node for pruning
    - Smart: considers pruning if the node contains less than 1% of instances (standard)
- The node threshold to limit the growth of the model (a lower threshold will help overfitting, a higher threshold might be useful for a large dataset with many important fields)
    - Can vary between 2 and 2000 nodes, standard is 512
- Weighting option
    - Balance objective or not

The splitting process of a decision tree needs a stopping criteria to prevent overfitting and excessive grow. Pruning is a technique to prevent this overfitting. Pruning examines whether the split increases the confidence or decreases the expected error: if not, the split is pruned. The node threshold is another method to prevent overfitting and is related to the depth of the tree. A low threshold simplifies the model and the understanding of it but can also decrease the predictive power. The threshold is a stopping criteria for the maximum nodes of a tree. The weighting option is only applicable to regression problems. If activated to 'balance objective', the model takes into account some classes are less represented and some more in the data and scales all classes to equal representation.

4.3.2 Decision Forest

The second machine learning technique is the (random) decision forest. This section will elaborate on how the method is applied, but a deeper background understanding is provided at Appendix F2. A DF is a combination of multiple decision trees generated by different dataset samples. Therefore, the same as described above applies. The algorithm averages the single trees to get a final prediction. Tuning parameters are:

- The number of input variables and which to include
- The number of models
    - Can vary between 2 and 1000 trees, standard is 10
- A pruning technique
- The node threshold
- The randomize option makes the decision forest random. Configuration of number of input fields:
    - Not or Default: square root of the total number of input fields
- Weighting option

Compared to decision trees, there are two new parameters: the number of models and the randomize option. The number of models refers to how many trees should be created from different samples to construct one final model. The randomize function implies the input set will be randomly selected at each split. This creates a random decision forest.

4.3.3 Logistic and Linear Regression

The third modeling technique applied is the one of logistic or linear regression. Logistic regression is only applicable to classification problems and not to regression problems, contrary to DTs and DFs. The method uses maximum likelihood techniques to estimate coefficients. These coefficients can only be interpreted as field importance of that variable solely if inputs are independent. Standardization is applied for numeric fields to overcome incomparability due to scaling differences. To use categorical values, the variables are one-hot coded by standard setting. This implies for each category a new variable is created which is coded 1 for one variable and 0 for the others. However, to optimize and evaluate the model it is essential to know the statistics from the variables and the model, such as likelihood ratio, standard error, Z scores and p-value. This requires another type of field coding, namely dummy coding which implies one value will be set to zero and all other values use this a reference of control class. Changing the value, it is advised to change to a value that has a representative number. Furthermore, the statistics allow the model to output a significance of a variable. The standard significance levels are 0,1; 0,05 and 0,01. The tuning parameters are:

- The number of input variables and which to include
- The stopping criteria of the solver, referred to as Eps
    - Can vary between 0 and 1, standard is 0,0001
- Including the bias or not, inclusion is standard
- Regularization to avoid overfitting
    - L1: to force more coefficients to be zero
    - L2: to force all coefficient to be zero, standard
    - Strength (c): inverse of regularization strength, standard is 1

The stopping criteria Eps refers to a comparison value. If the difference between a current result and the result before that one is smaller than Eps, the model stops. A higher Eps value makes the model faster but will perform less in predictive power. The bias refers to the intercept of the logistic regression model (β_0) and can be included in or excluded from the model. Usually, inclusion results in better performance. Regularization can be tuned in two ways, by changing L2 into L1 or by changing the strength c. The functioning of L1 and L2 is already explained. The strength can be interpreted as the inverse of regularization. Higher values make the model perfectly fit the training set but makes the model worse in predicting new instances never seen before; small values make the model vague and not fitting the data patterns thus a bad predictor as well. The c value has to be greater than 0.

4.3.4 Application of the Methods

For all three methods and output measures, the same steps are taken to tune the model. The first model will contain all variables and the standard settings of the parameters. Next, the parameters as described above are tuned to examine the effect, still using all variables. From here the 'best' parameter settings are set and variables are eliminated. Variable elimination is done based on lowest field importance or based on 'no significance'. Each new model aims to improve the performance or keep it at the same level, with the objective to scale down. Scaling down in terms of variable inclusion simplifies the model and the user-friendliness which is assumed of great importance for a company and a real-life implementation. Besides the parameter tuning at the beginning, changing the parameters is also tested after variable elimination. The performance of a model with less variables may be influenced differently by changing the tuning parameters and therefore, the parameter is tuning is repeated again.

4.3.5 Performance of a Model

Evaluating the performance of a model can be done in multiple ways, as widely discussed by Parker (2011). For regression, BigML provides two types of measure: R^2 and MAE. For classification problems it offers about 10 performance measures. Interpretation will be unclear examining all of them. Therefore, it is decided to look only at few measures. Most input variables are not skewed; thus Phi-coefficient and F1-measure are less appropriate according to Parker (2011). One integrated measure and one point measure is chosen to evaluate the performance of models and compare them on this. The AP, also referred to as average PR, is chosen as the integrated measure. Recall and precision are important aspects thinking about the implementation of the model. Since this is multiclassification problem, the AP is averaged for all classes. Due to the many critics to the measure of accuracy, this performance measure is assumed not to be appropriate for this case either. For the point-measure, now recall is assumed to be most appropriate. This makes sense to the practical implication as well, as it should recall instances as accurately as well.

However, these performance measures do not include any form of 'the costs for a misclassification'. While intuitively this does matter. For example, if there 5 classes, where A represents the best class that has highest priority, E represents the worst class that has lowest priority. Now, assume an A instance is predicted as a B instance. Obviously, this prediction is not correct, but way better than predicting it as E instance. To compare models with inclusion of this 'how wrong' the wrong the prediction is, another evaluation method is applied. The method is based on the comparison of the value of wrong predictions versus the value of good predictions. The average profit per bucket is used to handle the costs of misclassification. There can be either an under-classification, for example an A-instance that is classified as an E-instance; or an over-classification, for example an E-instance that is classified as an A-instance. This measure might give slightly different results than the standard performance but takes into account 'how bad' the misclassification is. For example, if an A-instance is misclassified, it better be classified as a D-instance than an E-instance; better as a C-instance than a D-instance; better as a B-instance than a C-instance. So, a matrix for these misclassification costs is made for both classification models Y2 and Y3. It is assumed

over-classification is as bad as under-classification. Table 8 contains these costs for Y2 and Table 9 contains these costs for Y3.

Table 8. *Costs for misclassification for Y2*

Actual | Predicted	**A**	**B**	**C**
A	0 (€1763)	€1136	€1554
B	€1136	0 (€627)	€418
C	€1554	€418	0 (€209)

Table 9. *Costs for misclassification for Y3*

Actual | Predicted	**A**	**B**	**C**	**D**	**E**
A	0 (€2982)	€1589	€2199	€2485	€2773
B	€1589	0 (€1393)	€610	€896	€1184
C	€2199	€610	0 (€783)	€286	€574
D	€2485	€896	€286	0 (€497)	€370
E	€2773	€1184	€574	€370	0 (€209)

On the diagonal of each table the class is predicted well, thus the costs are 0. Between brackets the average profit of that class is represented, B_i. The calculation of the costs is made by taking the average profit of the actual class minus the average profit of the predicted class. The absolute value of this formula represents the costs for an over-classification and an under-classification. This costs is MC_{ij} where the actual class i is predicted as an instance of class j, as represented in Table 8 and Table 9. P_{ij} represents the percentage of instance i that is predicted as class j. Thus, the costs for misclassification are:

$$costs\ for\ misclassification = \sum_{i=A}^{\text{lowest } class} \sum_{j=A}^{\text{lowest } class} P_{ij} * MC_{ij}$$

The lowest class is in case of Y2 class C, in case of Y3 class E. Taking solely the absolute value of these costs on itself is sufficient in case of comparing one Y2 model with another Y2 model: the lower the costs for misclassification, the better. A perfect model would score 0 on these costs. However, comparing a Y2 model with a Y3 model is not that justified, because the costs for MC_{ij} differ between these two measures. Y3 can distinct more classes, and as a consequence has a higher average MC_{ij}. To overcome this problem, some benefit from the distinctive power should be taken into account in this score. If a model predicts 100% correct, it performs optimal. The optimal score for a model of Y2 is 2599 ($1763 * 100\% + 627 * 100\% + 209 * 100\% - 0 = 2599$); and the optimal score for a model Y3 is 5864. Looking at this value, Y3 now outperforms Y2 in comparison to the costs for misclassification. This is referred to as the benefit of the model. The relative performance of a model in this benefit-costs comparison, can be found by taking the score of a model that performs perfectly ($\sum_{i=A}^{lowest\ class} B_i$) and compare this to the costs of

misclassification. One option is to extract the costs for misclassification from the benefit, but this does not solve the problem where it is wanted to compare Y2 and Y3 models: Y3 can have a way higher score than Y2 by definition. Another option is to divide the cost for misclassification by the total benefit of a model. This is related to a question 'how bad is the model compared to how good it could have been', and makes it a relative measure and more suitable with the objective of comparing Y2 with Y3 models. Therefore a benefit-cost score (BC) will be used with the formula:

$$BC = 1 - [\frac{\sum_{i=A}^{\text{lowest } class} \sum_{j=A}^{\text{lowest } class} P_{ij} * MC_{ij}}{\sum_{i=A}^{lowest\ class} B_i}]$$

This has preference because both Y2 and Y3 have the same range within the score: between 0 and 1. The first part of the formula [$1 - \ldots$], makes the score more intuitive: scoring 1 is optimal, scoring 0 is not, which makes more sense talking about 'benefit-cost'.

This BC score is applied to each first model, each best model and some models around this best model, for each modelling method and Y2 and Y3, in addition to the measurements already described above.

4.4 Tuning and Evaluation of Models

This Section will cover the proposed modeling methods as described above for each of the three different output measures, in the order of finding the 'best' model with Y1, Y2 and Y3. The models are improved by changing the parameters as described in Section 4.3.4.

4.4.1 Profit Continuous (Y1): Decision Tree

Starting with Y1 and a decision tree, the first model is trained under the standard settings: (all variables; smart pruning; and node threshold of 512). The first step is to examine the performance by changing the number of variables. The model created is represented by an interactive decision tree, but contains a description of field importance as well. Some variables will not be assigned any importance, because the importance was too low and therefore the variables were automatically deleted from the model.

The first model is trained with 26 input variables (Table 7). However, variables [4, 8, 9, 12, 17, 21, 27] are automatically deleted from the model due to too low field importance. The top 5 of field importance is [14: 64,87% ; 15: 14,17% ; 10: 5,04% ; 7: 3,36% ; 16: 3,19%], the low 5 of field importance is [20: 0,43% ; 26: 0,31% ; 11: 0,11% ; 13: 0,11% ; 1: 0,1%]. The minimum depth is 4 splits, maximum depth is 9 splits. The first three splits are based on variables 14, 15 and 10 which is in line with the results of field importance. The end notes represent one exact profit value based on the average of the instances that belongs to that node. This creates large errors varying between €107,09 and €2240,16 on one end node. 23 end nodes contain a concrete value; other end nodes have a too big error. In addition, it appears some instances cannot 'follow' the tree to one end node. This is caused by categorical values such as 6: these have so many categories, the tree is not able to distinct. Thus, out of this, it is already expected the

performance of this tree will be bad. The performance is tested on the untrained dataset and scores R^2 = 0,33 and a Mean Absolute Error (MAE) of €479,45. This is considered a bad performance.

Next, variables are eliminated by the field importance and automated deletion to find which input variables contribute to the best model. The first 2 splits did not change. With these parameters and iteration of the variable elimination, the best model found has 3 input variables [14, 15, 16] with R^2 = 0,39 and MAE = €456,51. This last model has preference because of the simplicity, i.e. intuitively and easy to understand, and slightly better performance, however still not convincing.

Another parameter to tune is the pruning strategy. Statistical pruning does lead to worse results, with R^2 between 0,19 and 0,30. No pruning does improve the results, in combination with doubling the node threshold to 1024: variable [14, 15, 16] with R^2 = 0,41 MAE = €439,86. Scaling up the node threshold further than 1024 does not improve the result. Knowing a node threshold of 1024 and no pruning leads to the best performance, a change in the input variables might improve again. However, tests with inclusion of more variables, according to the field importance of earlier models, than the [14, 15, 16] does not improve either.

Concluding on these tuned parameters, it seems a model with variables [14,15,16], node threshold 1024 and no pruning has the best performance: R^2 = 0,41 and MAE = €439,86. However, this is still a weak predictive model. In perspective, this makes sense, since a decision tree can only end up in a few predictive outcome values (end notes) while the real value are way more varied.

BigML also offers an automatic optimization, where all parameters are optimized automatically. The algorithm assesses 200 different combinations of the parameters and chooses the best. It is a vaguer technique since it is hard to depict what is considered and what not. To validate the performance found by manual tuning is not totally outperformed by a 'smart algorithm' of automatic optimization, it is good to compare the results with automatic optimization. Including all variables, the performance R^2 = 0,29 and MAE = €496,54 and excluded all variables except for the top 5 mentioned before. Including the top 5 variables only in automatic optimization, the performance scores R^2 = 0,34 and MAE = €467,56. Automatic optimization with top 3 does neither improve the MAE, with R^2 = 0,41 and MAE = €450,00. There is no big difference, however the manual approach has better results and the approach is known. This confirms the model found cannot easily be outperformed and thus is assumed to be near optimal within this method.

4.4.2 Profit Continuous (Y1): Decision Forest

The second method is a decision forest. The standard parameter values for a decision forest are (all variables; 512 threshold node; 10 models; smart pruning; randomization = default). The first model performs R^2 = 0,35 and MAE = €480,90. The performance improves by using 100 models; increasing to 150 models' improvement is too small thus 100 models seems sufficient. Having set these parameters, a

model using variable [1, 14, 15, 16, 27] or [14, 15, 16, 27] performs best, where using 4 variables is preferred due to the reason of simplicity. The performance is R^2 = 0,43 and MAE = €456,41. Scaling up the number of models does not improve the results that much, while the computation time is way higher. Therefore, no further changing on this parameter will be tested. Changing the randomization into no randomization does improve the model to R^2 = 0,45 and MAE = €446,51. The last parameters that can be tuned are the number of threshold nodes and the pruning technique. It appears the best performance is found with a node threshold of 2000 and having no pruning technique, resulting in R^2 = 0,50 and MAE = €414,36. An inclusion of more variables does not improve the model. Therefore, the best model found for continuous profit, using a decision forest, is having variables [14,15,16]; 2000 node threshold; 100 models; no pruning; no randomization.

4.4.3 Profit Continuous (Y1): Linear Regression

The last method to test it linear regression. Such a model can take all values in a range, contrary to the methods before. The only parameters of a linear regression model are the input variables and the bias. The first model with all variables has a performance of R^2 = 0,23 and MAE = €556,70. Deleting the variables where significance is higher than 0,1, does not improve the model. Stepwise more variables are deleted based on a lower significance level of 0,05 and 0,01. The best performance is found using 15 variables [1,2,3,7,9,10,12, 13, 14,16,17,19,21,24], R^2 = 0,35 and MAE = €482,68. A model with 15 variables does not work intuitively, so scaling down to 10 by deletion of [2,3,17,19,21] yields R^2 = 0,34 and MAE = €489,18. This performance is slightly worse but more intuitive. Scaling down, having less variables does make the performance, e.g. with [1,7,9,10,12,14,16,24], to R^2 = 0,33 and MAE = €489,84 or even less, [1,7,10,14,16,24], to R^2 = 0,33 and MAE = €491,55. It appears the effect of inclusion or exclusion of the bias is neglectable. Concluding on this method, the other methods outperform linear regression. These tests show linear regression is less preferred for Y1.

4.4.4 Profit Continuous (Y1): Other Methods

Besides these three methods discussed, there are other methods. A DNN on Y1 performs with BigML offers a full optimization method where it analyzes 200 different parameters and methods (DTs, DFs, regressions and deepnets), evaluates and represents their scores. This is called OptiML. However, as discussed before, there is also black-box problem with this method. Comparison of the performance can still be valuable to the analysis, such that it can be concluded the more straight-forward methods as described above are performing well and no better solutions are excluded by the argument of complexity. OptiML finds the best performing: a deepnet with 128 evaluated networks. The top 5 of field importance is [15: 42,41% ; 16:11,50% ; 26: 7,10% ; 2: 4,58% ; 10: 3,89%]. The performance is not better than found manually: R^2 = 0,42 using a random decision forest with 1647 nodes. An OptiML can also run with less variables. The first OptiML finds variables [10,14,15,16,17,19] to be most important, therefore also an OptiML is created with these variables.

4.4.5 Profit in 3 Buckets (Y2): Decision Tree

The first model is trained under standard conditions: smart pruning, 512 node threshold and no balanced objective. It auto-deleted variables [4,9] and had an average PR of 0,5988 and recall of 54,30%. Scaling down and tuning the node threshold and pruning technique, the best model is found using variable [15,16] with an average PR of 0,6294 and recall of 56,00%. The node threshold was 1024 and no pruning technique was used. Automatic optimization with all variables performs worse. Automatic optimization with the top 6 variables does neither outperform the model found, with an average PR of 0,5711 recall of 56,10%. A confusion matrix gives easy understandable insights into what it has predicted on the top row versus the actual values on the left column. The confusion matrix of this model is shown in Table 10.

Table 10. *Confusion matrix of the best model for 3 profit buckets using decision trees*

Actual | Predicted	**A**	**B**	**C**	**Actual**
A	**1754 |76,3%**	329 |14,3%	215 | 9,4%	2298
B	636 |31,9%	**837 |42,0%**	519 |26,1%	1992
C	446 |23,1%	523 |27,1%	**961 |49,8%**	1930
Predicted	2836	1689	1695	6220

The percentages should be read horizontally: 76,3% of the actual 'A' instances are predicted as A; 14,3% of the actual A instances are predicted as B etc. If the prediction is false for an A instance, it is preferred to predict a B instead of a C. Balancing the objective does not improve this matrix.

Besides the standard performance measures, we also want to evaluate the models on the benefit-cost score. A decision tree with no pruning; 1024 node threshold; no balancing objective and variable [15,16] appears to have the highest standard performance results and a BC of 0,5183. However, the benefit-cost score prefers the inclusion of more variables, namely [10,14,15,16]. The BC is 0,5227.

4.4.6 Profit in 3 Buckets (Y2): Decision Forest

The next method to apply on Y2 is the decision forest. The first model is trained under standard conditions: smart pruning, 512 node threshold, default randomization and 10 models. This model performs with an average PR of 0,6041 recall of 55,10%. Changing the number of models to 100 does improve the model (an average PR of 0,6180 and recall of 55,80%), but further scaling has no positive influence on performance. Elimination of variables based on field importance, found the best model performing with variables [14,15,16]. The model becomes better with no randomization and maximizing the node threshold to 2000. Changing the pruning technique does not improve either. So, the best-found model performs with an average PR of 0,6592 and recall of 58,50%. Automatic optimization with all variables or the top 6 does not outperform this model. The confusion matrix of this model is represented in Table 11.

Table 11. *Confusion matrix of best model for 3 buckets using decision forests*

Actual | Predicted	A	B	C	Actual
A	**1939 |83,7%**	210 |9,7%	149 | 6,6%	2298
B	688 |34,2%	**868 |43,7%**	436 |22,1%	1992
C	461 |24,8%	552 |28,2%	**917 |47,0%**	1930
Predicted	3088	1630	1502	6220

This model is better in predicting of A instances, better in predicting B instances, but worse in C instances compared to the best model of the decision tree. However, the average recall is higher than the model of decision trees.

In addition, we want to evaluate the decision forest models with the benefit-cost score. The best model found, with smart pruning; 100 models; 2000 node threshold; no balancing objective and variable [14,15,16] has a BC of 0,5463. The benefit-cost score confirms this is the best model as well, since no other model scores better.

4.4.6 Profit in 3 Buckets (Y2): Logistic Regression

The first model is executed under standard parameter settings: Eps = 0,0001, bias on, auto dummy coding, L2 Regularization, no balanced objective. It performs an average PR of 0,5743 and recall of 53,50%. Changing the Eps to 0,00001 does not influence the performance at all but does increase computational time. Eps = 0,0001 is sufficient. Turning off the bias leads to comparable results, thus it will be kept on during variable elimination. The same accounts for L1 Regularization. The dummy coding will be explored later. Based on the first model and the statistical output, variable elimination is applied by looking at the p-value and the significance of the variable. The standard significance level is 0,1. This is iterated, and the best model found has parameter settings Eps 0,0001; bias on; auto dummy coding; L2 Regulation; strength of 500 and no balanced objective. The 15 variables included are [1,2,3,7,10,12,14,15,16,19,20,21,24,26,28] and the performance measures are an average PR of 0,5751; and recall of 53,4%. The differences with the first model are very small or even assumed neglectable. The confusion matrix of the best-found logistic regression is shown in Table 12.

Table 12. *Confusion matrix of best model for 3 buckets using logistic regression*

Actual | Predicted	A	B	C	Actual
A	**1801 |78,4%**	274 |11,9%	223 | 9,7%	2298
B	761 |38,2%	**564 |28,3%**	667 |33,5%	1992
C	519 |26,9%	376 |19,0%	**1035 |53,6%**	1930
Predicted	2836	1689	1695	6220

This model is worse in predicting A and B instances, but better in predicting C instances. However, the average recall is worse than the other predictive models and therefore logistic regression is not preferred for Y2.

Evaluating this method with the benefit-cost score, it still performs week compared do DT and DF. The model with Eps = 0,0001; with bias; L2 Regularization ; c =500; no balanced objective and variables [1,2,3,7,10,12,14,15,16,19,20,21,24,26,28] appears to have the highest standard performance results and has a BC of 0,4794. Other models in this method do not perform better on benefit-cost, thus it can be concluded logistic regression is not appropriate for this classification problem.

4.4.7 Profit in 3 Buckets (Y2): Other Methods

Besides these three methods, also OptiML and the method of deepnets is applied to Y2. The OptiML evaluated 101 models and found a decision forest with 1652 nodes, 118 models, and a balanced objective to be the best. The performance has an average PR of 0,6413 and recall of 57,1%. This OptiML model does not outperform the models found using decision trees or decision forests. The deepnet performs with an average PR of 0,6910 of and recall of 54,8%. The benefit-cost score of this model is 0,4988. As can be concluded, both these models do not outperform the models found above.

4.4.8 Profit in 5 Buckets (Y3): Decision Tree

The third output measure is the division of profit in 5 buckets. Starting with DT, the first model is trained under standard conditions: smart pruning, 512 node threshold. It auto-deleted variable [27] and had an average PR of 0,4312 and recall of 41,70%. Scaling down on variables included and tuning the node threshold and pruning technique, the best model is found using variable [14,15,16] with an average PR of 0,4805 and recall of 44,90%. The node threshold was 1024 and no pruning technique was used. Automatic optimization is tested as well, with all variables and just the last 6, but these models perform worse. The confusion matrix of this model is shown in Table 13.

Table 13. *Confusion matrix of best model for 5 buckets using decision tree*

Actual | Predicted	**A**	**B**	**C**	**D**	**E**	**Actual**
A	**374 |58,7%**	228 |35,8%	16 |2,5%	2 |0,3%	17 |2,7%	637
B	188 |14,0%	**831 |61,7%**	100 |7,4%	6 |29,0%	222 |33,5%	1347
C	84 |6,2%	463 |34,4%	**411 |30,4%**	15 |1,1%	377 |27,9%	1350
D	38 |4,1%	238 |25,6%	177 |19,0%	**139|14,9%**	339 |36,4%	931
E	63 |3,2%	399 |20,4%	271 |13,9%	78 |18,5%	**1144 |54,0%**	1955
Predicted	747	2159	975	240	2099	6220

As the table above suggests, the model is relatively good in predicting A, B and E instances, however does predict instances as a B and E instance unfairly as well. The prediction of D is worst. This might be caused by the fact the classes are quite unbalanced. Applying a balanced objective results in Table 14. The new

model increases the average recall to 47,0%, and becomes better in recognizing A instances, worse in B instances, a little bit better in C instances, better in D instances, worse in E instances.

Table 14. *Confusion matrix of best model for 5 buckets using decision tree with balanced objective*

Actual | Predicted	A	B	C	D	E	Actual
A	**531 |83,4%**	91 |14,3%	8 |1,3%	4 |0,6%	3 |0,5%	637
B	426 |31,6%	**637 |47,3%**	113 |8,4%	72 |5,3%	99 |7,3%	1347
C	234 |17,3%	328 |24,3%	**442 |32,7%**	174 |12,9%	172 |12,7%	1350
D	97 |10,4%	164 |17,6%	207 |22,2%	**292 |31,4%**	171 |18,4%	931
E	177 |9,1%	305 |15,6%	287 |14,7%	401 |20,5%	**785 |40,2%**	1955
Predicted	1465	1525	1057	943	6220	6220

Also for Y3, the models are evaluated on benefit-cost score. The best decision tree with no pruning; 1024 node threshold; no balancing objective and variable [15,16] scores 0,5521 on the benefit-cost score.

4.4.9 Profit in 5 Buckets (Y3): Decision Forest

The first model is trained under standard conditions: smart pruning, 512 node threshold, default randomization and 10 models. This model performs with an average PR of 0,4627 and recall of 40,8%. Changing the number of models to 100 does improve the model (an average PR of 0,5218 and recall of 46,3%), but further tuning on the number of models has no positive influence on performance. Elimination of variables based on field importance, found the best model performing with variables [14,15,16]. The model becomes better with no randomization and setting the node threshold to 1536. Changing the pruning technique does not improve either. So, the best found model performs with an average PR of 0,5218 and recall of 46,3%. Automatic optimization with all variables or the top 6 does not outperform this model. The confusion matrix of the best model found, without balancing the objective, is represented in Table 15.

Table 15. *Confusion matrix of best model for 5 buckets using ensemble*

Actual | Predicted	A	B	C	D	E	Actual
A	**376 |59,0%**	221 |36,3%	15 |2,4%	2 |0,3%	13 |2,0%	637
B	164 |12,2%	**872 |64,7%**	100 |7,4%	3 |0,2%	208 |15,4%	1347
C	79 |5,9%	421 |31,2%	**427 |31,6%**	19 |1,4%	404 |29,9%	1350
D	33 |3,5%	210 |22,6%	165 |17,7%	**127 |13,6%**	396 |42,5%	931
E	60 |3,1%	353 |18,1%	247 |12,6%	72 |3,7%	**1223 |40,2%**	1955
Predicted	712	2087	954	223	2244	6220

However, as already found with the decision tree, balancing might lead to better recognition of the classes. Table 16 represents the confusion matrix of the same model from Table 15 but with balanced objective.

Table 16. *Confusion matrix of best model for 5 buckets using ensemble with balanced objective*

Actual | Predicted	A	B	C	D	E	Actual
A	**533 |83,7%**	88 |13,8%	12 |1,9%	4 |0,6%	0 |0%	637
B	380 |28,2%	**714 |53,0%**	131 |9,7%	57 |4,2%	65 |4,8%	1347
C	190 |14,1%	372 |27,6%	**480 |35,6%**	157 |11,6%	151 |11,2%	1350
D	86 |9,2%	181 |19,4%	227 |24,4%	**281 |30,2%**	156 |16,8%	931
E	140 |7,2%	331 |16,9%	348 |17,8%	355 |18,2%	**781 |39,9%**	1955
Predicted	1329	1686	1198	854	1153	6220

The model only becomes worse in recalling class B, while the recall of other classes are improved highly. Therefore, this is the preferred model for Y3 compared to DT.

Looking at the benefit-cost score, this model is not the best: it scores a BC of 0,5623. Evaluating other models on this benefit-cost score, another model seems more appropriate scoring 0,5638. Though, the difference is very small, it still has preference. This model has a 1536 threshold node, 100 models; a balanced objective; no randomization and no pruning. Variables included are [14,15,16].

4.4.10 Profit in 5 Buckets (Y3): Logistic Regression

The last method to apply is logistic regression to Y3. The first model under the standard parameters performs average PR of 0,5218 and recall of 36,8%. Tuning and elimination leads to a model performing with average PR of 0,4001 and recall of 41,0%. This is a worse performance than found before and it has a many more variables ([1,3,7,10,13,14,15,16,18,20,27,28]). Deleting variables does not improve the model. The confusion matrix looks like Table 17.

Table 17. *Confusion matrix of best model for 5 buckets using ensemble*

Actual | Predicted	A	B	C	D	E	Actual
A	**460 |72,2%**	153 |24,0%	12 |1,9%	0 |0,6%	12 |1,9%	637
B	352 |26,1%	**603 |44,8%**	187 |13,9%	18 |1,3%	187 |13,9%	1347
C	190 |14,1%	358 |26,5%	**268 |19,9%**	125 |9,3%	409 |30,3%	1350
D	97 |10,4%	167 |17,9%	154 |16,5%	**93 |10,0%**	420 |45,11%	931
E	168 |8,6%	355 |18,2%	252 |12,9%	93 |4,8%	**1087 |55,6%**	1955
Predicted	1267	1636	873	329	2115	6220

For Y3 with a logistic regression, the best model found is also the best for the benefit-cost score. The score is 0,3467.

4.4.11 Profit in 5 Buckets (Y3): Other Methods

The OptiML function finds the best model as a decision forest with 544 nodes and 76 models. The performance of this model is an average PR of 0,4554 and recall of 46,9% and a BC of 0,5281. A deepnet performs with an average PR 0,4310 of and recall of 42,0%. The corresponding benefit-cost score is 0,5199. Both do not perform better than models already established.

4.4.12 Evaluation on all Models

From the analysis above, there are about 300 models tested varying in the profit output type; modeling method and the tuning parameters.

Each Section concludes on which is the best model under that method and that output. All these best models and their performances are summarized in Table 18. Model from Section describing 'Other methods' are not included in this comparison, because they never appeared better.

Table 18. *Summary of best models and their results*

Output	**Method**	**R^2**	**MAE**	
Y1	DT	0,41	439,86	
Y1	DF	0,50	414,36	
Y1	Linear Regression	0,35	482,68	
Output	**Method**	Average PR	Recall	BC
Y2	DT	0,6294	56,0%	0,5227
Y2	DF	0,6592	58,5%	0,5463
Y2	Linear Regression	0,5751	53,4%	0,4794
Y3	DT	0,4805	44,9%	0,5521
Y3	DF	0,5218	46,3%	0,5638
Y3	Linear Regression	0,4001	41,0%	0,3467

The objective is to choose one of these models as the best and this one will serve as the predictive model that should be validated in the next Chapter. While Y1 might have preference since the output measure is most specific of all of them, the results are disappointing. General statistic rules state that R^2 equal 0,5 or lower, is not weak. In addition, the error is quite big. One argument for choosing a classification model is that classification models are more easy to change into priority labels. Out of this, it is chosen to no select a Y1 model. Now, comparing Y2 and Y3 models, Y2 using a DF has preference based on the standard performance measure of average PR and DF. However, with inclusion of BC, a DF of Y3 has preference. Besides, a classification system with 5 classes can distinguish more precise than a classification system with 3 classes. Looking at the division in classes and there ranges, Y3 does distinguish loans that are highly profitable while Y2 does not. Based on these arguments, the model of Y3 using a DF is chosen.

The model selected needs input of 3 variables. These variables are the loan volume, the monthly payment and the interest that should be paid according to the contract. If the loan volume is above €50.000 it

predicts an A instance in 97,7% of the cases; a loan volume between €25.000 and €50.000 is predicted as an A in 60,1% of the cases, but also 32% of these instances were predicted as a B instance. Only 0,002% of the instance with a loan volume between €5.000 and €10.000 were predicted as an A instance. So, as expected, these type of loans will about never get a high priority. About 41,4% of these instances are predicted as an E instance, where the about half of these instances also has an InterestC below €412. As the InterestC increases, the predictions shift towards B, where above €1200 the mots predictions are for a C instance etc. There are some exceptions, but as the InterestC increases per loan volume (bucket), there is an increase in priority label, towards A. The monthly payment shows a similar pattern: the higher this is, the higher the probability for a higher class, especially taken into account the loan volume.

5. Implementation

This new priority label for each application can be implemented in the process. To recall the underlying wish of the MT, the priority rule is not meant to ignore applications beforehand but to give more attention or faster help to the once that expect to deliver more profit.

There has been executed another pilot while this research is undertaken. For this pilot, a third party was involved for the extra work effort and consulting services. The pilot took place at KCC to find out how applicants could be reached better and consisted of two pilots. For both pilots the applications were split in two groups. With the first pilot, one group was called according to the regular process, with the first call within 24 hours, and a second call in a few days, by the KCC. After that, the application is automatically cancelled in 30 days if the applicant does not call CompX. The other group was called twice by the KCC as well, but was then called by a third party for maximum 5 more times. The outcome of this research was simple: the extra calls had no effect. The conclusion of the pilot is, that if an applicant really wants a loan at CompX, they will get in contact anyway.

More promising was the second pilot. With the second pilot a new process step was added: calling the applicants who had received an offer, above €10.000, but did not return yet. The objective was to test if calling an applicant who did not return within the first 7 days, leads to a higher offer return rate. Currently, the only thing CompX did after sending the offer was sending an email after 4 days, as a reminder to return the offer. Under these circumstances, so without calling, about 67% of the applications is returned in 7 days. For the test, two groups of applications were made: one group followed the regular process with only the email, the second group was called in addition. The group was called by the third party after 7 days if the application was not returned yet, as many times as needed until the applicant picked up the phone. The result was that in the first group about 29% returned the offer somewhere between the 7 days and 30 days, in the second group this was about 43%. This is an increase in the offer return rate.

Several conclusions were made based on this second pilot. Based on the increase in offer return rate, the consulting advise of the third party was to call the applicants that have received an offer but do not return in the first 7 days. Another conclusion was CompX's expertise in these calls is required thus it is not advised to outsource this on the long term. Even if the call ends up in a cancellation, there still is a benefit from the call. The conversation can lead to more exact information which can help improve service or product conditions. In addition, applicants who were still in doubt and cancelled in the end, are expected to return to CompX in future, because of the good experience in service quality. However, this effect will be harder to measure.

Having this information and having a model that predicts the added value, i.e. relative profitability, of a loan, CompX likes to experiment with the implementation of this new activity into their existing process. In combination with this new step, the priority model as concluded to in Section 4.4.12 is simulated.

Section 5.1 suggests a recall scheme and examines the expected increase in performance of implementing the priority label. This is tested with a simulation of an A/B test. An A/B test implies there are two groups, where only one factor changes. Comparison of these two groups leads to insights about the effects of changing that one factor. After the explanation of the set-up and the results of the simulation, the A/B test could be executed in real life. How this looks like, is explained in Section 5.1.3. Section 5.2 suggests other options to implement the priority labels.

5.1 Simulation at Call after Offer

A more detailed description of implementing the 'Call after Offer' will be discussed. The implementation should first be tested by A/B testing: one group will apply the new suggested priority rule, another group will have all same circumstances except for the priority rule. All conditions for the A/B test will be described in Section 5.1.1. Based on these conditions, and some assumptions, the A/B test is simulated as well. With this objective is to confirm the benefit of implementing the priority rule. Section 5.1.2 dives deeper into executing this A/B test in real life. Section 5.2 elaborates on more possibilities for applying the priority rule, without simulating it.

5.1.1 Set-Up for the A/B test

As described before, there have to be made two groups to execute an A/B test. One group is referred to as Group A, the other group is referred to as Group B. For clarification: an A instance does not refer to this group but to the priority label ('An A instance of Group A' or 'A B instance of Group B' etc.). The priority model will be applied to Group B, while Group A will not use the priority model. With this A/B test, both groups will work at the same time such that circumstances remain constant and as similar as possible. If one test group is executed for a few months, and the other group the months after, the differences might also be declared by differences in time, e.g. seasonality. This is unwanted, therefore both groups with different priority rule have to be executed simultaneously.

The first condition that has to be set for recalling applicants, is the scheme of when to call. Calling applicants directly after they have received the offer is useless. The applicant should get some time to think about the offer and collect the documents needed. This is also why CompX sends the reminding email after 4 days. The offer is sent at $t = 0$, so sending the email will remain on $t = 4$ which is automatically sent by the system. Now, at $t = 7$, KCC will call all offers that did not return or cancel yet. This is the first call. To recall, after 30 days the offer will be cancelled automatically, so an applicant can only be called between $t = 7$ and $t = 30$. Avoiding stalking behavior, the applicant will be called at most 5 times within these 30 days. Also, appeared from the previous pilot, most applicants did not need more than 5 calls in order to reach them. Therefore, the calling scheme consists of 5 calls at most for each applicant. In addition to the call at $t = 7$, the applicant is called at $t = 10, t = 15, t = 20, t = 25$. The time in between these calls is assumed to be sufficient in order to give the applicant enough time to respond. Only if the applicant did not respond at the first call, the applicant will be called for a second time at $t = 10$. If the applicant

does not respond again, it will be called for a third time at $t = 15$. If the applicant does not respond at the third call, it is called for a fourth time (at $t = 20$) and if there is no response again, it is called for the last time at $t = 25$. So only if an applicant does not respond in 20 days after four calls, it is called for a fifth time. Calling after $t = 25$, it becomes highly unlikely the applicant is the interested and the responding time left is very short: only 5 days. The probabilities of responding, no responding, returning and cancelling will be discussed later.

If all applicants are called at Group A and Group B, the effect of the priority model will most likely not be visible, because all are called anyway so the order of calling based on priority will not have any influence. If there is a capacity constraint, a decision has to be made about who to call and who not. It is expected that prioritizing based on the priority model in this case will improve. This should be test with the A/B test and therefore a capacity constraint is assumed. This capacity constraint is only set at the arrivals at the first call. The workload on one day consists of 5 parts: the applications that are called for the first time; second time; third time; fourth time; and fifth time. So, the constraint is set on the arrivals at the 'first time queue' (A_1) only. Based on historical data, it is known that 67% of the offers is returned at after 7 days; 5% is cancelled; so the remaining part is A_1: 28% of the offers sent (O) at $t = 0$. The average A_1 over the last year (1st of October 2018 and 31st of September 2019) was 18 applications. This A_1 has to be divided over two groups: Group A with A_{1A} and Group B with A_{1B}. Therefore, the constraint is set to $A_{1A} \leq 6$ and $A_{1B} \leq 6$. These will add up to a handling capacity of 12, while the average total inflow is 18. On a few days $A_1 \leq 12$, then all applications will be called. This implies a decision has to be made on most days, on which applications to call and which not. This A_1 will be equally divided into two, and both groups have to decide which 6 applications to call. Group A will select these randomly, Group B will select the applications with highest priority label. Lastly, one practical fact should be covered: KCC is not open at Sundays. The schedule as described might not be possible to follow if one of the recall dates is a Sunday. The solution is to postpone the call with one day, to Monday, but remain the same interception time. The recall date will be on a Sunday at most 1 time during the recall scheme. So, for example if the second call at $t = 10$ is a Sunday, the second call will take place at $t = 11$, the third call at $t = 16$, the fourth call at $t = 21$ and the last call at $t = 26$.

With this recall scheme and the inflow restriction, there can be made an estimation about the workload. The workload depends on the number of offers sent per day; the probability the applicant has to be called again; and the schedule of recalling. For the simulation, real data is used from one year. Next, for this expectation it is assumed, if the applicant is reached and says to return the offer, the application will diminish from the workload. When the application is cancelled, this action is immediately executed by the KCC employee and the application will not flow through. So, only unreached applicants can pass on to the next queue. The process flow of the first call is represented with Figure 18, but the flow around the 'first call queue' can be repeated until the fifth call.

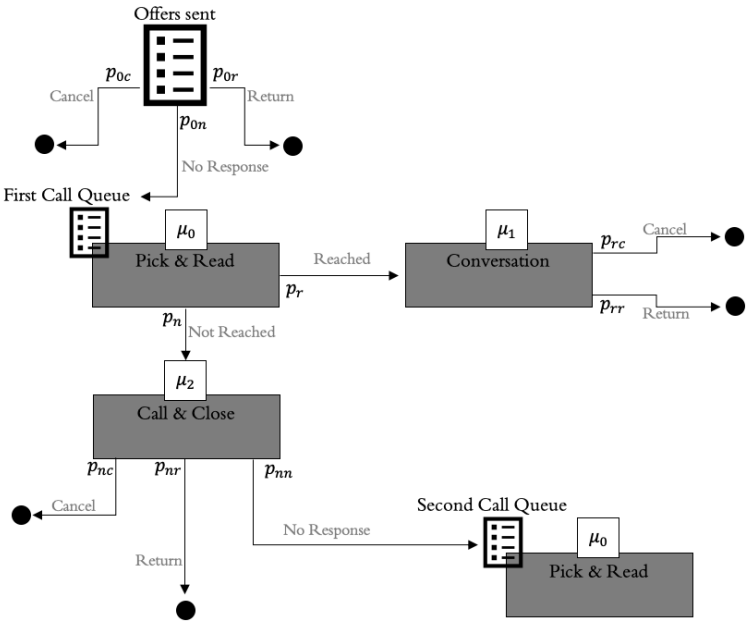


Figure 18. Process flow of the first call after offer sent

Within the first 7 days, 5% of the applicants cancels on own initiative ($p_{0c} = 0{,}05$); 67% returns the offer ($p_{0r} = 0{,}67$); 28% does not return or cancel ($p_{0n} = 0{,}28$). To recall, in the current situation 75,2% of the offers is returned in 30 days: 8,2% of the offers is returned between day 7 and day 30 if CompX is not calling them. Now, the offer return rate is expected to increase to at least 79%: the offer return rate between day 7 and day 30 should increase to 12%. From p_{0n}, it is assumed 28% of the offers sent will arrive at the queue for the first call. Mathematically this look like $A_1(t = 7) = 0{,}28 * O(t = 0)$. From this 28% arriving at the recall schedule should return 42,9% (12%/28% = 0,429) of the application within the 30 days and the 5 calls, to end up with a return rate of 79%. From the third party, it is known, 44% of the applicants were not reached by phone in 5 calls, but this does not imply the application is not returned. An applicant might return the offer without answering the phone. From Figure 18, it becomes clear a few probabilities are involved and all of these together do determine the workload on each day. The probabilities do differ per call. It is less likely an applicant wants to return the offer at the fifth call than at the first call. All the probabilities after each call are determined in Table 19.

Table 19. *Probabilities at each call after offer sent*

Result | Call		**1**	**2**	**3**	**4**	**5**
Not Reached	p_r	0,8425	0,8350	0,8450	0,8600	0,8600
Reached	p_n	0,1575	0,1550	0,1550	0,1400	0,1400
Not Reached: No Response	p_{nn}	0,6740	0,7098	0,07605	0,7740	0,0000
Not Reached: Cancel	p_{nc}	0,0337	0,0752	0,0704	0,0774	0,8170
Not Reached: Return	p_{nr}	0,1348	0,2255	0,2113	0,2580	0,0430
Reached: Return	p_{rr}	0,1181	0,0990	0,0775	0,0420	0,0070
Reached: Cancel	p_{rc}	0,0394	0,0660	0,0775	0,0980	0,1330

These probabilities are constructed based on experience, data and intuition, because they are not known exactly. The process has never been executed in exactly this manner before. The calculation of each probability can be found at Appendix G. For the A/B test, the probabilities are the same for both groups. In addition, these probabilities lead to the objected target rate of return for this process, namely 0,427 which is very close the 42,9%.. One remarkable number in this table is the probability of 0,0000 for no response after the fifth call. This is because there will be no sixth call and applications will be automatically canceled if the applicant does not return: the no response rate is taken into account at the cancel rate. Based on these probabilities and the input of A_1 from last year, the workload can be simulated for each group. The total workload might differ from day to day, but the A_{1A} and A_{1B} will never exceed 6. The variety in workload is caused by days where A_{1A} and A_{1B} are less than 6; and this effects the workload of second calls, third calls, fourth calls and fifth calls.

Knowing the size of the queue, the time of handling one item is needed in order to estimate the work effort needed. This handling time is not deterministic and will have variety since it is sensitive to the employee's qualities, the complexity of the application and what kind of conversation is held in case the customer picks up. However, since the process is new, it is assumed for now these process times are deterministic. For picking and reading an application the estimated duration is 1,5 minute: $\mu_0 = 1{,}5$. This is an assumption based on experience of MT, an interview with a KCC employee, who analyzed similar activities at the calling process, and own experience and intuition. If the applicant is not reached, the waiting time and closing of the application, is approximately 1,5 minute: $\mu_2 = 1{,}5$. If the applicant is reached, the waiting time, conversation time and closing time is approximately 5,5 minutes: $\mu_1 = 5{,}5$. Thus, a non-reached applicant takes 3 minutes on average; a reached applicant takes 7 minutes on average.

Lastly, the difference between Group A and Group B should be set. The objective is to test the effect of the priority rule so other circumstances should be as equal as possible. If not, the difference in outcome can be declared by more factors instead of the objective one. As already stated, the size of A_{1A} will be the same as A_{1B}. These applications are labeled with a priority label. The division between A instances, B instances, C instances, D instances and E instances is the same for Group A and Group B in this simulation. So, if A_{1A} consists of 2 A instances, 3 B instances, 2 C instances, 1 D instances and 2 E instances, Group B will have exactly the same A_{1B}.

The model from 4.4.12, shows 21,37% of the instances is predicted to be an A instance; 27,11% B instances; 19,26% C instances; 13,73% D instances and 18,54% E instances. This division is assumed at each A_{1A} and A_{1B}. However, both Group A and Group B can only take 6 at most. Group A randomly picks 6 instances, Group B picks the 6 instances with highest priority label.

With all conditions and assumptions described above, this A/B test is simulated over a whole year with the historical data. So, Group A and Group B function simultaneously where A_1 was split every day and each group took 6 applications according randomness (Group A) and highest priority label (Group B). It

is expected, Group B would handle more A instances than Group A, and less E instances than Group B. This is confirmed by the simulation, showed in Table 20. This table represents how many of each instance were handled for each group looking over the whole simulation period of 1 year.

Table 20. *Workload per group per instance in 1 year of A/B test*

Instances in A_{1A} / A_{1B}	**Group A**	**Group B**
Total A instances	331	587
Total B instances	506	698
Total C instances	335	345
Total D instances	282	69
Total E instances	308	64

The average queue size was 17 applications. This queue consisted of the new applications, maximum of 6, and all other applications that had to be called for a second, third, fourth or fifth time. On average, this took 61 minutes per day. Mondays From this Table 20, it can be concluded both groups had a workload of 1763 applications in total, which were picked from A_1. Now, based on the model with confusion matrix as shown in Table 16, the difference in expected profitability can be calculated. First, assuming the predictions are 100% correct. The expected profitability of Group A is $331 * €2982 + 506 * €1393 + 335 * €783 + 282 * €497 + 308 * €209 = €2.161.713$ and the expected profitability of Group B is $587 * €2982 + 698 * €1393 + 345 * €783 + 69 * €497 + 64 * €209 = €3.040.552$. From the process outcome of this step and the probabilities of reachability, return and recall, it is known only 42,9% of these applicants will be eventually return their application after these recalls and only 38,2% of these applicants finally gets approved (38,2% is extracted from information in Chapter 2). So, the real expected profit from Group A, based on the contract activation is $€2.161.713 * 0{,}429 * 0{,}382 = €354.257$ and for Group B $€3.040.552 * 0{,}429 * 0{,}382 = €498.280$. However, from the evaluation of the model, it also appeared not all predictions are correct. The error percentages, shown in Table 21, should be included in the calculation as well.

Table 21. *Percentages of misclassification*

Predicted Class	**Actual Class**				
	A	**B**	**C**	**D**	**E**
A	40,1%	28,6%	14,3%	6,5%	10,5%
B	5,2%	42,3%	22,1%	10,7%	19,6%
C	1,0%	10,9%	40,1%	18,9%	29,0%
D	0,5%	6,7%	18,4%	32,9%	41,6%
E	0,0%	5,6%	13,1%	13,5%	67,7%

So, taking into account the predicted priority labels are not 100% accurate, the real instances per group will be different from Table 20. These numbers are shown in Table 22 for Group A and Table 23 for Group B.

Table 22. *Predicted instances vs. real instances for Group A*

Group A	**Actual Class**				
Predicted	**A**	**B**	**C**	**D**	**E**
A: 322	133	95	47	21	35
B: 506	26	214	112	54	99
C: 335	3	37	134	63	97
D: 282	1	19	52	93	117
E: 308	0	17	40	42	209
Total	164	382	386	274	557

Table 23. *Predicted instances vs. real instances for Group B*

Group B	**Actual Class**				
Predicted	**A**	**B**	**C**	**D**	**E**
A: 587	235	168	84	38	62
B: 698	36	296	154	75	137
C: 345	3	38	138	65	100
D: 69	0	5	13	23	29
E: 64	0	4	8	9	43
Total	276	509	397	210	371

From these two tables, the expected profitability can be recalculated for each group, where the chances of misclassification are included. The expected profitability of Group A now becomes €1.576.003 while this is €2.024.829 for Group B. Even with errors of prediction, the application of the model and priority rule improves the expected profitability with about 25%, comparing to not applying the rule.

5.1.2 Practical Implementations

The A/B test of 5.1.1 suggests the priority model will improve the profitability. However, the test involves a lot of deterministic estimations because of lack of real data; and assumes the outcome of the training set and the model is representative for any prediction in future. In reality, there might be more uncertainty in reachability, return rate and more variety in how many instances of which type do arrive. In practice, it even might be the case that certain priority labels do behave differently. For example, an A instance does return faster than other instances or E instances might have a higher reachability.

From the simulation test it appeared the total average queue (including all calls: first, second etc.) per day will consist of 17 applications. At most 6 of these applications are called for the first time (according to $A_{1A} \leq 6$ and $A_{1B} \leq 6$), others are called for second, third, fourth or fifth time. The average time it takes

to call this total queue, assuming the $\mu_0 = 1{,}5$, $\mu_1 = 5{,}5$ and $\mu_1 = 1{,}5$, is 61 minutes per day. These values of 17 applications and 61 minutes account for one group. Therefore, in order to execute this test, CompX should take into account a work effort of 2 hours a day on average. Group A and Group B can be called simultaneously, which implies 2 employees are needed for approximately 1 hour. This has preference compared to calling the groups in sequel, because the time of calling does differ in that case which also might influence results.

CompX does already send the email after 4 days, so nothing has to change on that process. The current system also makes it easy to find applications that did not respond yet after exact 7 days, so $A_1$is available under current circumstances already. Splitting this set into two, can be done randomly. However, one critical factor for the A/B test that is not available yet, is the prediction of the profitability. An application does not have a priority label yet. The priority model determines the priority label by a more complex model, a decision forest, which ideally should be integrated into the current systems. The model is really needed and cannot be replaced by 'human interpretation' which would have made it easy for an employee to assess the priority label himself. Therefore, the script of the decision forest is needed. It could be programmed into the software of Lara. The priority label can also be assigned manually, with the use of the software BigML. Each day, all applications from A_1 should be uploaded to the model which is able to determine the priority label in a few minutes. Then, the applications have a known priority label. Group A just randomly selects 6 applications, or less if $A_1 < 12$, Group B selects the best based on the priority label. Now, for Group A and Group B, a separate queue has to be registered in order to keep track on these A_{1A} and A_{1B}. There need to be records about the reachability of each call, if and when an application is returned and if and when an application is cancelled.

Due to seasonal changes it is advised to execute this A/B test for at least 6 months, but more ideally for 12 months. Evaluating the difference between the groups, can only be done after a few years because we need to know the repay behavior. The exact profitability is known if the loan is ended, which takes time. It is expected Group B should lead to higher profitability than Group A. Looking at the durations of loans, the first measure can be taken after 2 years. It is expected a sufficient share of the loans will be ended by then. On individual level, loans of Group B are expected to yield more profit, but a comparison can also be made on group level between Group A and Group B. A review of this profitability after each year is needed, as where with increasing the years more results will be available. The longer it takes, the higher the chance the loan is ended and the profit is known. If Group B indeed outperforms Group A, it is advised to execute 'call after offer' only for applications with higher priority label.

5.2 Other Suggestions for Pilots

In Section 5.1, it is suggested to implement the priority rules at one step of the process flow, which is a new process step. However, there are more steps in the process flow to which the priority rule might be applicable. Next Sections will shortly suggest other ways to implement the priority model.

5.2.1 Check by Call

Before the 'call after offer' step, the applicant is called to check the application and to send an offer. In comparison with Section 5.1, this implementation suggests to focus on the step before that. As the problem description also describes, there are periods of high demand where it becomes impossible to call all new applications in 24 hours. Or, if the applicant was not reached the first time, give all applications a second call. Therefore, it is suggested to give priority on A instances to call them in 24 hours, if time left B instances etcetera. The same accounts for the second check call. CompX could experiment with only giving A, B and C instances a second call, but not D and E. When experimenting, it is important there is a another group as well, to where the rule is not applied. Results can be analyzed after a short period (for example, one month), to see if one group might have a higher approval rate, faster processing and/or better reachability. If these effects come out significantly, these can be beneficial or unbeneficial side-effects. The real aim is to create higher revenues with the same number of loans, because there is a prediction made about this profitability. This effect can only be checked after a few years, since it depends on how the customer is going to repay the loan in reality. The majority of the loans ends somewhere within 2 and 5 years, so checking the effectiveness is also logical to take place after 2 years.

As a consequence of putting more effort on higher priority instances, it is expected to increase the profitability of all applications that are handled. To examine the effect of this adjustment in the first process step, it is advised to keep these two streams of Group A and Group B separately.

5.2.2 Risk Assessment

It is expected an A instance will have higher profit over time than lower priority instances. Losing these potential customers because of the risk assessment took too long, is relatively a big missed opportunity. The time between handing in documents and receiving feedback from CompX, takes on average about 5 days, up to 8 days at peaktimes. Therefore, an A/B test could be held at this phase as well. If A instances do have priority at one group and not at the other group, CompX could test whether these A instances do have a lower cancel rate. And if the offers accepted of these group do yield more profit over time. This performance can only be tested after a few years, since the real repayment of a loan has to reviewed, which obviously takes time.

5.2.3 Fast Lane

An A instance has a high expected profitability. Therefore, another idea is to create a fast lane for A instances over the whole line of the process. In comparison with the suggestion of Section 5.2.1, where the priority handling disappeared in later steps of the application process, this suggestion implies a priority handling at each step. At each department there should be a separate set of employees that only handle A instances. With this fast lane, the total process time should be significantly lower for A instances than for other applications that follow the 'regular' process. The expected advantages for this are, the service level of these A instances is very high which make it a happy customer, and more likely to return to CompX

when needed; and the chance these instances will go with another party, a competitor, should be lower as where the loan at CompX is provided very fast. This idea needs more calculation for the effect at the regular process flow; the extra costs that might be needed for extra employees; and most importantly, it should create a higher conversion of A instances. If not, it can be concluded it is useless to give them extra priority; or there should be a non-financial benefit as already suggested with the higher service level.

6. Conclusion

6.1 Summary of Results

The overarching objective of this research, was captured in the main research question was:

How to model the expected profitability and how can this knowledge be used at the application process?

The first main finding of this research is the definition of profitability of a loan, which consists of revenue and costs. The aspect of the costs structure at CompX could not be fully analyzed due to some overlap and intertwining with linked companies, CompZ and CompY. However, with the purpose of comparing loans, only commission costs and risk costs are found to be relevant because these do differ per loan. Commission costs relate to a third party and risks costs relate to the expected loss, i.e. provision, of a loan. Provision is money set aside which cannot be lend, while money that is lend has an objective profit of 1,317% per month. Therefore, in the formula of profit, this is interpreted as an opportunity cost. In terms of revenue, and still keeping the comparison purpose in mind, the interest rate that changes per loan has high influence on the perception of that revenue. An approximate adjustment has been created to overcome this issue with the objective to evaluate each likewise loan with a likewise interest rate, independent from the starting period. The formula for profit becomes:

$$profit = \sum_{m=1}^{ad} interest\ paid_m - 1{,}317\% * \sum_{m=1}^{ad} provision_m - \sum_{m=1}^{ad} commission_m$$

where:
$ad = the\ actual\ duration\ in\ months$
$m = the\ month\ of\ the\ payment\ where\ m = 1\ at\ the\ month\ of\ the\ first\ repayment$

Furthermore, it appeared there are many characteristics of applications and loans in which a selection had to be made. Characteristics and their notation changed over time. The 28 final selected variables remain valid over time and were available for almost all applications (details in Table 7). From the modeling analysis, it can be concluded many characteristics have little influence and were therefore excluded from almost all models. About 300 models were evaluated with different modeling methods and varying the 28 input variables. For the output of the model, three measures were chosen: one that has a continuous output of profit, one that has an output of three different profit categories and one that has an output of five different profit categories. This has led to one regression problem and two classification problems, where different modeling methodologies were applied: decision trees, decision forests, linear and logistic regressions. Looking at all models, it can be concluded the most predictive characteristics were the loan volume, the monthly payment, the interest according contract, but also the source and the prospect type came out as a factor with relative high influence. It can be concluded, a classification model has preference since the performance of regression model is weak in all cases. The use of decision forests is found to be the best technique compared to the others. Within these decision forests, a classification problem with 5 classes labeled A, B, C, D, E came out to be the best. In this classification, A represents the most profitable

class with an average profit of €2298 and E represent the less profitable class with an average profit of €209. The model needs input only from the loan volume, the interest according contract and the monthly payment. One important reason to choose this model is the fact it scores well on the benefit-cost score. This measure has been developed and evaluates on the benefit of a good classification and the cost of a misclassification in terms of monetary values.

The last step of this research covers a practical implementation of this model and a simulation of this to examine the added value of the priority labels to the current process. For MT, the implementation is preferred by adding a new step to the process, where applicants are called if they do not response in 7 days. The suggestions it to call these applicants 5 times: after 7 days, after 10 days, after 15 days, after 20 days and after 25 days. If the applicant is reached or responds between two calls, the applicant will not be called again of course.

A deterministic simplified simulation has been executed in the form of A/B testing. This implies two groups of applications are called, where one condition differs per group. The main capacity restriction for the test was the number of new instances that were involved into the process. At most 6 applications per day were called after 7 days and taken further into this process, while the total of applications per group was higher than 6 in most cases. So, a choice had to be made here about which application to call and which not. Group A picked these 6 randomly, while Group B picked these 6 based on the priority label: first picked all A instances, then B instances etc. until 6 were selected. This process is simulated with historical data of one year. Assuming the predictions of the model are accurate and never wrong, the implementation of the priority label improved the expected profitability with about 41% comparing Group B to Group A over the whole simulation year. However, the model found does misclassify applications as well, as where the priority label ended to be incorrect. Taking into account the probability of misclassification (as established with Table 21), the improvement dropped a bit: Group B's expected profitability was 25% higher than the expected profitability of Group A. This A/B test suggests the priority rule will be beneficial at the new 'Call after Offer' step and is therefore suggested to execute the A/B test in real life.

6.2 Discussion and Limitations

Though to all questions has been found an answer or at least an indication of an answer, it cannot be assumed that all results are optimal and that no other answer may be found. During the research assumptions have been made by lack of knowledge, information or time which may have consequences on the bias of the research.

One important decision for the analysis is the manner of determining the profit of a loan. Since the repayment of a loan can be complex due to the freedom of the applicant, taking the exact schedule of repay into account for each loan on the size of the used data set, has been assumed to be a too extensive process for this research. However, in the field of revenue of loans or outstanding debts, it is very common

to take into account the present value of a payment in future. This usually is referred to as the net present value (NPV), but this type of calculation is not applied to this research. Including the NPV of the future repayments instead of solely summing the interest parts of all payments, might influence the view on this term profitability. Taken into account the NPV, the actual duration of a loan will have more influence on the profitability.

One other crucial part of this research has been the establishment of the model that will predict this profitability. Existing literature has been investigated on modeling methods and multiple methods were evaluated and used. The training and modification of the models has been an extensive, time-consuming process, but it can never be assumed that there is no other model that will outperform the current one. The reason for this, is the fact that the field of predictive modeling is a research area of fast development. New methods become available rapidly. The wide variety in methods and software or tools to execute them, makes it impossible to state this model will be optimal. In addition, the difference in performance between models was in most cases very small. One conclusion can be that there simply is no clear pattern in the data; or the pattern has not been found yet by application of these models and within in the scope and knowledge of this research.

Thirdly, the search of Section 4.4 to a model has been biased by using three different types of object, i.e. profit. There is an infinitive variety in finding the number of classes and their ranges. Extending the modeling with one new profit measure increases the testing phase extensively: each new y_i requires testing of all methods with tuned parameters. Looking at this research, each y_i has been tested by the execution of about 100 models and evaluations with the objective of exploring, tuning which is a trial and error process. However, the classification problems are potentially better to model using other ranges or more or less classes. This is a time-consuming process which has been limited in the scope of this research: every new established classification problem needs extensive training and there is an infinity amount of combinations.

Lastly, the implementation part is based on many assumptions and deterministic input values. In practice, obviously, the process flow is sensitive to external factors and the input values will deviate on a daily basis. However, for the A/B test, the differences between the two groups are still likely, because all restrictions and conditions were likewise. With inclusion of real values and uncertainty, both groups will be almost likewise affected. It is important to take into account, results in practice are different from the ones found with the A/B test here, because of this uncertainty in real life.

The last point to mention in terms of limitations to this research, is the input data of loans. As discussed before, only ended loans are taken into account. This is still assumed to be a valid choice in this company's situation: CompX is a young company where the data of long running loans is too scarce to evaluate. The fictive duration showed there are not many loans that do run for that long. However, in a few years, when there is more information available of loans that do run for a long time, the view on the duration might

slightly change. Then, the conclusion might be this research overlooked the relative 'long runners' and is biased in that sense.

6.3 Recommendations

The data preparation has been the most time-consuming part of this research. The data is structured, but there still are many exceptions or multi-interpretative instances in the data base. Though the data base has been restructured lately and is more easy to access already, a recommendation is to create more unity of interpretation, especially by handling relative old data. On the side of data extraction, it is also recommended to create a manual or a dictionary to assure the knowledge and right understanding of each table, variable and instances. From experience in this research, it appeared there are only a few people who really understand everything, or a specific part. In the light of knowledge management and sustainability, it is recommended to assure the knowledge in future, more independent from specific people. A manual containing information about where information is extracted from, from which system and under what conditions it entered the system; and about what each variable stands for and definitions of abbreviations, can highly contribute to this knowledge management. In addition, a log book about changes of systems or changes in notation over time will help interpret data accurately and assure that all employees do this likewise.

One more thing to recommend based on experience with the data base, is the practical structure. The connecting key between characteristics that belong to an instance, is the contract number. This makes it hard to view from the perspective of a customer. Instead of structuring data around one loan, it would be useful to structure data around one customer. This gives deeper insight into the added value of a customer instead of a loan. However, this recommendation will be hard to implement because of the existing structure.

Based on the objective of the research, a few more recommendations can be made. First of all, the most obvious recommendation, is to implement the rule at the process of calling 7 days after the offer is sent. As the A/B test suggests, there should be an improvement in return rate by adding the step (this is tested by a third party, not this research); and by applying the priority rule, these returned offers are expected to have more revenue in future. Secondly, it is recommended to add the priority label into the system of the KCC employee (Lara). To work with the priority, it should be easily accessible. Ideally the workload would automatically be order by the priority label such that the interpretation becomes less sensitive to human interaction. Thirdly, it is advised to check the performance of the priority model yearly, to make sure it remains valid. Or, even better, and if the expertise is within the company, it is advised to check the performance and look for improvement on a yearly basis. Fourthly, small pilots could be implemented as suggested in Section 5.2, by creating for example a fast lane.

Lastly, it is suggested to create more insight into the profitability on loans in specific. CompX is a small company, with the ability to adapt to new tools and developing own tools. The current business is already

using continuous measurement of performance via Power BI, for example by keeping track on the funnel of the application process or the state of NBV. This data is visualized in a dashboard, accessible by MT and updated daily or weekly, which makes the company consciousness about what happens now and can steer more accurate and in time based on the updated information. However, there is no such knowledge about performance in terms of profit per loan or per loan group. The advice is to create such a dashboard to have better insight into the average profitability of loans over time. This can be distinguished per loan group, e.g. based on the priority label, or the characteristics that determine the priority label (loan volume, interest to be paid and monthly payment). Or a dashboard that shows what the expected profitability is of the loans that are provided in a certain period. This generates insight into how the portfolio will develop in future, based on what is provided now.

6.4 Future Work

This research is relatively new in field of predicting loans behavior. The majority in this field focuses on 'bad' payers and their probability of default, while this research focuses on 'too good' payers and their repay behavior. Besides answers to research questions, this outline also creates new questions which inspire new research.

Marketing strategy, especially online strategy, has not been the objective of this research, but is an important and costly activity of the company. The world of online advertisement is rapidly evolving and opens new opportunities. This is key in remaining competitive advantage and attracting new customers in a more saturated market. The knowledge about expected profitability might be valuable in targeting the most profitable customers as well. Marketing and advertisement strategies are not in scope of this research, but it would be interesting to investigate the added value of an expected profitability model in the marketing field.

As the process flow already showed, there is a relative low percentage that is accepted. This approval conversion rate is investigated earlier by another thesis of Tijdink (2018). Another research in future might combine the model of expected profitability with and expected probability of acceptance. The combination of these two, can improve the choices about time and effort division in more efficient way: knowing if the effort returns in a paying customer ánd knowing how much that customer is going to pay.

The last suggestion for future research, is to investigate the exclusion of loans under €10.000. As appeared from Section 4.2, loans between €5.000 and €10.000 are on average not profitable, taking into account the process costs of approximately €400. Therefore, it could be argued excluding these loans will lead to higher margins on the loans offered. However, exclusion of one group can have more consequential effects such as increasing the process costs of the other loans or missing ITO's. Future research could focus on how reasonable such an exclusion is, taking into account other effects.

References

AFM. (2018). *Leengedrag onder de loep*.

Ahmed Ali. (2016). Diminishing Returns in Performance Marketing | Optimization Up. Retrieved February 29, 2020, from https://optimizationup.com/diminishing-returns-performance-marekting/

Armstrong, G., Adam, S., Denize, S., & Kotler, P. (2015). *Principles of Marketing* (11th ed.). Melbourne: Pearson Australia. Retrieved from https://books.google.nl/books?hl=nl&lr=&id=UKyaBQAAQBAJ&oi=fnd&pg=PP1&dq=law+of+diminishing+returns+marketing+&ots=RXvdPOT5iA&sig=7zHwRVFXIVRb0XIvUkra1goftho#v=snippet&q=diminishing returns&f=false

Castelvecchi, D. (2016). The Black Box. *Nature News*, *538*(7623), 20–23. Retrieved from http://www.nature.com/news/can-we-open-the-black-box-of-ai-1.20731

Chernyak, O., & Pavlenko, T. (2010). Performance optimization of object comparison. *International Journal of Intelligent Systems*, *25*(4), 326–344. https://doi.org/10.1002/int

Criminisi, A., Shotton, J., & Konukoglu, E. (2011). Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Computer Graphics and Vision*, *7*(2–3), 81–227. https://doi.org/10.1561/0600000035

Dietterichtt, T. G. (2000). Multiple Classiefier Systems. In R. Roli & J. Kittler (Eds.), *First International*. Cagliari: Springer. Retrieved from https://link.springer.com/content/pdf/10.1007%2F3-540-45014-9.pdf

Dukić, D., Dukić, G., & Kvesić, L. (2011). A credit scoring decision support system. *Proceedings of the International Conference on Information Technology Interfaces, ITI*, 391–396.

Grefen, P. (2016). 4.2.7 Patterns for other aspects. In *Information System Architecture*. Eindhoven University of Technology.

Gupta, B., Uttarakhand, P., Rawat, I. A., Arora, A., & Dhami, N. (2017). *Analysis of Various Decision Tree Algorithms for Classification in Data Mining. International Journal of Computer Applications* (Vol. 163). Retrieved from https://pdfs.semanticscholar.org/fd39/e1fa85e5b3fd2b0d000230f6f8bc9dc694ae.pdf

Gurney, K. (1997). Neural networks - An Overview. In *An Introduction to Neural Networks* (pp. 1–18). London: UCL Press.

Hand, D. J., & Henley, W. E. (1997). Statistical classification methods in consumer credit scoring: A review. *Journal of the Royal Statistical Society. Series A: Statistics in Society*, *160*(3), 523–541. https://doi.org/10.1111/j.1467-985X.1997.00078.x

Hlawatsch, S., & Ostrowski, S. (2010). Economic Loan Loss Provision and Expected Loss. *Business Research*, *3*(2), 133–149. https://doi.org/10.1007/BF03342719

Kim, J., & Kim, K. (2007). Loss Given Default Modelling under the Asymptotic Single Risk Factor

Assumption. *Asia-Pacific Journal of Financial Studies*, *36*(2), 223–236.

Leong, C. K. (2016). Credit Risk Scoring with Bayesian Network Models. *Computational Economics*, *47*(3), 423–446. https://doi.org/10.1007/s10614-015-9505-8

Mehta, M., Rissanen, J., & Agrawal, R. (1995). MDL-based Decision Tree Pruning. In *KDD-95 Proceedings* (pp. 216–221).

Mester, L. J. (1997). What's the Point of Credit Scoring ? *Business Review*, *Sep*, 3–16. Retrieved from https://www.phil.frb.org/research-and-data/publications/business-review/1997/september-october/brso97lm.pdf

Mike Vladimer. (2018). This '20/80 Rule of Big Data' has huge implications for IoT tech. Retrieved April 1, 2020, from https://medium.com/internet-of-things-from-osv/this-20-80-rule-of-big-data-has-huge-implications-for-iot-tech-e8e7cdf42387

Mitroff, I. I., & Sagasti, F. R. (1973). Operations research from the viewpoint of general systems theory. *Omega*, *1*(6), 695–709. https://doi.org/10.1016/0305-0483(73)90087-X

Montgomery, D. C., & Runger, G. C. (2011). *Applied Statistic and Probability for Engineers* (5th ed.). Hoboken: Wiley.

Myles, A. J., Feudale, R. N., Liu, Y., Woody, N. A., & Brown, S. D. (2004). An introduction to decision tree modeling. *Journal of Chemometrics*, *18*(6), 275–285. https://doi.org/10.1002/cem.873

Parker, C. (2011). An Analysis of Performance Measures For Binary Classifiers. In *IEEE International Conference on Data Mining*. https://doi.org/10.1109/ICDM.2011.21

Sanchez-Barrios, L. J., Andreeva, G., & Ansell, J. (2016). Time-To-Profit Scorecards for Revolving Credit. *European Journal of Operational Research*, *249*(2), 397–406. https://doi.org/10.1016/j.ejor.2015.09.052

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2000). Analysis of recommendation algorithms for e-commerce. In *Proceedings of the 2nd ACM conference on Electronic commerce - EC '00* (pp. 158–167). New York, New York, USA: ACM Press. https://doi.org/10.1145/352871.352887

Sengupta, R., & Bhardwaj, G. (2015). Credit Scoring and Loan Default. *International Review of Finance*, *15*(2), 139–167. https://doi.org/10.1111/irfi.12048

Starkweather, J., & Moske, A. K. (2011). Multinomial Logistic Regression. https://doi.org/10.1097/00006199-200211000-00009

Tijdink, F. (2018). *Loan application classification and service time optimization at a financial service provider*. TU Eindhoven.

Tu, J. V. (1996). Advantages and Disadvantages of Using Artificial Neural Networks versus Logistic Regression for Predicting Medical Outcomes. *Journal of Clinical Epidemiology*, *49*(11), 1225–1231. Retrieved from https://www.jclinepi.com/article/S0895-4356(96)00002-9/pdf

West, D. (2000). Neural network credit scoring models. *Computers and Operations Research*, *27*(11–

12), 1131–1152. https://doi.org/10.1016/S0305-0548(99)00149-5

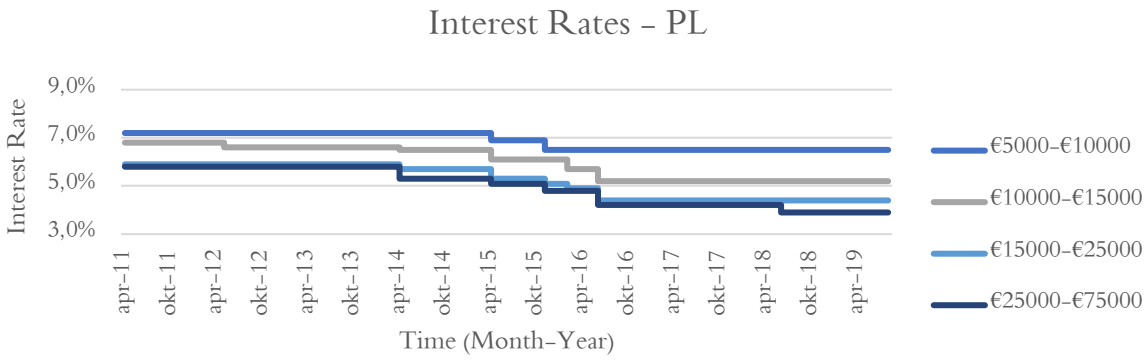
Wouters, M., Selto, F. H., Hilton, R. W., & Maher, M. W. (2012). *Cost Management: Strategies for Bussiness Decisions* (1st ed.). New York: McGraw-Hill.

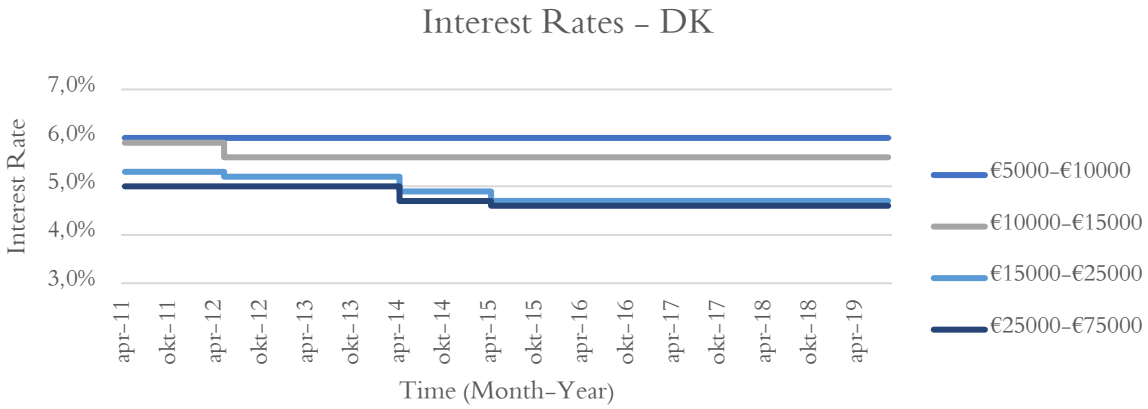
Xaio, J. J. (2016). Consumer Finance Capability and Wellbeing. In J. J. Xaio (Ed.), *Handbook of Consumer Finance Research* (Second, pp. 3–17). Springer. https://doi.org/10.1007/978-3-319-28887-1_2

Appendices

Appendix A

The interest rates during 2011 and 2019.





Appendix B

Appendix B1

Loan 1	3,9%	0,3193%	120	Loan 2	5,8%	0,4709%	120	
	25000	€ 251,12			25000	€ 273,19		
Month	Net Debt	Interest Paid	New Net Debt		Net Debt	Interest Paid	New Net Debt	DIFFERENCE(%)
1	25000	€ 79,83	€ 24.828,71		25000	€ 117,74	€ 24.844,54	0,6781
2	€ 24.828,71	€ 79,29	€ 24.656,88		€ 24.844,54	€ 117,00	€ 24.688,35	0,6776
3	€ 24.656,88	€ 78,74	€ 24.484,49		€ 24.688,35	€ 116,27	€ 24.531,43	0,6772
4	€ 24.484,49	€ 78,19	€ 24.311,56		€ 24.531,43	€ 115,53	€ 24.373,76	0,6768
5	€ 24.311,56	€ 77,63	€ 24.138,07		€ 24.373,76	€ 114,79	€ 24.215,35	0,6763
6	€ 24.138,07	€ 77,08	€ 23.964,03		€ 24.215,35	€ 114,04	€ 24.056,20	0,6759
7	€ 23.964,03	€ 76,52	€ 23.789,43		€ 24.056,20	€ 113,29	€ 23.896,30	0,6755
8	€ 23.789,43	€ 75,97	€ 23.614,28		€ 23.896,30	€ 112,54	€ 23.735,64	0,6750
9	€ 23.614,28	€ 75,41	€ 23.438,57		€ 23.735,64	€ 111,78	€ 23.574,23	0,6746
10	€ 23.438,57	€ 74,85	€ 23.262,29		€ 23.574,23	€ 111,02	€ 23.412,05	0,6742
11	€ 23.262,29	€ 74,28	€ 23.085,45		€ 23.412,05	€ 110,26	€ 23.249,12	0,6737
12	€ 23.085,45	€ 73,72	€ 22.908,05		€ 23.249,12	€ 109,49	€ 23.085,41	0,6733
13	€ 22.908,05	€ 73,15	€ 22.730,08		€ 23.085,41	€ 108,72	€ 22.920,94	0,6729
14	€ 22.730,08	€ 72,58	€ 22.551,55		€ 22.920,94	€ 107,94	€ 22.755,69	0,6724
15	€ 22.551,55	€ 72,01	€ 22.372,44		€ 22.755,69	€ 107,17	€ 22.589,66	0,6720
16	€ 22.372,44	€ 71,44	€ 22.192,76		€ 22.589,66	€ 106,38	€ 22.422,85	0,6715
17	€ 22.192,76	€ 70,87	€ 22.012,51		€ 22.422,85	€ 105,60	€ 22.255,26	0,6711
18	€ 22.012,51	€ 70,29	€ 21.831,68		€ 22.255,26	€ 104,81	€ 22.086,87	0,6707
19	€ 21.831,68	€ 69,72	€ 21.650,27		€ 22.086,87	€ 104,02	€ 21.917,69	0,6702
20	€ 21.650,27	€ 69,14	€ 21.468,29		€ 21.917,69	€ 103,22	€ 21.747,72	0,6698
21	€ 21.468,29	€ 68,55	€ 21.285,72		€ 21.747,72	€ 102,42	€ 21.576,95	0,6694
22	€ 21.285,72	€ 67,97	€ 21.102,57		€ 21.576,95	€ 101,61	€ 21.405,37	0,6689
23	€ 21.102,57	€ 67,39	€ 20.918,84		€ 21.405,37	€ 100,81	€ 21.232,98	0,6685
24	€ 20.918,84	€ 66,80	€ 20.734,52		€ 21.232,98	€ 99,99	€ 21.059,78	0,6680
25	€ 20.734,52	€ 66,21	€ 20.549,61		€ 21.059,78	€ 99,18	€ 20.885,77	0,6676
26	€ 20.549,61	€ 65,62	€ 20.364,11		€ 20.885,77	€ 98,36	€ 20.710,93	0,6672
27	€ 20.364,11	€ 65,03	€ 20.178,01		€ 20.710,93	€ 97,54	€ 20.535,27	0,6667
28	€ 20.178,01	€ 64,43	€ 19.991,33		€ 20.535,27	€ 96,71	€ 20.358,79	0,6663
29	€ 19.991,33	€ 63,84	€ 19.804,05		€ 20.358,79	€ 95,88	€ 20.181,47	0,6658
30	€ 19.804,05	€ 63,24	€ 19.616,16		€ 20.181,47	€ 95,04	€ 20.003,32	0,6654
31	€ 19.616,16	€ 62,64	€ 19.427,68		€ 20.003,32	€ 94,20	€ 19.824,33	0,6649
32	€ 19.427,68	€ 62,04	€ 19.238,60		€ 19.824,33	€ 93,36	€ 19.644,50	0,6645
33	€ 19.238,60	€ 61,43	€ 19.048,92		€ 19.644,50	€ 92,51	€ 19.463,82	0,6641
34	€ 19.048,92	€ 60,83	€ 18.858,62		€ 19.463,82	€ 91,66	€ 19.282,29	0,6636
35	€ 18.858,62	€ 60,22	€ 18.667,72		€ 19.282,29	€ 90,81	€ 19.099,91	0,6632
36	€ 18.667,72	€ 59,61	€ 18.476,21		€ 19.099,91	€ 89,95	€ 18.916,66	0,6627
37	€ 18.476,21	€ 59,00	€ 18.284,09		€ 18.916,66	€ 89,09	€ 18.732,55	0,6623
38	€ 18.284,09	€ 58,39	€ 18.091,36		€ 18.732,55	€ 88,22	€ 18.547,58	0,6618
39	€ 18.091,36	€ 57,77	€ 17.898,01		€ 18.547,58	€ 87,35	€ 18.361,73	0,6614
40	€ 17.898,01	€ 57,15	€ 17.704,04		€ 18.361,73	€ 86,47	€ 18.175,01	0,6609
41	€ 17.704,04	€ 56,53	€ 17.509,45		€ 18.175,01	€ 85,59	€ 17.987,41	0,6605
42	€ 17.509,45	€ 55,91	€ 17.314,25		€ 17.987,41	€ 84,71	€ 17.798,93	0,6601
43	€ 17.314,25	€ 55,29	€ 17.118,41		€ 17.798,93	€ 83,82	€ 17.609,56	0,6596
44	€ 17.118,41	€ 54,66	€ 16.921,96		€ 17.609,56	€ 82,93	€ 17.419,30	0,6592
45	€ 16.921,96	€ 54,04	€ 16.724,87		€ 17.419,30	€ 82,03	€ 17.228,14	0,6587
46	€ 16.724,87	€ 53,41	€ 16.527,16		€ 17.228,14	€ 81,13	€ 17.036,08	0,6583
47	€ 16.527,16	€ 52,78	€ 16.328,82		€ 17.036,08	€ 80,23	€ 16.843,11	0,6578
48	€ 16.328,82	€ 52,14	€ 16.129,84		€ 16.843,11	€ 79,32	€ 16.649,24	0,6574

49	€ 16.129,84	€ 51,51	€ 15.930,22		€ 16.649,24	€ 78,41	€ 16.454,45	0,6569
50	€ 15.930,22	€ 50,87	€ 15.729,97		€ 16.454,45	€ 77,49	€ 16.258,75	0,6565
51	€ 15.729,97	€ 50,23	€ 15.529,08		€ 16.258,75	€ 76,57	€ 16.062,13	0,6560
52	€ 15.529,08	€ 49,59	€ 15.327,55		€ 16.062,13	€ 75,64	€ 15.864,58	0,6556
53	€ 15.327,55	€ 48,95	€ 15.125,37		€ 15.864,58	€ 74,71	€ 15.666,10	0,6551
54	€ 15.125,37	€ 48,30	€ 14.922,55		€ 15.666,10	€ 73,78	€ 15.466,68	0,6547
55	€ 14.922,55	€ 47,65	€ 14.719,08		€ 15.466,68	€ 72,84	€ 15.266,33	0,6542
56	€ 14.719,08	€ 47,00	€ 14.514,97		€ 15.266,33	€ 71,90	€ 15.065,03	0,6538
57	€ 14.514,97	€ 46,35	€ 14.310,20		€ 15.065,03	€ 70,95	€ 14.862,78	0,6533
58	€ 14.310,20	€ 45,70	€ 14.104,77		€ 14.862,78	€ 70,00	€ 14.659,58	0,6529
59	€ 14.104,77	€ 45,04	€ 13.898,69		€ 14.659,58	€ 69,04	€ 14.455,43	0,6524
60	€ 13.898,69	€ 44,38	€ 13.691,95		€ 14.455,43	€ 68,08	€ 14.250,31	0,6520
61	€ 13.691,95	€ 43,72	€ 13.484,55		€ 14.250,31	€ 67,11	€ 14.044,23	0,6515
62	€ 13.484,55	€ 43,06	€ 13.276,49		€ 14.044,23	€ 66,14	€ 13.837,17	0,6510
63	€ 13.276,49	€ 42,40	€ 13.067,77		€ 13.837,17	€ 65,16	€ 13.629,14	0,6506
64	€ 13.067,77	€ 41,73	€ 12.858,38		€ 13.629,14	€ 64,19	€ 13.420,14	0,6501
65	€ 12.858,38	€ 41,06	€ 12.648,32		€ 13.420,14	€ 63,20	€ 13.210,14	0,6497
66	€ 12.648,32	€ 40,39	€ 12.437,58		€ 13.210,14	€ 62,21	€ 12.999,16	0,6492
67	€ 12.437,58	€ 39,72	€ 12.226,18		€ 12.999,16	€ 61,22	€ 12.787,19	0,6488
68	€ 12.226,18	€ 39,04	€ 12.014,10		€ 12.787,19	€ 60,22	€ 12.574,21	0,6483
69	€ 12.014,10	€ 38,36	€ 11.801,34		€ 12.574,21	€ 59,22	€ 12.360,24	0,6479
70	€ 11.801,34	€ 37,69	€ 11.587,91		€ 12.360,24	€ 58,21	€ 12.145,25	0,6474
71	€ 11.587,91	€ 37,00	€ 11.373,79		€ 12.145,25	€ 57,20	€ 11.929,25	0,6470
72	€ 11.373,79	€ 36,32	€ 11.158,99		€ 11.929,25	€ 56,18	€ 11.712,24	0,6465
73	€ 11.158,99	€ 35,63	€ 10.943,50		€ 11.712,24	€ 55,16	€ 11.494,20	0,6460
74	€ 10.943,50	€ 34,95	€ 10.727,33		€ 11.494,20	€ 54,13	€ 11.275,14	0,6456
75	€ 10.727,33	€ 34,26	€ 10.510,46		€ 11.275,14	€ 53,10	€ 11.055,05	0,6451
76	€ 10.510,46	€ 33,56	€ 10.292,90		€ 11.055,05	€ 52,06	€ 10.833,92	0,6447
77	€ 10.292,90	€ 32,87	€ 10.074,65		€ 10.833,92	€ 51,02	€ 10.611,74	0,6442
78	€ 10.074,65	€ 32,17	€ 9.855,70		€ 10.611,74	€ 49,98	€ 10.388,53	0,6438
79	€ 9.855,70	€ 31,47	€ 9.636,05		€ 10.388,53	€ 48,92	€ 10.164,26	0,6433
80	€ 9.636,05	€ 30,77	€ 9.415,70		€ 10.164,26	€ 47,87	€ 9.938,93	0,6428
81	€ 9.415,70	€ 30,07	€ 9.194,65		€ 9.938,93	€ 46,81	€ 9.712,54	0,6424
82	€ 9.194,65	€ 29,36	€ 8.972,89		€ 9.712,54	€ 45,74	€ 9.485,09	0,6419
83	€ 8.972,89	€ 28,65	€ 8.750,42		€ 9.485,09	€ 44,67	€ 9.256,56	0,6415
84	€ 8.750,42	€ 27,94	€ 8.527,24		€ 9.256,56	€ 43,59	€ 9.026,96	0,6410
85	€ 8.527,24	€ 27,23	€ 8.303,35		€ 9.026,96	€ 42,51	€ 8.796,28	0,6405
86	€ 8.303,35	€ 26,52	€ 8.078,74		€ 8.796,28	€ 41,43	€ 8.564,51	0,6401
87	€ 8.078,74	€ 25,80	€ 7.853,42		€ 8.564,51	€ 40,33	€ 8.331,65	0,6396
88	€ 7.853,42	€ 25,08	€ 7.627,38		€ 8.331,65	€ 39,24	€ 8.097,70	0,6391
89	€ 7.627,38	€ 24,36	€ 7.400,61		€ 8.097,70	€ 38,14	€ 7.862,64	0,6387
90	€ 7.400,61	€ 23,63	€ 7.173,12		€ 7.862,64	€ 37,03	€ 7.626,47	0,6382
91	€ 7.173,12	€ 22,91	€ 6.944,91		€ 7.626,47	€ 35,92	€ 7.389,20	0,6378
92	€ 6.944,91	€ 22,18	€ 6.715,97		€ 7.389,20	€ 34,80	€ 7.150,80	0,6373
93	€ 6.715,97	€ 21,45	€ 6.486,29		€ 7.150,80	€ 33,68	€ 6.911,28	0,6368
94	€ 6.486,29	€ 20,71	€ 6.255,88		€ 6.911,28	€ 32,55	€ 6.670,64	0,6364
95	€ 6.255,88	€ 19,98	€ 6.024,74		€ 6.670,64	€ 31,41	€ 6.428,86	0,6359
96	€ 6.024,74	€ 19,24	€ 5.792,85		€ 6.428,86	€ 30,28	€ 6.185,94	0,6354
97	€ 5.792,85	€ 18,50	€ 5.560,23		€ 6.185,94	€ 29,13	€ 5.941,88	0,6350
98	€ 5.560,23	€ 17,76	€ 5.326,87		€ 5.941,88	€ 27,98	€ 5.696,67	0,6345

99	€ 5.326,87	€ 17,01	€ 5.092,76		€ 5.696,67	€ 26,83	€ 5.450,30	0,6341
100	€ 5.092,76	€ 16,26	€ 4.857,90		€ 5.450,30	€ 25,67	€ 5.202,78	0,6336
101	€ 4.857,90	€ 15,51	€ 4.622,29		€ 5.202,78	€ 24,50	€ 4.954,08	0,6331
102	€ 4.622,29	€ 14,76	€ 4.385,93		€ 4.954,08	€ 23,33	€ 4.704,22	0,6327
103	€ 4.385,93	€ 14,01	€ 4.148,81		€ 4.704,22	€ 22,15	€ 4.453,18	0,6322
104	€ 4.148,81	€ 13,25	€ 3.910,94		€ 4.453,18	€ 20,97	€ 4.200,96	0,6317
105	€ 3.910,94	€ 12,49	€ 3.672,31		€ 4.200,96	€ 19,78	€ 3.947,55	0,6313
106	€ 3.672,31	€ 11,73	€ 3.432,91		€ 3.947,55	€ 18,59	€ 3.692,95	0,6308
107	€ 3.432,91	€ 10,96	€ 3.192,75		€ 3.692,95	€ 17,39	€ 3.437,15	0,6303
108	€ 3.192,75	€ 10,20	€ 2.951,83		€ 3.437,15	€ 16,19	€ 3.180,14	0,6299
109	€ 2.951,83	€ 9,43	€ 2.710,13		€ 3.180,14	€ 14,98	€ 2.921,92	0,6294
110	€ 2.710,13	€ 8,65	€ 2.467,67		€ 2.921,92	€ 13,76	€ 2.662,49	0,6289
111	€ 2.467,67	€ 7,88	€ 2.224,42		€ 2.662,49	€ 12,54	€ 2.401,83	0,6285
112	€ 2.224,42	€ 7,10	€ 1.980,41		€ 2.401,83	€ 11,31	€ 2.139,95	0,6280
113	€ 1.980,41	€ 6,32	€ 1.735,61		€ 2.139,95	€ 10,08	€ 1.876,84	0,6275
114	€ 1.735,61	€ 5,54	€ 1.490,03		€ 1.876,84	€ 8,84	€ 1.612,48	0,6270
115	€ 1.490,03	€ 4,76	€ 1.243,67		€ 1.612,48	€ 7,59	€ 1.346,88	0,6266
116	€ 1.243,67	€ 3,97	€ 996,52		€ 1.346,88	€ 6,34	€ 1.080,03	0,6261
117	€ 996,52	€ 3,18	€ 748,58		€ 1.080,03	€ 5,09	€ 811,92	0,6256
118	€ 748,58	€ 2,39	€ 499,85		€ 811,92	€ 3,82	€ 542,55	0,6252
119	€ 499,85	€ 1,60	€ 250,32		€ 542,55	€ 2,56	€ 271,91	0,6247
120	€ 250,32	€ 0,80	€ 0,00		€ 271,91	€ 1,28	€ 0,00	0,6242
		€ 5.134,55				€ 7.783,25		

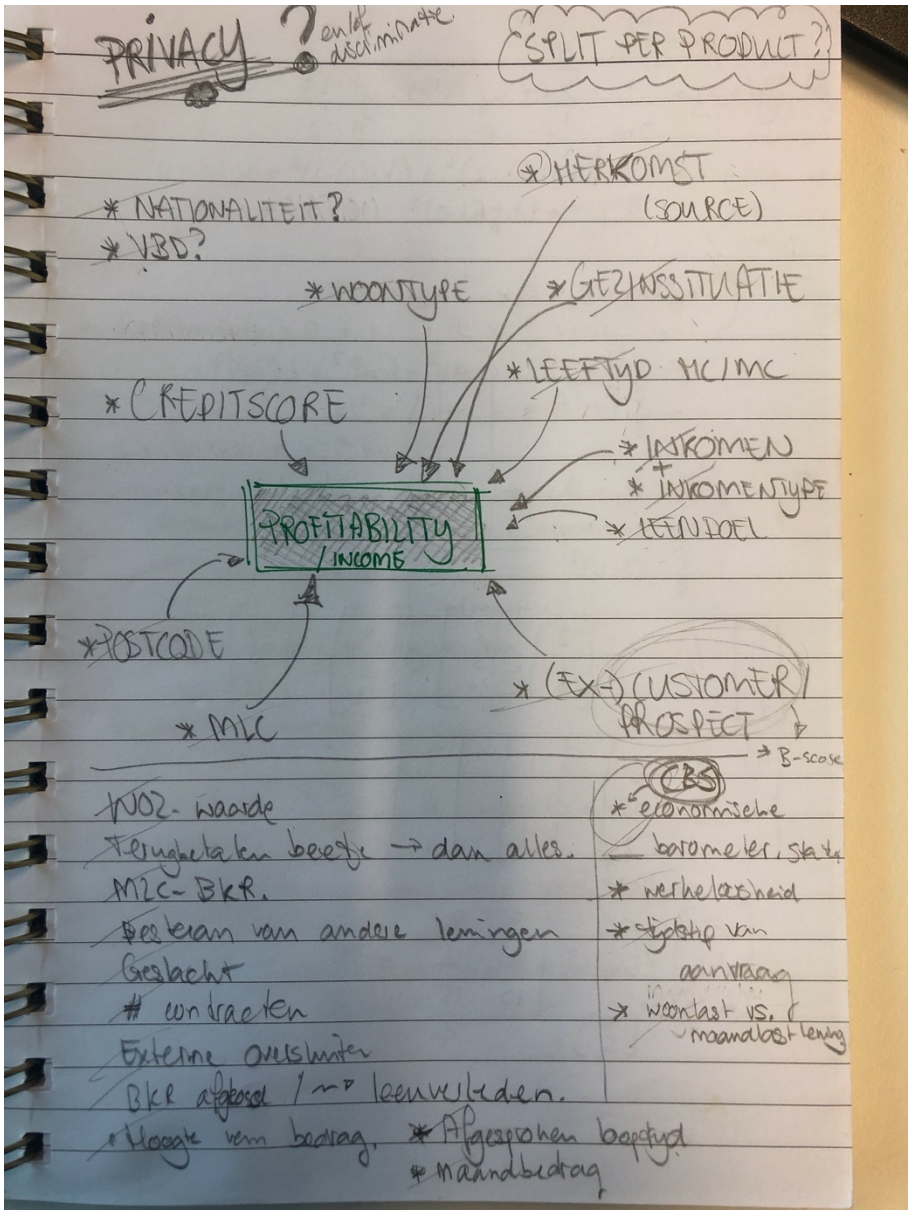
Average difference over first 36 months	0,6704
Average difference over first 60 months	0,6649
Average difference over first 120 months	0,6515

Appendix B2

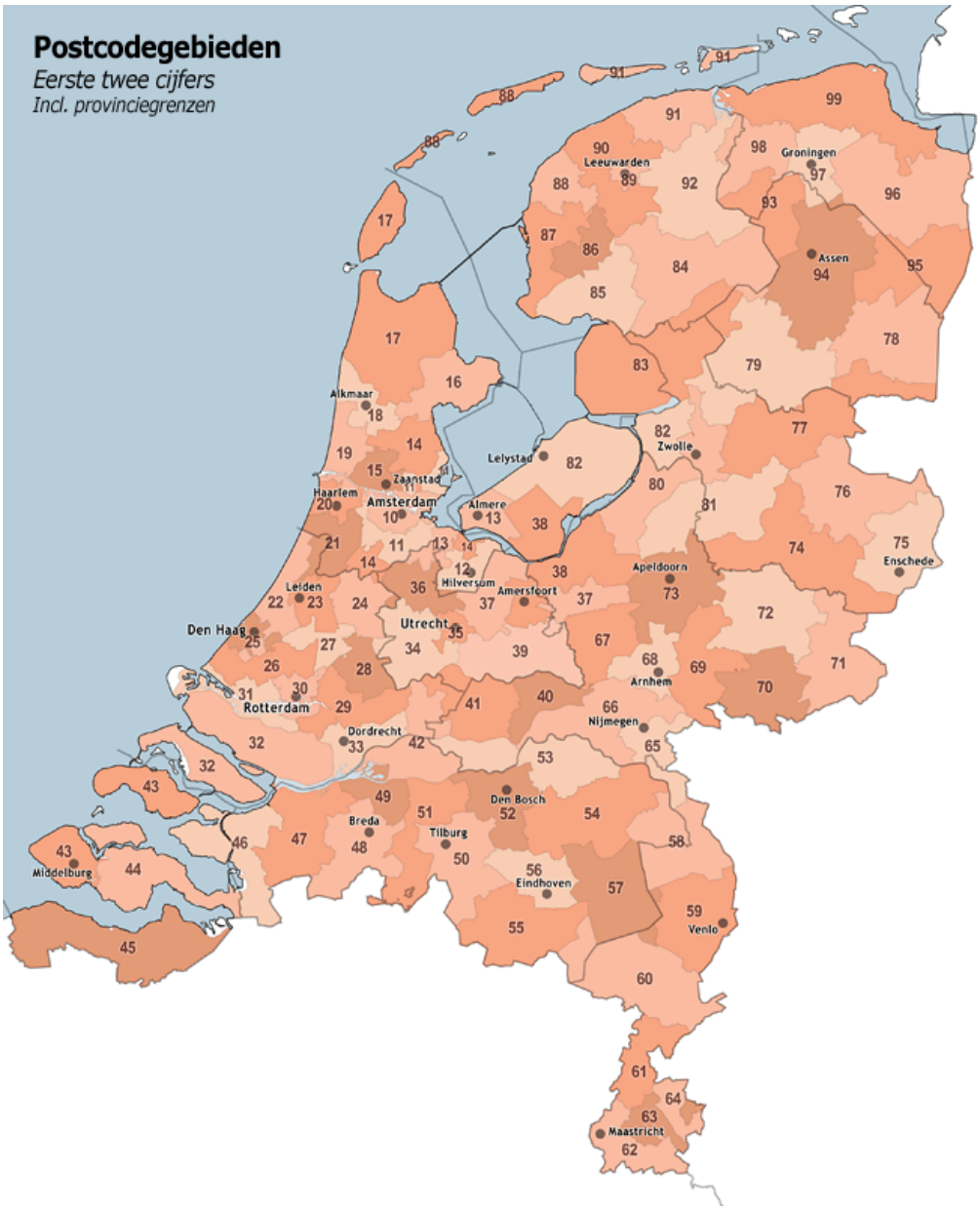
LoanVolume | Old Interest Rate	Duration	Old' Interest	Correction Factor	Corrected Interest	Current Interest	Accuracy
75000 | 5,8%	120	23349,75	0,6515	15212,36	15403,66	98,76%
	80	20383,29	0,6515	13279,71	13516,21	98,25%
	66	18089,85	0,6515	11785,54	12031,77	**97,95%**
	40	12308,29	0,6651	8186,24	8246,68	99,27%
	6	2086,08	0,6704	1398,51	1412,27	99,03%
65000 | 5,8%	120	20236,54	0,6515	13184,11	13349,84	98,76%
	72	16593,7	0,6515	10810,80	11023,28	98,07%
	61	14843,52	0,6515	9670,55	9887,67	**97,80%**
	18	5211,33	0,6704	3493,68	3514,84	99,40%
30000 |5,3%	120	8493,07	0,7187	6103,97	6161,46	99,07%
	72	6976,91	0,7187	5014,31	5087,67	98,56%
	37	4208,85	0,7299	3072,04	3090,7	99,40%
	6	763,86	0,7342	560,83	564,91	99,28%
75000 | 5,3%	61	15613,66	0,7187	11221,54	11408,86	98,36%
75000 | 5,1%	61	15009,94	0,7494	11248,45	11408,86	98,59%
75000 | 4,8%	61	14106,19	0,8002	11287,77	11408,86	98,94%
75000 | 4,2%	61	12305,59	0,9238	11367,90	11408,86	99,64%
24000 | 5,9%	120	7608,11	0,7276	5535,66	5589,88	99,03%
	61	5577,77	0,7276	4058,39	4129,52	98,28%
	60	5512,13	0,7395	4076,22	4081,76	99,86%
	37	3751,12	0,7395	2773,95	2791,73	99,36%
	6	678,81	0,7441	505,10	508,96	99,24%
24000 | 5,3%	61	4996,37	0,8178	4086,03	4129,52	98,95%
24000 | 4,9%	61	4610,3	0,8904	4105,01	4129,52	99,41%
14000 | 6,8%	120	5158,94	0,7455	3845,99	3844,8	100,03%
	61	3765,24	0,7455	2806,99	2858,18	98,21%
	6	454,91	0,7631	347,14	349,87	99,22%
9500 | 7,2%	120	3720,38	0,8931	3322,67	3336,86	99,57%
	84	3323,72	0,8931	2968,41	2985,75	99,42%
	61	2709,95	0,8931	2420,26	2439,03	99,23%
	60	2677,6	0,8996	2408,77	2410,14	99,94%
	36	1772,11	0,9021	1598,62	1598,82	99,99%

Appendix C

Appendix C1



Appendix C2

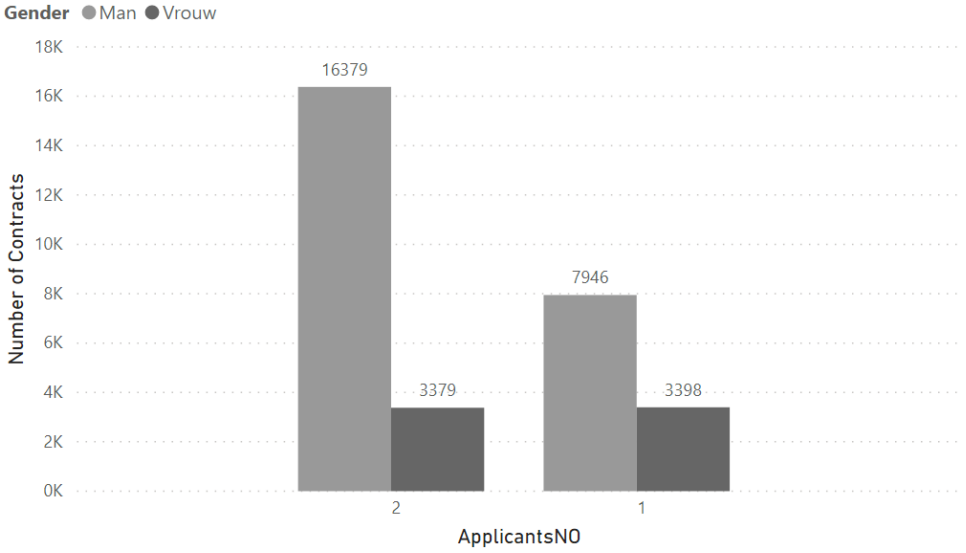


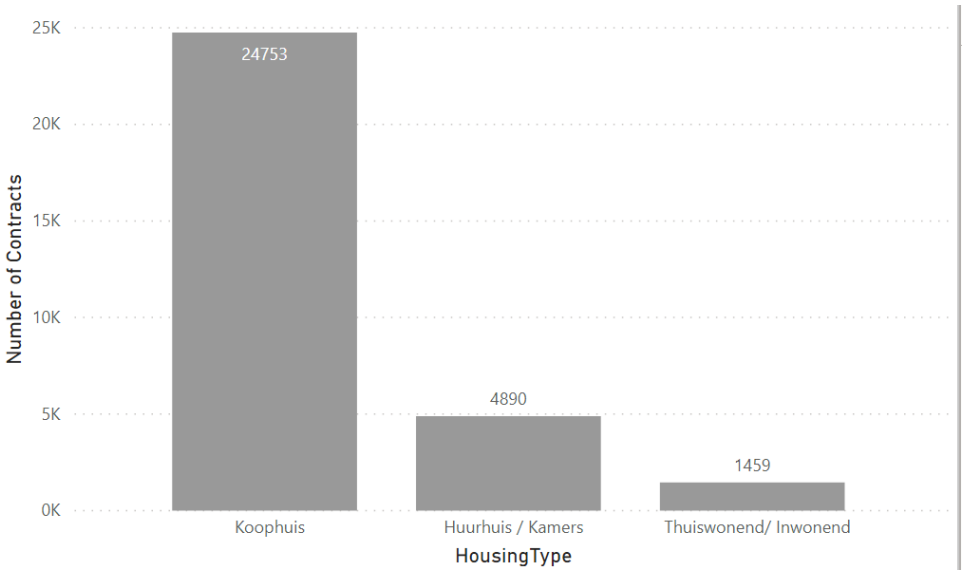
https://nl.wikipedia.org/wiki/Postcodes_in_Nederland#/media/Bestand:2010-PC2-Prov-650px.png

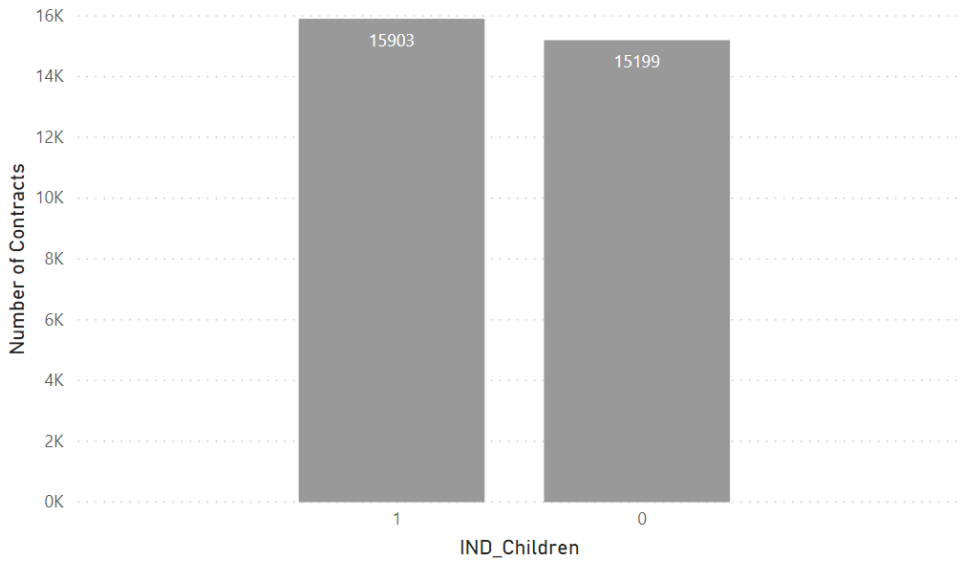
Appendix C3

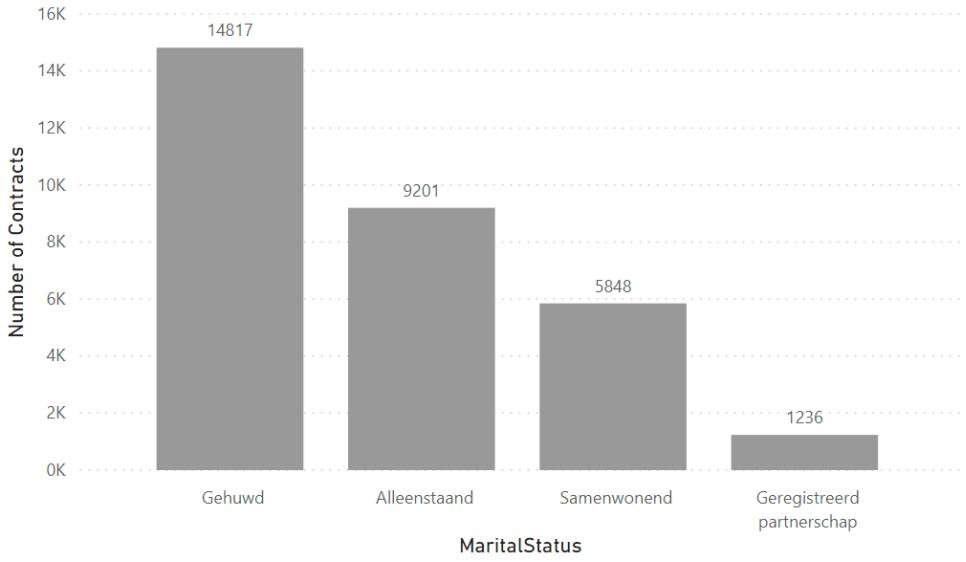
Applicant 1 | Applicant 2	**Permanent (P)**	**Temporary (T)**	**Retirement (R)**	**Government Benefit (GB)**	**Self-employed (ZZP))**	**No income: Tax Credit (TC)**
P	1	6	7	8	9	10
T		2	11	12	13	14
R			3	15	16	17
ZZP				4	18	19
TC					5	20
GB						

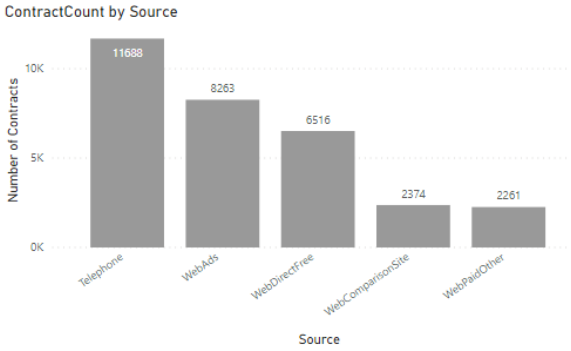
Appendix D

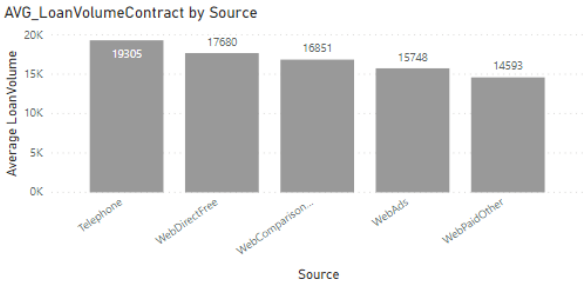

Gender Man Vrouw
18K
16K
14K
12K
10K
8K
6K
4K
2K
0K
Number of Contracts
16379
3379
7946
3398
2
1
ApplicantsNO

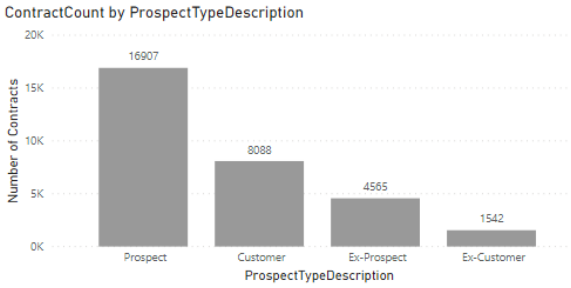

25K
20K
15K
10K
5K
0K
Number of Contracts
24753
4890
1459
Koophuis
Huurhuis / Kamers
Thuiswonend/ Inwonend
HousingType

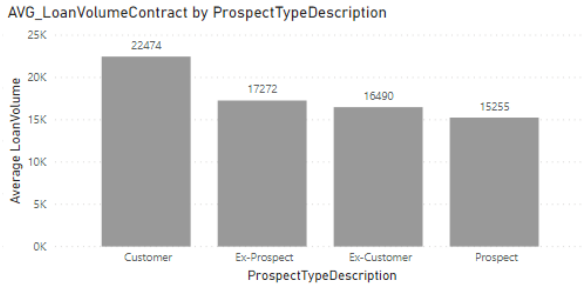

16K
14K
12K
10K
8K
6K
4K
2K
0K
Number of Contracts
15903
15199
1
0
IND_Children

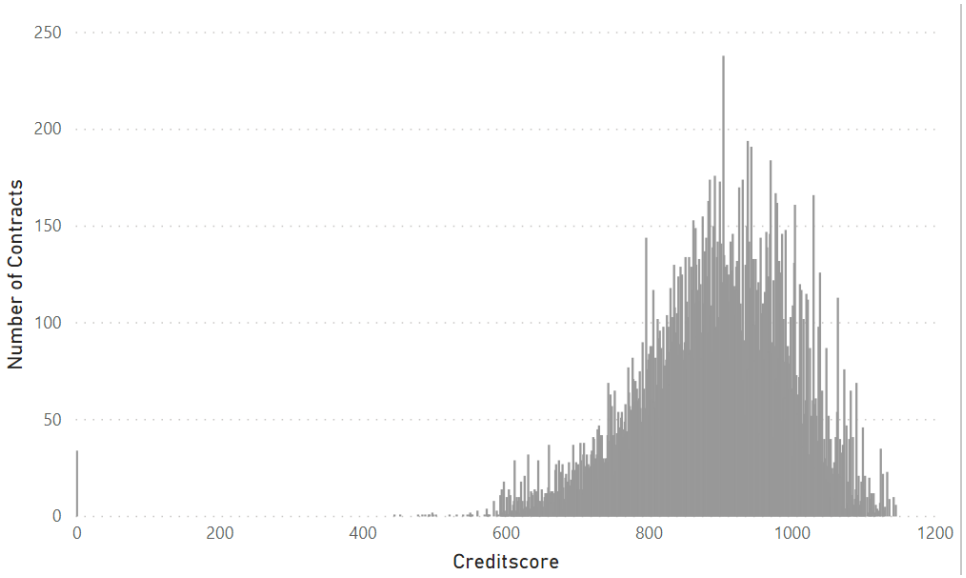

16K
14K
12K
10K
8K
6K
4K
2K
0K
Number of Contracts
14817
9201
5848
1236
Gehuwd
Alleenstaand
Samenwonend
Geregistreerd partnerschap
MaritalStatus

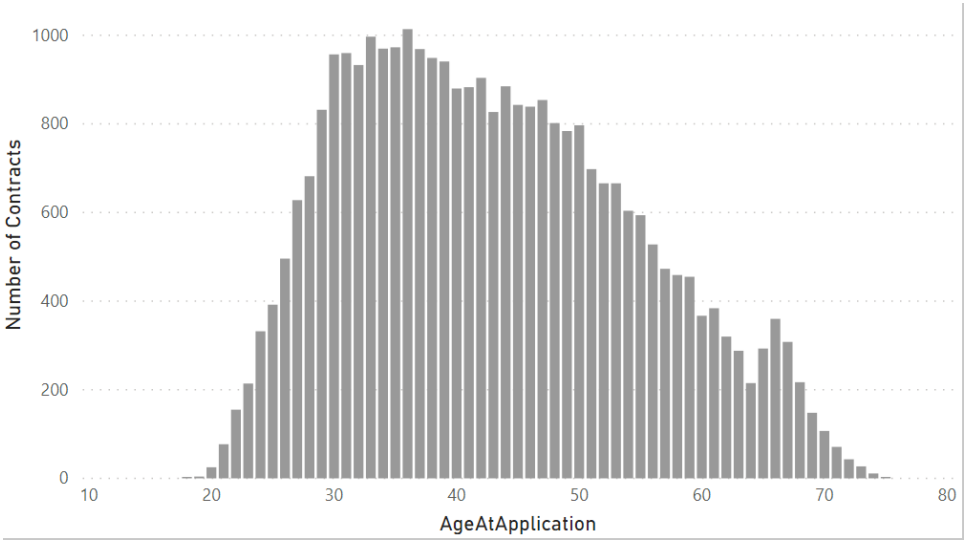

ContractCount by Source
11688
8263
6516
2374
2261
Telephone
WebAds
WebDirectFree
WebComparisonSite
WebPaidOther
Source


AVG_LoanVolumeContract by Source
19305
17680
16851
15748
14593
Telephone
WebDirectFree
WebComparison...
WebAds
WebPaidOther
Source

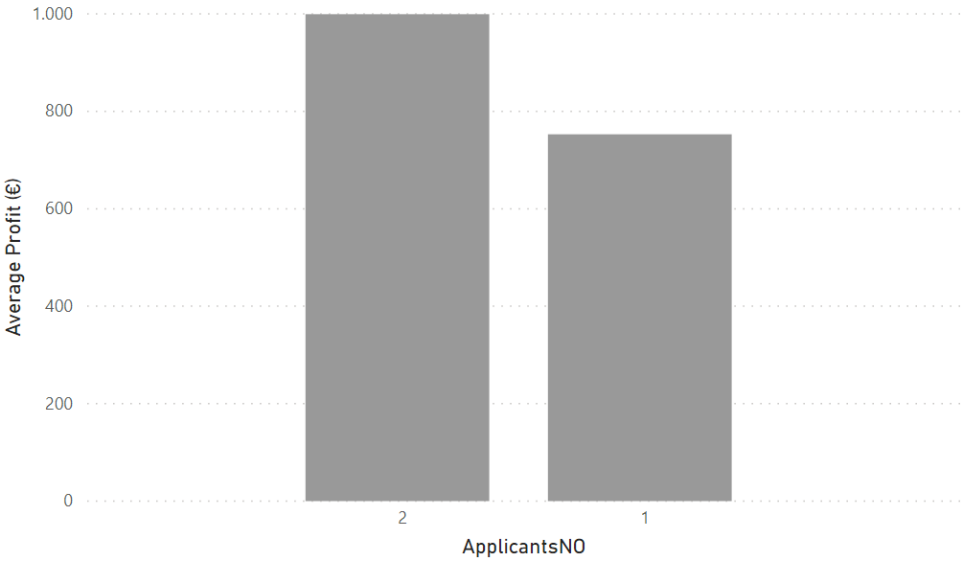

ContractCount by ProspectTypeDescription
16907
8088
4565
1542
Prospect
Customer
Ex-Prospect
Ex-Customer
ProspectTypeDescription

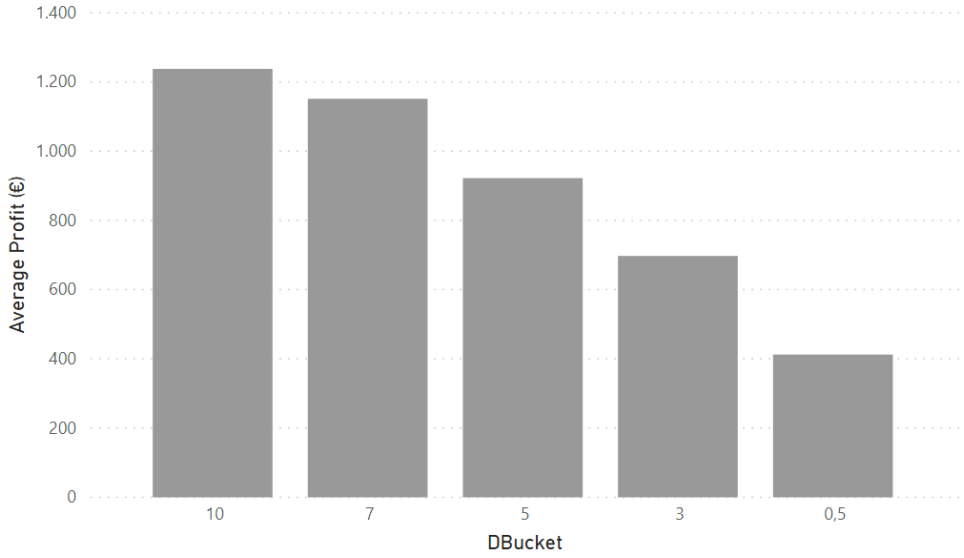

AVG_LoanVolumeContract by ProspectTypeDescription
22474
17272
16490
15255
Customer
Ex-Prospect
Ex-Customer
Prospect
ProspectTypeDescription


250
200
150
100
50
0
Number of Contracts
0
200
400
600
800
1000
1200
Creditscore


1000
800
600
400
200
0
Number of Contracts
10
20
30
40
50
60
70
80
AgeAtApplication

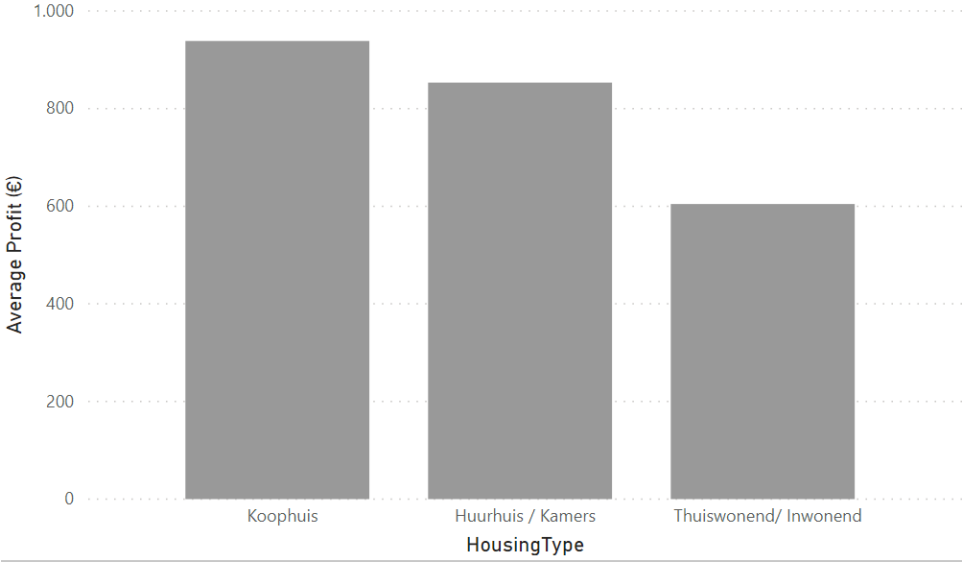
Appendix E


1.000
800
600
400
200
0
Average Profit (€)
2
1
ApplicantsNO

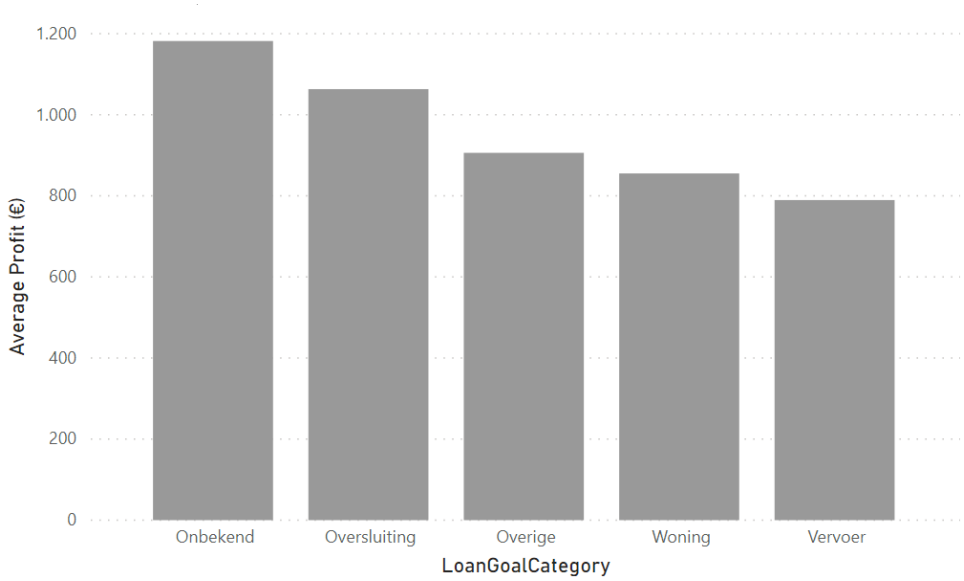

1.400
1.200
1.000
800
600
400
200
0
Average Profit (€)
10
7
5
3
0,5
DBucket

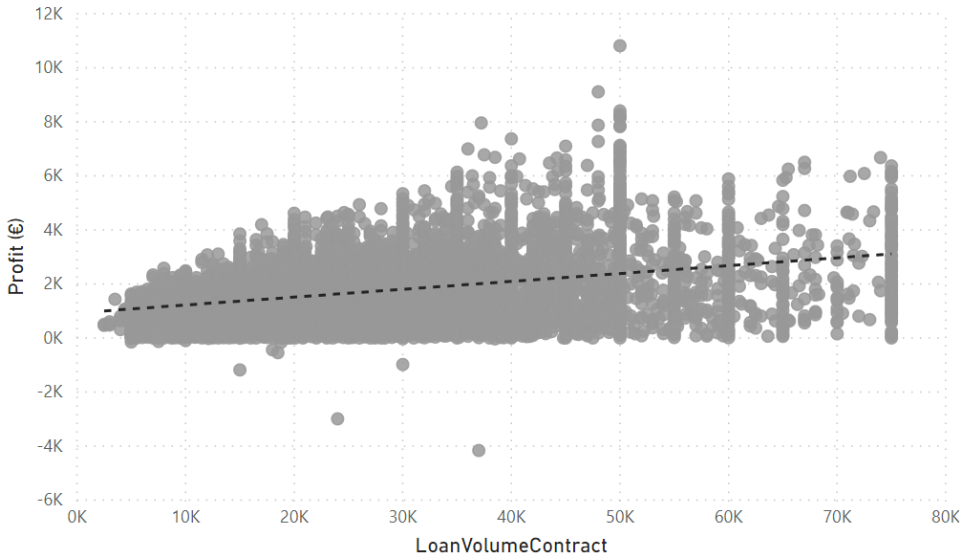

12K
10K
8K
6K
4K
2K
0K
-2K
-4K
-6K
Profit (€)
400
500
600
700
800
900
1000
1100
1200
Creditscore

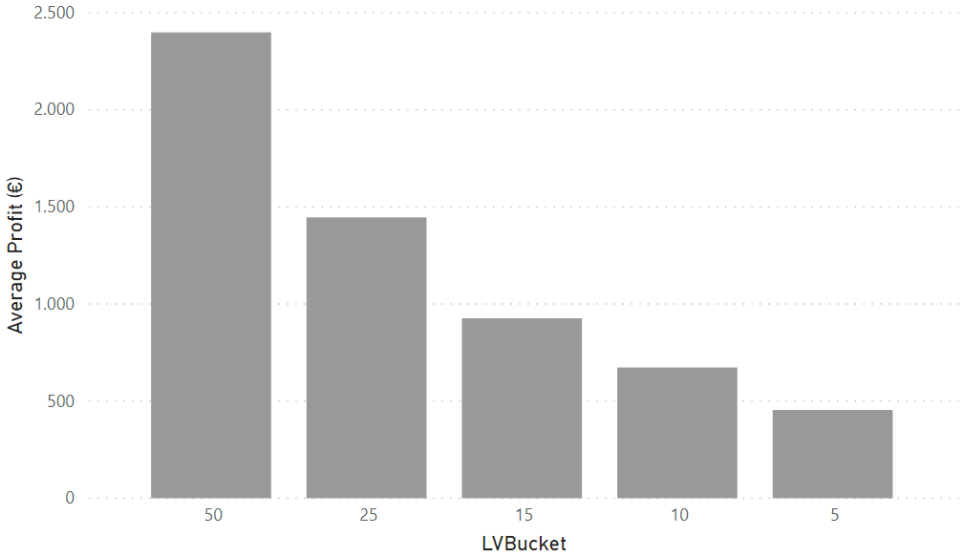

1.000
800
600
400
200
0
Average Profit (€)
Man
Vrouw
Gender

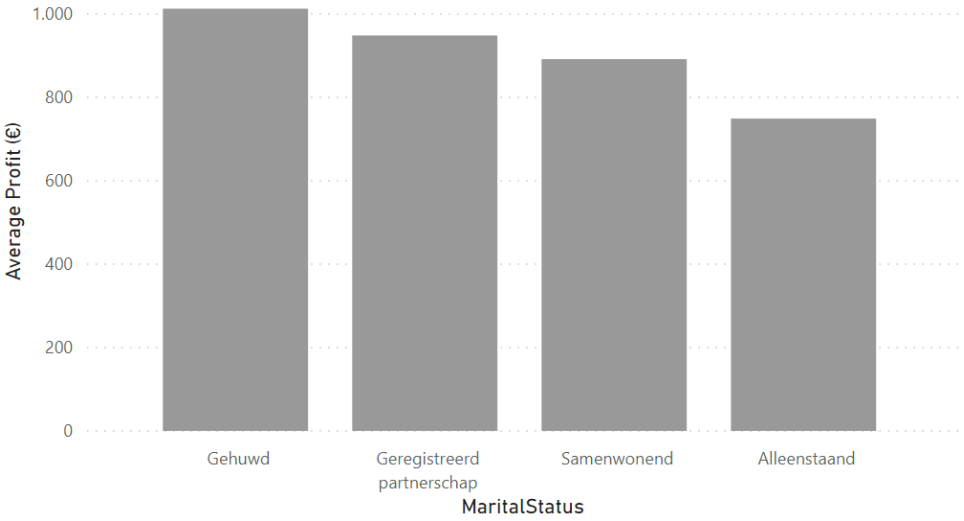

1.000
800
600
400
200
0
Average Profit (€)
Koophuis
Huurhuis / Kamers
Thuiswonend/ Inwonend
HousingType

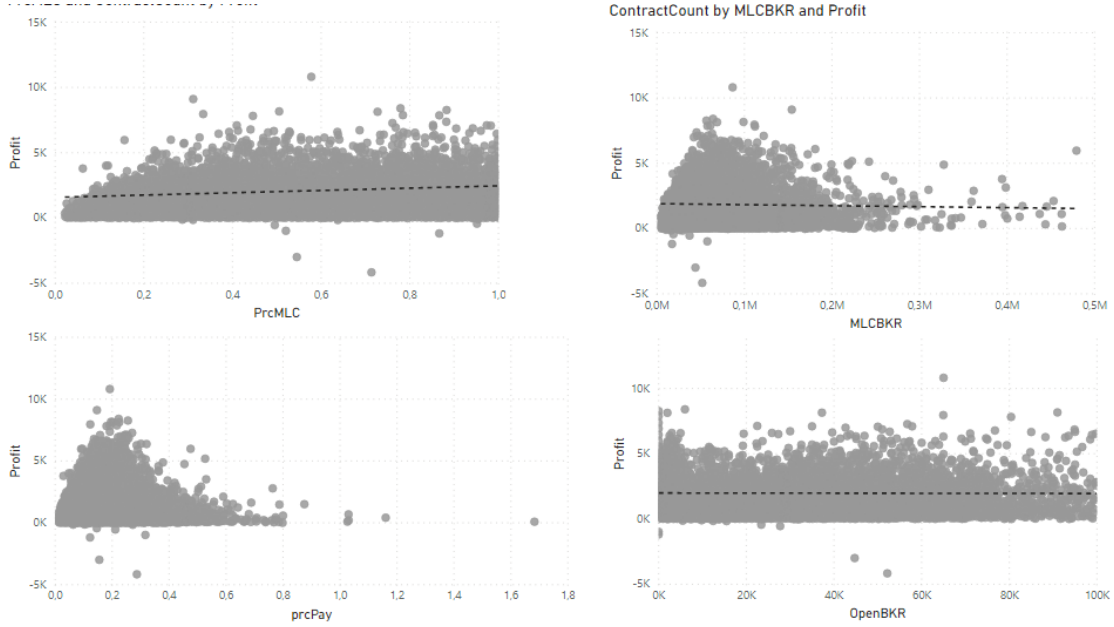

1.600
1.400
1.200
1.000
800
600
400
200
0
Average Profit (€)
18 20 9 8 5 1 7 13 6 4 15 19 12 10 0 16 3 14 17 11 2
IncomeType

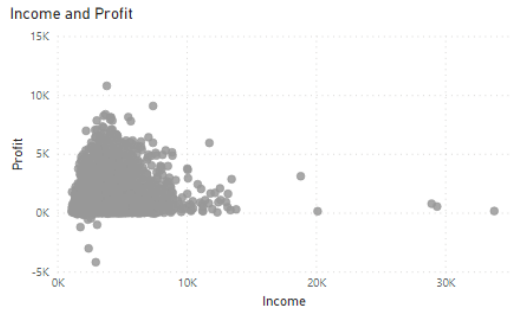

1.200
1.000
800
600
400
200
0
Average Profit (€)
Onbekend
Oversluiting
Overige
Woning
Vervoer
LoanGoalCategory

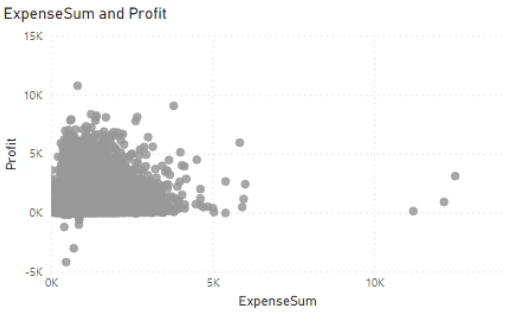

12K
10K
8K
6K
4K
2K
0K
-2K
-4K
-6K
Profit (€)
0K
10K
20K
30K
40K
50K
60K
70K
80K
LoanVolumeContract

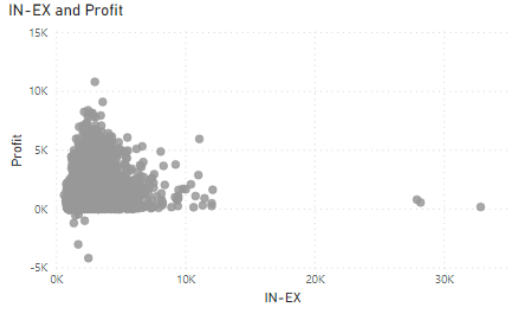

2.500
2.000
1.500
1.000
500
0
Average Profit (€)
50
25
15
10
5
LVBucket

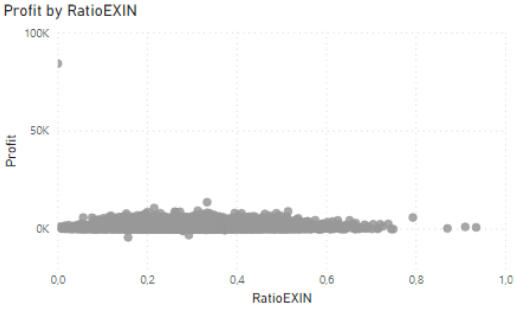

1.000
800
600
400
200
0
Average Profit (€)
Gehuwd
Geregistreerd partnerschap
Samenwonend
Alleenstaand
MaritalStatus

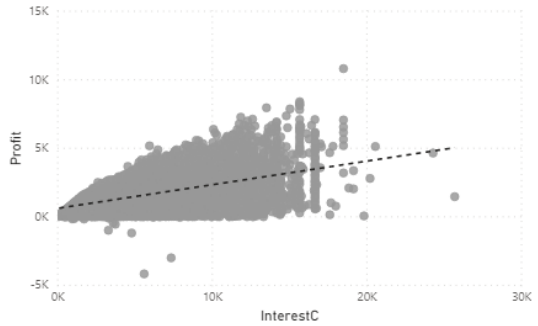

Profit
PrcMLC
ContractCount by MLCBKR and Profit
MLCBKR
prcPay
OpenBKR

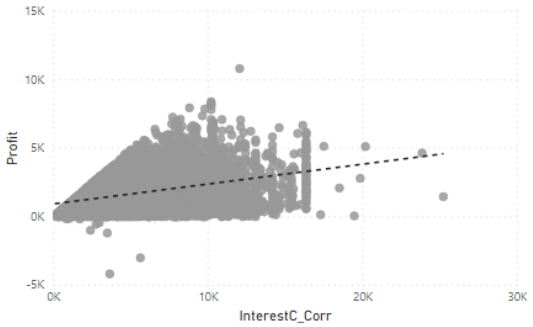

Income and Profit
Profit
Income


ExpenseSum and Profit
Profit
ExpenseSum

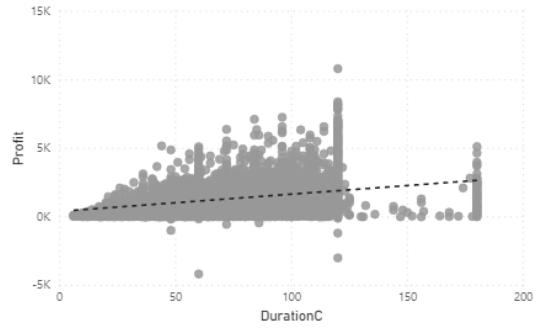

IN-EX and Profit
Profit
IN-EX


Profit by RatioEXIN
Profit
RatioEXIN

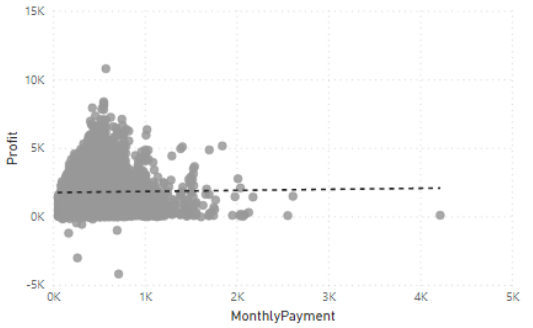


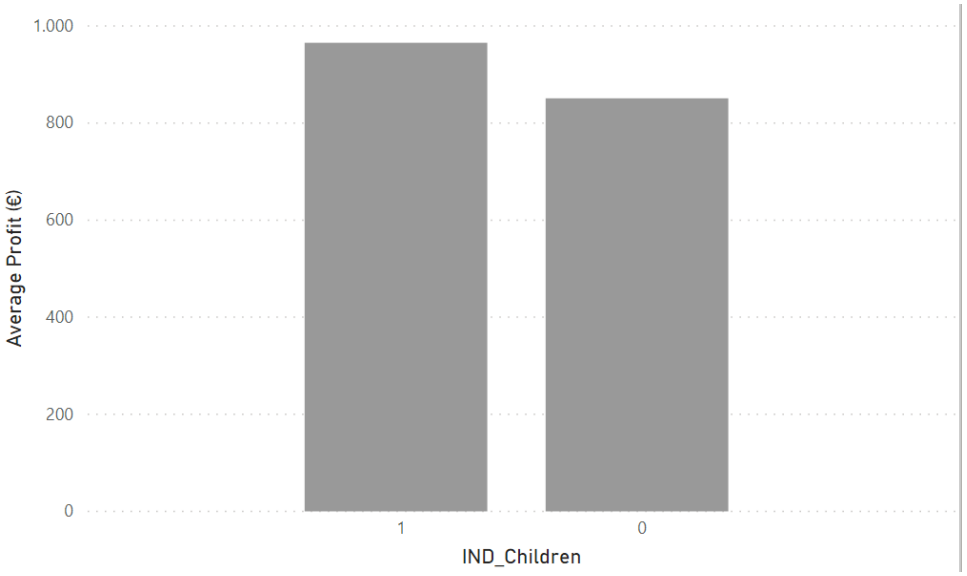


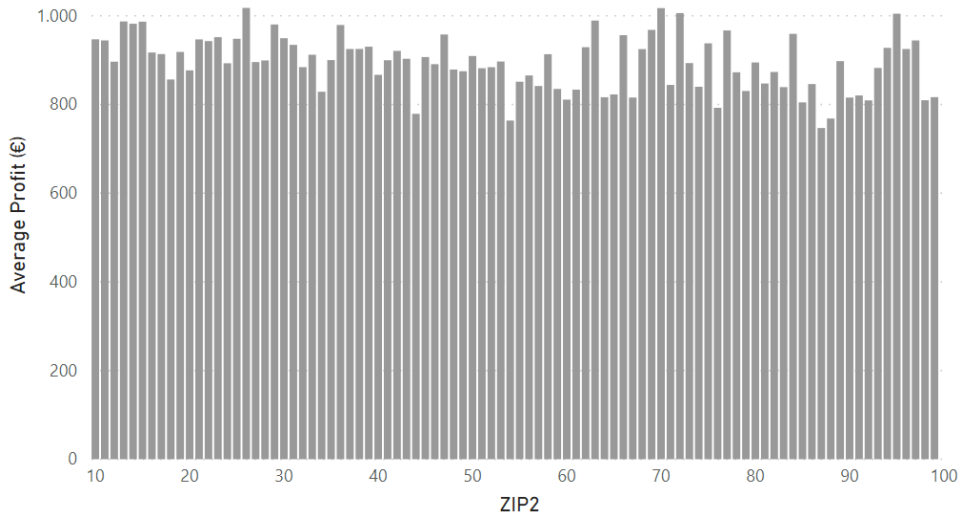
DurationC and ContractCount by Profit



MonthlyPayment and ContractCount by Profit







Appendix F

Appendix F1

Explanation of Decision Tree

A decision tree is originally founded to assist decision making by structuring the problem. It is build on three basic input to the model: multiple outcomes, consequences and probabilities something occurs [1]. The decision tree is also a widely-used concept in the field of healthcare decisions. One simple example is given with Figure F1.1. The first question is referred to as the *root note*. The possible outcomes at the bottom of the tree is referred to as the *end note*. An end note represents an outcome with a probability this outcome is sure. Each split determines a smaller decision space. The number of splits is referred to as the depth of the tree. The output can be either a class predicting a discrete value, but a decision tree can also be applied to regression problems (regression tree) where it predicts a real number [2].

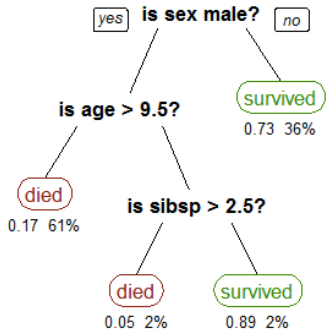


Figure F1.1 An example of a simple decision tree [2]

A decision tree as above is relatively simple, consequences can easily be thought of by intuition and experience and probabilities can be found by applying statistics to historic data or can also be chosen based on experience and intuition. The decision tree is a concept that exists over decades, but found a new application in machine learning (ML). Having more complex problems, with more consequences and outcomes, it can be impossible to determine probabilities and to select a consequence that should follow after a node. In the field of ML, multiple algorithms are developed that iterate over all possible nodes (consequences), determine the probability and decide which consequence will follow next. Simply said, it splits the set in two parts on one attribute, splits each of these two in two parts again etc. Most distinctive variables will be in the top of the three and more specific distinctions are made at the bottom of the top. The algorithms that exist will not be extensively discussed here, but the one applied in this research is the CART algorithm. CART stands for Classification and Regression Trees and is founded by Leo Breiman [3]. The method uses binary tree to divide the forecast space into subsets, which implies the node cannot be split into more than 2 consequences [4]. The algorithm is inductive and partitions the measurement space which are the decision trees produced by each split [5]. Each partition The Gini Index is used to determine the attribute that should follow: the Gini Index should be minimum after splitting [4].

However, a risk in decision trees can be excessive growth. Having many attributes and outcomes, the tree can become too large where interpretation will be impossible. One method to overcome is cross-validation. A node threshold sets a maximum to the number of nodes, i.e. splits. For example, a node threshold of 4, can only have 4 nodes in the tree which implies the depth will be at most 2, while the algorithm actually is able to specify more and create more nodes. This concept is also referred to as overfitting, with the risk is too specific to the trained data and is less likely to adapt and predict new instances. Another technique to prevent overfitting is pruning. Creating a tree, the algorithm first tries to classify all training attributes as specific as possible. The second step is to decrease the tree with pruning. Pruning examines statistics reliability for each split [6]. Pruning examines the grow in confidence of the model at each split, it uses pessimistic error estimates. BigML uses smart pruning where only pruning is executed at nodes that contain less than 1% of the data. Statistical pruning implies each nodes is considered to delete because of statistical reliability.

References of this overview

[1] Clemen R. (1996), *Making Hard Decisions: An Introduction to decision analysis* (2nd Edition). Duxbur. Chapter 3, 4 and 12

[2]https://en.wikipedia.org/wiki/Decision_tree_learning

[3] BigML Team (2019), Classification and Regression with BigML Dashboard, Version 2.1 https://static.bigml.com/pdf/BigML_Classification_and_Regression.pdf?ver=173eeff

[4] Hssina, B., Merbouha, A., Ezzikouri, H., & Erritali, M. (n.d.). *A comparative study of decision tree ID3 and C4.5. IJACSA) International Journal of Advanced Computer Science and Applications, Special Issue on Advances in Vehicular Ad Hoc Networking and Applications* (Vol. 13). Retrieved from www.ijacsa.thesai.org

[5] Crawford, S. L. (1989). Extensions to the CART algorithm. *International Journal of Man-Machine Studies*. https://doi.org/10.1016/0020-7373(89)90027-8

[6] Mingers, J. (1989). *An Empirical Comparison of Pruning Methods for Decision Tree Induction* (Vol. 4). Retrieved from https://link.springer.com/content/pdf/10.1023/A:1022604100933.pdf

Appendix F2

Explanation of Decision Forest

A diversion from single decision trees are decision forests. There are different types of ensembles, but the basic principle is the algorithm is to find a best model based on combining multiple decision trees. The algorithms applied in this thesis are the one bagging or bootstrap aggregating (referred to as decision forests); and random decision forests [1]. The bagging strategy builds single models from a random subset of the dataset. It aims to improve the accuracy and stability of the model. It produces samples according to the bootstrap strategy. The models are fitted and combined by averaging the voting (or the output in case of regression problems) [2]. Random decision forests do work almost the same except it selects a new subset at each split. So, it combines DTs not only at the end of the production but also in the decision tree itself. It constructs trees in subspaces that are randomly [4]. At each split, the algorithm selects a new subset

and seeks for a new split that optimizes the objective function [5]. By voting of all decision trees per path, the final class is chosen. This is related to concept of 'wisdom of crowds'. The DTs are relatively uncorrelated, but because of the combination the prediction works well. Combining the results of each tree lowers the errors, and exclude the individual errors of trees [6].

References of this overview

[1] BigML Team (2019), Classification and Regression with BigML Dashboard, Version 2.1 https://static.bigml.com/pdf/BigML_Classification_and_Regression.pdf?ver=173eeff

[2] https://en.wikipedia.org/wiki/Bootstrap_aggregating

[3] https://en.wikipedia.org/wiki/Random_forest

[4] Tin Kam Ho, "Random decision forests," Proceedings of 3rd International Conference on Document Analysis and Recognition, Montreal, Quebec, Canada, 1995, pp. 278–282 vol.1.

[5] Criminisi, A., Shotton, J., & Konukoglu, E. (2011). Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Computer Graphics and Vision*, *7*(2–3), 81–227. https://doi.org/10.1561/0600000035

[6] Yiu, T. (2019). Understanding Random Forest - How the Algorithm Works and Why it Is So Effective. Retrieved April 5, 2020, from https://towardsdatascience.com/understanding-random-forest-58381e0602d2

Appendix G

Call 1			Call 2			Call 3			Call 4			Call 5		
Unreached		84,25%	Unreached		83,50%	Unreached		84,50%	Unreached		86,00%	Unreached		86,00%
Reached		15,75%	Reached		16,50%	Reached		15,50%	Reached		14,00%	Reached		14,00%
Unreached: No reaction	0,8	67,40%	Unreached: No reaction	0,85	70,98%	Unreached: No reaction	0,9	76,05%	Unreached: No reaction	0,9	77,40%	Unreached: No reaction		0,00%
Unreached: calls, cancel	0,04	3,37%	Unreached: calls, cancel	0,09	7,52%	Unreached: calls, cancel	0,0833333	7,04%	Unreached: calls, cancel	0,09	7,74%	Unreached: calls, cancel	0,95	81,70%
Unreached: calls, returns	0,16	13,48%	Unreached: calls, returns	0,06	5,01%	Unreached: calls, returns	0,0166667	1,41%	Unreached: calls, returns	0,01	0,86%	Unreached: calls, returns	0,05	4,30%
Reached: wants to return	0,75	11,81%	Reached: wants to return	0,6	9,90%	Reached: wants to return	0,5	7,75%	Reached: wants to return	0,3	4,20%	Reached: wants to return	0,05	0,70%
Reached: wants to cancel	0,25	3,94%	Reached: wants to cancel	0,4	6,60%	Reached: wants to cancel	0,5	7,75%	Reached: wants to cancel	0,7	9,80%	Reached: wants to cancel	0,95	13,30%