Computing Science

Thesis presented for the degree of Master of Science

Analysis of Unemployment Data and Intervention Instruments

Author:
Ruben Peters
ICA-3590232

Supervisor:
Dr. A.J. Feelders

Department of Information and Computing Sciences,
Faculty of Science,
Utrecht University
August 28, 2016

Universiteit Utrecht

Abstract

It is important to analyse unemployment data, so intervention instruments can be improved and the unemployed can exit social security faster. To this end this paper presents a number of data analysis techniques that can be used to estimate the probability of outflow and the effectiveness of intervention instruments.

Firstly classification trees and survival analysis are used to give insight into the influence of features on the chance of outflow. Furthermore the classification trees can also be used to predict the chance of outflow for new clients. We found some features that seem to greatly influence the chance of outflow.

Secondly novel matching algorithms are used to create a treatment and comparable control group for four intervention instruments. Survival analysis is then used to analyse the differences between the groups. We conclude that two of the four intervention instruments are significantly better than other intervention instruments.

Acknowledgements

Firstly I would like to thank the people of Ynformed for giving me an interesting problem for my thesis and for supporting me both during the analysis phase and the writing phase and especially Martijn Minderhoud en Tim Paauw for their support and help during the project.

Secondly I also like to thank my supervisor dr. Ad Feelders for his enthusiasm and help with finding relevant and interesting literature, as well as his ability to respond to any e-mail within a day.

Finally I would also like to thank the municipality where this project was performed for their data and help during the first phase of this research.

Contents

Chapter 1

Introduction

Reintegration from unemployment is an interesting problem. According to Eurostat[1] the unemployment rate in the Netherlands in April 2015 was 7 percent. Financially supporting this group can be costly, therefore the government offers training programs and other ways of support to help people find a job. Nevertheless these trainings are also costly. According to the State Budget of 2016[2], the costs of unemployment expenditure will be 12 billion euros, whereas reintegration and support costs 7 billion euros a year (to put this into perspective, the Dutch defence budget is 7,5 billion euros).

It is important to monitor the effectiveness of reintegration, although this might prove to be difficult. Traditional methods rely on surveys or controlled experiments. This approach however ignores the vast amount of data that is already present in different databases. This project focuses on using machine learning techniques on various datasets to give insight into the data and effectiveness of intervention instruments. This insight can then be used to identify possible difficulties and improving the intervention instruments. This research was performed on the data of a municipality in the Netherlands in collaboration with Ynformed[3].

In the remainder of this chapter the research questions and the study design will be discussed and in Section 1.3 a summary of the reintegration system in the Netherlands will be given. In Chapter 2 the used datasets will be discussed and the preprocessing on the datasets will be explained. In Chapter 3 the theoretical background will be discussed by referring to related literature. In Chapter 4 the used algorithms and methods will be discussed and in Chapter 5 the experiments and the result from those experiments will be given. Finally Chapter 6 will conclude the results of this thesis.

1.1 Study design

This project is intended as a pilot to find out how data science can improve the effectiveness of reintegration. Therefore the research question for this project is:

> 'How can we use data mining techniques on social security data to improve the effectiveness of reintegration?'

To illustrate the steps involved and the study plan Figure 1.1 is used.

[1] `http://ec.europa.eu/eurostat/en/web/products-datasets/-/UNE_RT_M`

[2] `https://www.rijksoverheid.nl/onderwerpen/prinsjesdag/inhoud/miljoenennota-rijksbegroting-en-troonrede/huishoudboekje`

[3] `www.ynformed.nl`

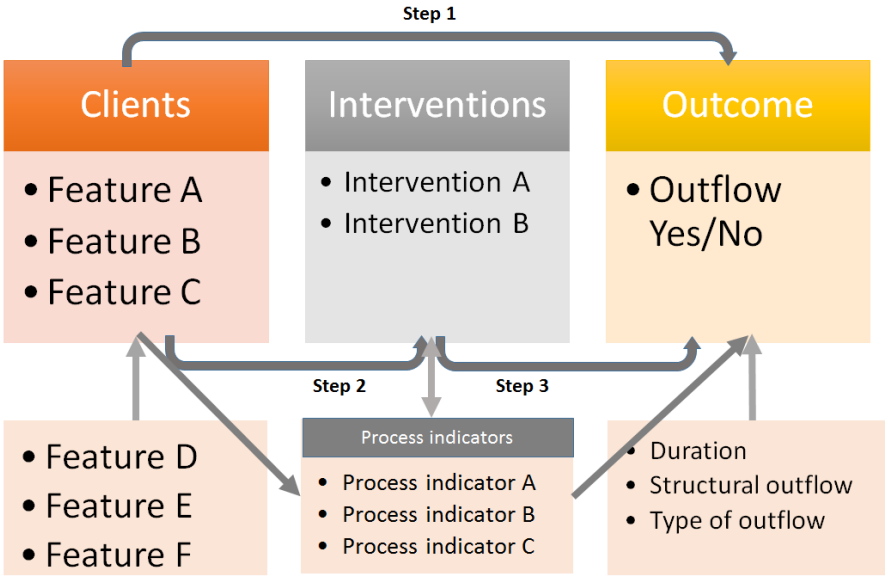


Figure 1.1: Diagram that illustrates the design of this study. The left blocks indicates the clients and their features, the middle blocks indicate the intervention instruments and the right blocks indicate the outflow. The steps correspond to the sub-questions.

As we can see in Figure 1.1 some sub-questions can be defined to specify the research:

1. Can client features be used to predict the outflow from social security?

   This sub-question can give us more insight into groups that have a small or large probability of outflow. The outcome is influenced by the intervention instruments, so if we find groups of clients that have a large chance of outflow than this might also be the effect of the intervention instruments. A classification model can thus give us insight into the effects of the client features given the current intervention policy.

2. Can client features be used to predict the outflow from social security for clients that have participated in a specific intervention?

   The answer to this sub-question might help to find groups of clients for which an intervention does or does not seem to help. Note that we can't answer this question for clients that have not received the specific intervention.

3. Which intervention(group) has a higher outflow rate than other intervention(group)s?

   This sub-question will give more insight into the effectiveness of an intervention. We also might have to adjust for bias introduced by the assignment of interventions, which is not at random. Solving this problem is a large part of this research, see Section 3.2 for a more in depth look at this problem.

1.2 Research methodology

The research methodology used in this research project is CRISP-DM (CRoss-Industry Standard Process for Data Mining), which is clearly explained in the article by Colin Shearer[25]. CRISP-

DM is a data mining framework that can be used in practically every data mining project. It consists of six steps: business understanding, data understanding, data preparation, modelling, evaluation and deployment (see Figure 1.2). Furthermore, this project will also follow the Scrum paradigm, including weekly Scrum-meetings and four sprints of three weeks.

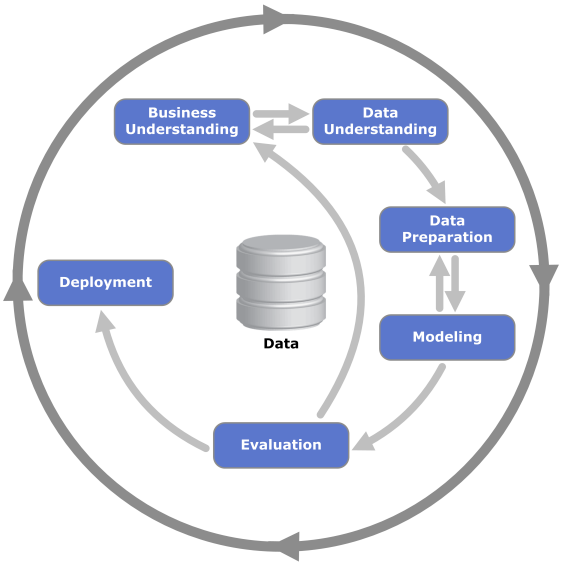


Figure 1.2: The CRISP-DM framework. Arrows indicate the order of the steps, the outer arrows denote the cyclic nature of the process. (By Kenneth Jensen, licensed under `http://creativecommons.org/licenses/by-sa/3.0/`)

Below the steps are briefly explained.

Business understanding In this step the business objectives are stated. These objectives are then transformed into data mining goals and a project plan is made. These steps are mostly covered in this chapter.

Data understanding In this step the data is collected and explored on its surface properties. Furthermore, the quality of the data will be verified.

Data preparation In the data preparation step the data is prepared for modelling by combining or selecting data and by cleaning the database. The data understanding and preparation steps will be covered in Chapter 2.

Modelling In this step modelling techniques are selected and applied on the prepared data set. Furthermore, the model is assessed on the quality of the outcome.

Evaluation In this step the generated models are evaluated to ensure that the outcome is the answer to the original business objectives and the next steps are chosen.

Deployment In this step the model is presented in such a way that it satisfies the client goals. For instance, as a presentation or as part of a tool. The modelling, evaluation and deployment steps are discussed in Chapter 5.

As we can see in Figure 1.2 most steps are cyclic, especially Data Preparation and Modelling since it is often necessary to modify the data if we want to use a different model.

1.3 Background on social security

Social security is organised differently in every country. In the Netherlands if a person loses his job then he can first apply for unemployment insurance (*Werkloosheidswet*), which is a temporary insurance that lasts at most three years, if the client has worked long and frequent enough before losing his job. In the Netherlands this is regulated by the UWV (*Uitvoeringsinstituut Werknemersverzekeringen*). The unemployment insurance is not part of this research.

If someone has not worked (enough) before losing his job or if the unemployment insurance ends before finding a job and if he does not have enough capital then he can apply for social security (*Participatiewet*, former *Wet Werk en Bijstand*). The social security is regulated by the individual municipalities.

These municipalities can within the boundaries of the law, make their own rules and are also financially responsible for paying both the social security and the intervention instruments.

Social security can end for various reasons:

Work The client finds a full-time job

Schooling The client is fit for (additional) schooling

Income The client receives income from another source

Enforcement The client is not eligible for social security, or the client does not keep to agreements

Transition The client moves to another municipality, dies, gets incarcerated or starts a relationship

Remaining The partner exits the social security for one of the above reasons or other reasons

These outflow reasons are used throughout the land and are drafted by Divosa[4].

The amount of social security benefit received may also be reduced for various reasons, such as receiving part-time income or having cost-divisors. People (children and partners excluded) who live together with a client are counted as cost divisor since they share in their costs. Therefore the social security is reduced on the basis of the number of cost divisors.

This research focusses not only on the social security records but also on the intervention instruments. Intervention instruments are treatment programs for clients which can help them in various ways. There are many possible intervention instruments, but we will focus on the following interventions:

Intervention E Nowadays everyone receives this intervention. It requires its participant to perform simple manual labour to test their perseverance and skills. This intervention instrument will not be analysed in the matching, since everyone should get this instrument removing the possibility of a control group.

Intervention A This intervention instrument gives clients a temporary workplace to get working experience. This intervention is typically given to clients that have a greater distance to the labour market.

Intervention B This intervention instrument actively helps clients to get a job. This intervention instrument is mostly assigned to clients with a smaller distance to the labour market.

[4] `www.divosa.nl`

Intervention C This intervention instrument is actually a collection of instruments meant for clients with a smaller distance to the labour market.

Intervention D This intervention instrument is also a collection of instruments. These intervention instruments are however meant for clients with a greater distance to the labour market.

The last two intervention instruments are grouped together by experts from the municipality, based on their target audience and purpose.

Chapter 2

Dataset

In this chapter we will discuss the available dataset and preprocessing steps. In Section 2.1 a description of the data will be given along with the challenges that are present in this dataset. In Section 2.2 the pre-processing steps will be discussed.

2.1 Data understanding

2.1.1 Dataset description

The available data is structured in several CSV (Comma Separated Value) files, which are exported from various database systems. The reference date of all datasets is April 2016. Most of the data lies between 2011 and April 2016. There is also some data from before 2011, however in 2011 a new administration system was put into operation and only active records were transferred (see Figure 2.1). Table 2.1 gives an overview of the datasets.

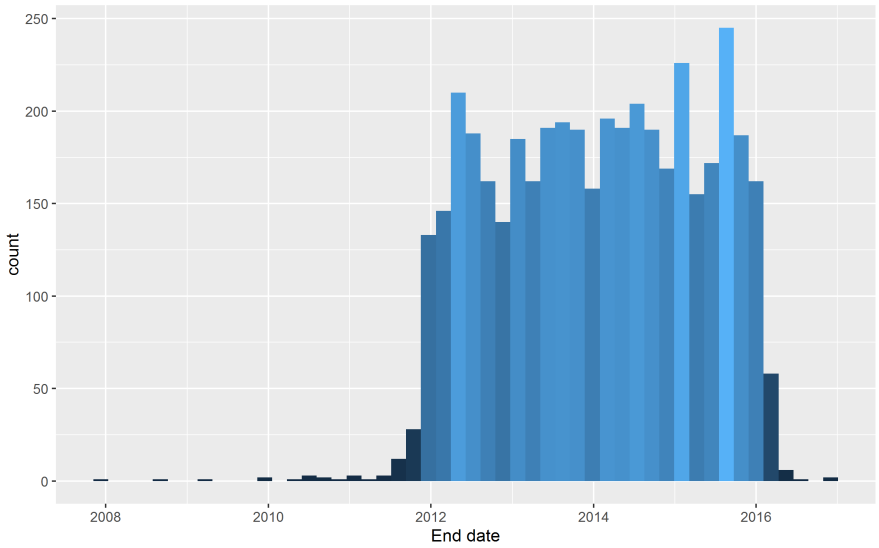


Figure 2.1: Histogram of the end dates in years of the social security records from 2006.

Dataset name	Number of records	Description
Client_Kenmerken	10391	Birth-date, nationality, gender, marital status, living arrangements, housing and number of children
Buurten	10311	Neighbourhoods
Geboorteland	7058	Native countries
Opleiding_Taalniveau	9891	Education and language levels (contains many missing values)
Kostendelers	1126	Cost-divisors
Schuldhulpverlening	1599	Debt relief
Kenmerk_aggressief	63	Clients who have been found aggressive
Competensys	4079	Survey data about education, social skills, addiction etc.
Instroom_Uitstroom_Partner	9574	Social security records, contains start and end date and reason of outflow
Agendaplanner	52874	Appointments and presence of clients (4086 clients)
Maatregelen	2776	Penalty statements
Boetes	836	Fines
Dvl_Voorzieningen	40370	Intervention instruments (9660 clients)
Parttime_Inkomsten	18750	Declarations for part-time income (2170 clients)

Table 2.1: Names, sizes and description of the datasets used in this research

Three very interesting covariates from the Competensys dataset are *Loonwaarde*, *Participatie_trede* and *Werk_afgelopen_jaar*. *Loonwaarde* (Wage value) is an estimate of the percentage of paid work a client can handle. So a *Loonwaarde* between 80 and 100 percent indicates that the client most likely will be able to get a full-time job, while a client with a *Loonwaarde* between 0 and 30 percent might never be able to handle a full-time job.

Participatie_trede (Participation step) indicates the level in which a client might participate in society and it ranges from 1 (isolated) to 6 (paid job).

Werk_afgelopen_jaar (worked past year) indicates if the client has worked in the past year.

2.1.2 Challenges of the datasets

For every client there is potentially a lot of data, however not every client appears in every dataset. For some datasets this introduces no problems. Take for example the debt relief dataset, if a client does not appear in the dataset then he has no debt relief. However if a client does not appear in the neighbourhoods dataset, then his neighbourhood is unknown. See Figure 2.2 for an example of the overlap between the datasets.

Another challenge with this data is unknown values in the datasets themselves. Since the datasets are the result of an administrative process, values that are not known at the time or not important are not filled in. This might lead to problems later on. See Figure 2.3 for a missingness

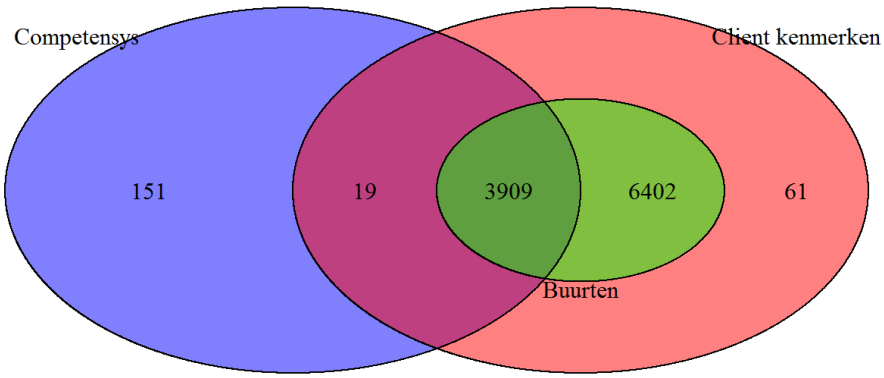


Figure 2.2: The overlap between the Competensys, neighbourhoods (*Buurten*) and client data (*Client_kenmerken*). It can be clearly seen that the Competensys data is only known to part of the clients, which introduces a lot of unknown values.

map of all datasets after joining on client number.

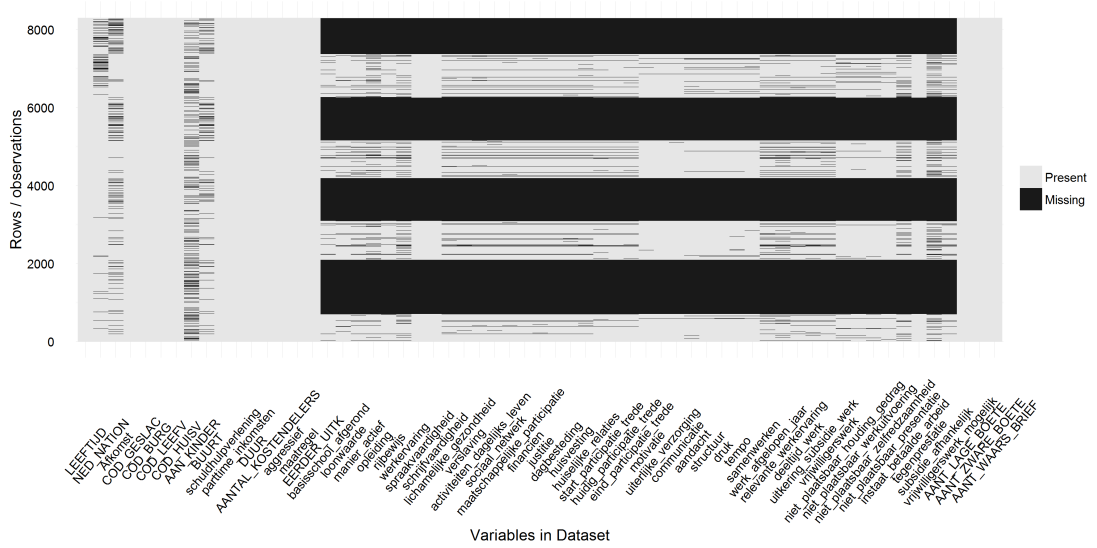


Figure 2.3: The missingness map for the data. The large black stripes represent the missing data introduced by the Competensys data.

Next to the missing values, some covariates might also change without our knowledge. For example the number of children is only administrated when it is necessary, which means that it is usually recorded at the start of the social security but never updated. Also some data might be

entered differently depending on which consultant entered the data. For example if a client has no children one consultant might administrate 0 as value, while another one might leave it blank, thus creating an unknown value. In this case we assumed that a 0 really means no children and an empty field means unknown.

Furthermore many datasets, such as the data from Competensys are filled in by both the client and consultant based on their conversations. Although the quality of this data is not easily verified, experts at the municipality guarantee that this is the most up-to-date and complete data.

2.2 Data preparation

Before we can use the data it has to be cleaned and ordered. To easily access the data a MySQL database is used and all CSV files are imported as database tables. Then all data is formatted (numbers to Integer, dates to Date objects etc.), obviously wrong data is changed (for example 19-11-2105 to 19-11-2015) by manually checking the maximum and minimum date-values and missing values are denoted by NULL.

Some values have too many categorical values, for example nationality which has 88 different possible values. These values are categorized (e.g. nationality becomes the binary variable 'Dutch nationality'). The reason of outflow is also categorized into the six categories defined in Section 1.3.

The next step is to create a data table where every row corresponds to a social security record from a client, that contains all (important) covariates and the response variable(s). To create this table we join all data sets about a client with the social security data set. Furthermore if a client has two or more social security records that are less than three months apart we combine them into one record to avoid analysing recurrent clients or administrative transitions. The reason of outflow for this combined record will be the final reason of outflow of the last record (or none if the clients is still in social security). This data table can then be used in the subsequent analysis steps (see Chapter 4).

To incorporate the intervention programs we also create a data table combining the social security records and intervention programs. In this table each row corresponds to an intervention program that falls into a social security record. Intervention programs that do not fall into a social security record are not taken into account, since we can never know the result of such an intervention. The resulting data table thus consists of all information in the previous data table combined with the intervention instrument followed and the start time of this intervention.

Since intervention instruments can start up to three months before the start of the social security we subtract three months from the start date of the record. Then we combine an intervention program with a social security record when more than ten percent of the program falls in the period of the social security record. This is done to avoid assigning a treatment to a social security record when the treatment falls mostly in the previous record.

All preprocessing steps have been discussed with experts working for the municipality.

2.2.1 Imputation

As described in the previous section, there are a lot of missing values. Some machine learning methods, such as classification trees can handle missing values. However if we want to use a method that does not handle missing values we have to get rid of them.

There are three ways to get rid of missing values:

1. Remove the rows that contain missing values
2. Remove the columns that contain missing values

3. Impute (fill in) the missing data

Since using only the first two approaches can remove potentially important information we opt for a combination of the three approaches.

For the imputation we use Single Imputation (SI). For more details about the imputation algorithm see Section 4.3.1.

Because Competensys has proven to be important information and since more than half of the clients have no Compentensys data we filter out all the client that have no known Competensys data. We then impute the unknown values in the remaining rows.

We could also have imputed all missing values, however since more than half of the Compentensys values are unknown, single imputation would increase the uncertainty of the data too much. Furthermore, since Competensys contains much more useful data than the rest we can't simply ignore it, since that would increase the bias in the dataset.

Chapter 3

Theoretical background

In this section related work will be discussed to provide a theoretical background on analysis on unemployment data. In Section 3.1 related work which involves survival analysis will be discussed. In Section 3.2 we will discuss related work that uses matching. In Section 3.3 we will discuss literature about data science in the domain of social security. In Section 3.4 Dutch literature about the Dutch social security will be discussed and finally in Section 3.5 the problems of causality vs. correlation will be addressed.

3.1 Survival analysis

To analyse the social security records and the effect of the intervention instruments, it is useful to look at the length of the social security. A popular technique for this situation is survival analysis.

Survival analysis[18] is most commonly known in the medical field to predict the survival time, for example for leukaemia patients[3]. This technique might also be of use in this project, because social security records are somewhat similar to medical survivor records. In both cases clients receive 'treatment' during a certain period and clients can have an event (outflow) or clients can be censored (still receiving social security at the end of the study or leaving the study prematurely).

A paper that resembles the topic discussed in this paper is the paper by Bruce D. Meyer[22] where they use survival analysis to analyse the unemployment insurance duration and height of men from twelve different states in the United States. They found that a higher benefit negatively impacts the chance of leaving unemployment and that the chance of leaving unemployment rises in the weeks prior to when benefits lapse. As explained in Section 1.3 there is a difference between the unemployment insurance and social security. The project described in this paper focuses solely on social security data.

For survival analysis it is often useful to estimate the survival curve for an entire group. One of the estimators often used for this task is the Kaplan-Meier estimator[15]. When we want to know the influence of different covariates on the survival time we can use a (semi-)parametric survival model. One of the most popular models is the Cox proportional hazards model[10]. This model can handle multiple covariates and we can estimate the effect of these covariates on the length of the unemployment duration. The paper by Alenka Kavkler et al.[16] uses Cox regression models to compare the unemployment duration in five Central and Eastern European countries. They found differences in unemployment duration between the various countries and

they also found that the differences between age groups become less pronounced the longer the unemployment spell lasts.

Another paper by S. van Buuren et al.[29] uses survival analysis to find a relation between missing blood values and the survival rate. To impute the missing blood values they used Multiple Imputation (MI). The combination imputation and survival analysis used in the previous paper will also be of use in this project. Multiple imputation and other missing data techniques are also discussed in the paper by Shafer and Graham[24].

3.2 Matching

To help the outflow of clients, the effectiveness of intervention instruments has to be analysed. The paper by Seamus McGuinness et al.[21] describes the effect of six types of training on the employment chances in Ireland. They found that the clients that participated in a training were less likely to be unemployed at the end of the study period. However, the effect varies by the type and duration of the training received. These training programmes are not assigned at random. For example, consider a training that helps young people manage their finances better. This training will only be assigned to young people, so we might bias the effect of this training since young people tend to get a job faster regardless of the training. Therefore, Seamus McGuinness et al. use a matching algorithm called propensity score matching[23] which accounts for the covariates that predict the type of training to reduce the impact of this bias. Propensity score matching is a very popular matching method, but it is not without criticism. As King and Nielsen describe in their paper[17] propensity score matching tries to approximate a randomized experiment, where both the observed and the unobserved covariates are in balance between treatment and control *on average*. However other matching methods approximate a fully blocked experiment, where the observed covariates are (almost) exactly in balance and the unobserved covariates are balanced on average. Therefore propensity score matching might even increase the imbalance and it might be a good idea to look at alternatives. An alternative to propensity score matching is using the Mahalanobis distance[19]. Instead of using one dimension for matching it calculates the distance using all the covariates, similar to Euclidean distance, but it also takes the spread of the data into account.

Another paper that describes the treatment effect of training programmes is written by Bram van Dijk[30]. In his paper he describes the effect of three different job-training programmes on the unemployment duration in Slovakia. To correct for the non-random assignment of training programmes he incorporates the unobserved heterogeneity into a multivariate mixed proportional hazards model. He found that two of the three programs shorten the unemployment duration substantially. The two papers of McGuinnesss et al. and van Dijk resemble our situation, however in our case the number of training programmes is much higher and the number of records is smaller.

3.3 Social security data mining

As said before not much work has been done on social security data mining, however there is a special interest group that specializes in social security data mining[1]. The members of this group have published papers about different kinds of social security data mining such as the paper by H. Zhang et al.[32] where sequential pattern mining is used to train a sequence classifier that can be used for debt prevention in the social security area. The paper by Longbing Cao[9] gives

[1] `http://datamining.it.uts.edu.au/ssdm/`

an overview of the field of social welfare and social security data mining. Almost all papers published by this group use a combination of (sequential) pattern mining and classification.

Another paper by W. Xu et al.[31] takes a different approach and tries to predict the unemployment rate using search engine query data, instead of the social security data itself. They conclude that their tool can be used to analyse and predict the unemployment rate without delay.

3.4 Dutch reports on social security

The implementation of social security differs a lot between countries. Therefore it is useful to look at Dutch papers about social security. These papers are often sociological and use surveys or simple statistical models instead of data mining. Nevertheless, they can still be informative and give more insight into the domain of social security and intervention instruments. The new social security legislation called the Participation Act was installed in 2015 and uses among other methods wage subvention as an intervention instrument (see Borghouts et al.[6]). Other reports about social security are the quick scan[4] commissioned by the ministry of social affairs and employment, which gives an overview of scientific papers about social security and a report about the effectiveness of reintegration instruments[5]. According to this report using a waiting period before starting the social security procedure has a great effect on the inflow, furthermore participating in an intervention program has more effect on the outflow chance than doing nothing but this effect only occurs after thirty weeks. Note that in our research we do not consider the inflow, but focus only on the outflow. Also useful is a report about the effectiveness of reintegration in the unemployment insurance[13], which concludes that effectiveness of reintegration instruments depends on the client group and the moment the instrument is used.

3.5 Causality vs. correlation

One of the focusses of this research is to examine the effectiveness of intervention instruments, but we must keep in mind that there is a difference between prediction and causality. In the first case we want to train a model that makes predictions about the dependent variable using the observed values of the independent variables whereas in the latter case the independent variables are regarded as causes of the dependent variable. Furthermore, when dealing with missing or unobserved variables we can still get a good prediction, but the causality relation might be false since important information is missing or unobserved. This dilemma is described in the first chapter of the book by Paul Allison[1].

Next to the dilemma of causality vs. correlation we are also interested in quantifying the causal effect of the treatment. Consider for example a regression model, with a binary variable indicating treatment (T), client features (X) and the outcome (O). If we want to quantify the effect of the treatment, we must also include the client features that influence the assignment to the treatment and the outcome, otherwise we might over- or underestimate the effect of the treatment on the outcome. See Figure 3.1 for an illustration.

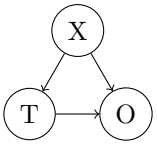


Figure 3.1: An illustration of the effect of the client features (X) and the treatment (T) on the outcome (O)

Chapter 4

Methods

In this chapter the different models and algorithms that are used in this project are discussed. In Section 4.1 the different survival analysis models and metrics are discussed. In Section 4.2 the matching methods are discussed and in Section 4.3 the imputation of missing data is discussed.

In this project we will mostly use R as a programming language. The book by Peter Dalgaard[11] gives an introduction to statistical and data mining tasks in R, including survival analysis.

4.1 Survival analysis

Survival analysis is an analysis method that models time-to-event data. It has many applications in the medical domain, but can also be used to model the time until failure for machines or time until recidivism for ex-convicts. In this case we use survival analysis to model the time it takes until a client exits the social security.

An event in survival analysis corresponds to death or failure in most examples, luckily our event is a lot more positive. The time component can be viewed as survival time; the time period a client survives in the dataset. Notice that long survival times are in this case unwanted, because we want a client to flow out of the system as soon as possible.

Another aspect of survival analysis is censoring. There are two types of censoring: left and right censoring. Left censoring means that a client gets the event before it is observed, either because he receives the event before the start of the study or because the observation of an event always happens after the first exposure (for example when testing for a disease that manifests itself later).

Right censoring means that the observed survival time is shorter than the actual survival time. There can be two types of right-censoring:

- The client experiences the event after the study period (or never)
- The client withdraws from the study (for example if he moves to another city, if he is being detained, or if he exits the social security for other reasons)

In our case we only have to deal with right-censored data, since we immediately know when a client exits the social security. As opposed to regression models, survival analysis uses the censored data up until the point where censoring occurs, since the client that experiences censoring would also survive until that time if censoring had not occurred.

Survival analysis uses two quantitative terms; the survival function ($S(t)$) and the hazard function ($h(t)$). The survival function gives the probability that a client survives longer than some time t:

$$S(t) = \Pr(T > t), \tag{4.1}$$

where T is the survival time. The survival function is a monotonically decreasing function:

$$S(t_1) \geq S(t_2) \; \forall t_1 < t_2, \tag{4.2}$$

which follows from the definition of the survival function, since the survival rate can never increase when t increases. The hazard function is given by:

$$h(t) = \lim_{\Delta t \to 0} \frac{\Pr(t \leq T < t + \Delta t | T \geq t)}{\Delta t}. \tag{4.3}$$

The hazard rate can be explained as: "The instantaneous potential per unit time for the event to occur, given that the individual has survived up to time t".[18]

There is a clear relation between the hazard function and the survival function. If the hazard function is a constant λ then the following function describes the relation between the two functions:

$$h(t) = \lambda \Leftrightarrow S(t) = e^{-\lambda t}. \tag{4.4}$$

For most survival analysis tasks the *survival*[27] package in R is used.

4.1.1 Kaplan-Meier estimator

The Kaplan-Meier estimator is a non-parametric method to estimate the survival function[15]. To calculate the survival function a product limit function is used:

$$S(t_j) = \prod_{i=1}^{j} \Pr(T > t_i | T \geq t_i), \tag{4.5}$$

where t_i is the i'th failure time and $\Pr(T > t_i | T \geq t_i)$ can be calculated by dividing the number of observations that survive past time t_i by the total number at risk at time t_i. Observations are at risk when they have not yet experienced the event and they are also not (yet) censored. Note that since this is a non-parametric method, it does not depend on any of the covariates, nor can it be used as a good model to predict survival times for new observations. It does however give us an estimation of the survival function for the data set and we can easily plot the resulting curve (see Figure 4.1). Since the Kaplan-Meier estimator calculates the survival function only at every failure time, the Kaplan-Meier estimator produces a step curve, while the actual survival function should result in a smooth curve. However when the number of failure times increases, the Kaplan-Meier estimator will resemble the actual survival curve.

4.1.2 Cox proportional hazards model

The Cox proportional hazards model[10] is a very popular method for survival analysis. It is a semi-parametric model and instead of directly defining the survival function it defines the hazard function as:

$$h(t, \mathbf{X}) = h_0(t) \exp\left(\sum_{i=1}^{p} \beta_i X_i\right), \tag{4.6}$$

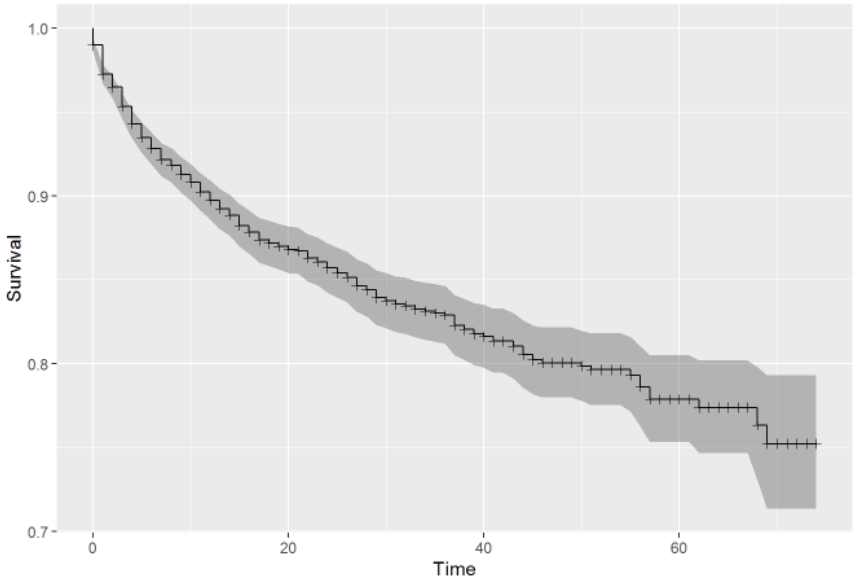


Figure 4.1: A Kaplan-Meier plot. A step indicates a failure time where one or more clients experience an event. The tick marks indicate censoring and the shaded areas correspond to the confidence intervals.

where $\mathbf{X}$ is the vector of covariates. $h_0(t)$ is the baseline hazard which does not contain any of the covariates while the exponential expression contains the covariates but not the time. Therefore this model can only handle variables that do not change over time. Though there exist implementations of this model that can handle time-varying variables[28]. Because the baseline hazard remains unspecified the model is called semi-parametric.

If we divide both sides of the previous equation by $h_0(t)$ we get de definition of the hazard ratio:

$$\frac{h(t)}{h_0(t)} = \exp\left(\sum_{i=1}^{p} \beta_i X_i\right) \tag{4.7}$$

If we take the logarithm of both sides the right-hand side becomes a linear combination of covariates and coefficients similar to a linear regression model.

Since the baseline hazard remains unspecified, we do not have to make any assumptions about the distribution of the hazard function. However since the Cox proportional hazards model can only handle time-independent variables, the effect of each variable is assumed to be constant over time. This assumption is called the proportional hazards assumptions, which is the basis of the Cox proportional hazards model.

4.1.3 Parametric models

An alternative to the Cox proportional hazards model is a parametric model. Parametric models fit a specified distribution in terms of unknown parameters to calculate the survival function.

Examples of parametric models, with their survival function are:

Exponential $S(t) = e^{-\lambda t}$

Weibull $S(t) = e^{-\lambda t^p}$

Log-logistic $S(t) = \frac{1}{1+\lambda t^p}$

Here the λ is parametrized by the covariates and other parameters, while p is held fixed. Note that when $p = 1$, Weibull is the same as exponential. Furthermore the hazard function for the exponential model is only the constant λ, so the proportional hazards assumption will always hold in this case.

Not all parametric models are proportional hazards models, though many are Accelerated Failure Time models. The AFT assumption is that the effect of the covariates is proportional with respect to the survival time.

4.1.4 Survival tree

A very different approach to survival analysis is using a Classification and Regression Tree (CART). Classification trees use recursive partitioning to split the data in groups using a splitting criterion.[7]

For Survival Analysis, the splitting criterion that is mostly used is the log-rank statistic, which will be discussed in Section 4.1.5. The algorithm splits the data on the split with the best log-rank score until a stopping criterion is met, for instance minimum split size or bucket size.

The principle of a survival trees can also be extended to a survival random forest, which trains multiple survival trees on samples of the data and covariates and then pools the result of all the trees.

For creating survival trees the *party* package is used and for creating the Survival Forests the *randomForestSCR* package is used.

4.1.5 Log-rank test

The log-rank test or Mantel-Cox test[20] is a statistic that tests if two or more groups are statistically equivalent, with respect to their survival function. This test is used in the splitting criterion of survival trees, but it can also be used to test for differences in multiple Kaplan-Meier curves.

The log-rank test first calculates the expected cell counts for each failure time $j \in \{0, 1, ..., q\}$ for group $i \in \{1, 2\}$ between (in this case) two groups:

$$e_{i,j} = \left(\frac{n_{i,j}}{n_{1,j} + n_{2,j}} \right) \times (m_{1,j} + m_{2,j}), \tag{4.8}$$

where $e_{i,j}$ is the expected cell count for group i at time j, $n_{i,j}$ is the total number at risk for group i at time j and $m_{i,j}$ is the number of failures in group i at time j.

Then the log-rank statistic is calculated using the following formula:

$$\text{Log-rank statistic} = \frac{(O_2 - E_2)^2}{\text{Var}(O_2 - E_2)}, \tag{4.9}$$

where $O_2 - E_2 = \sum_{j=1}^{q} (m_{2,j} - e_{2,j})$. The estimated variance can be calculated as follows:

$$\text{Var}(O_i - E_i) = \sum_{j=1}^{q} \frac{n_{1,j} n_{2,j} (m_{1,j} + m_{2,j})(n_{1,j} + n_{2,j} - m_{1,j} - m_{2,j})}{(n_{1,j} + n_{2,j})^2 (n_{1,j} + n_{2,j} - 1)}. \tag{4.10}$$

Note that in this case we defined the log-rank statistic in terms of group two, however using group one will yield the same result.

The null hypothesis being tested is that there is no significant difference between the groups. Since the log-rank statistic approximates a chi-square distribution with one degree of freedom, the p-value can be read from the tables of the chi-square distributions. A p-value < 0.05 indicates that the null hypothesis should be rejected, indicating a clear difference between both groups.

4.1.6 Concordance index

The Concordance index (or c-index)[12] is a popular metric for survival analysis. In contrast to a mean squared error metric, the concordance index only considers the order of the predicted event times.

To compute the Concordance index we first form all pairs of observations. Then remove all pairs where the shorter time is censored and remove all pairs where the survival time is the same and they are both censored. Now we can use the following formula to compute the c-index:

$$\text{c-index} = \frac{\#Concordant + 0.5 \times \#Ties}{\#Pairs}, \tag{4.11}$$

where a predicted pair is concordant if the predicted order is the same as the actual order and a pair is tied if the predicted survival time for a pair is the same. Note that a c-index of 1 denotes perfect concordance, while a c-index of 0.5 means that the model performs not better than a random order.

Instead of calculating the c-index using a model that is trained on the entire dataset it might be better to use a bootstrapping approach.

Bootstrapping creates a dataset by sampling from the dataset with replacement, where the sample contains as many rows as the original dataset. Since we sample with replacement there will be duplicates in the sample. This sample is then used as a dataset to train the model and the rows that are not sampled are used as test set. We finally take the average over all samples to calculate the bootstrapped c-index. In our case we use 100 bootstrap samples to get a reliable estimate.

To calculate the c-index we use the *validate* function from the *rms* package, which gives us the Somers' D score[26] which can be easily transformed to the Concordance index using:

$$D_{xy} = 2 \times (\text{c-index} - 0.5). \tag{4.12}$$

4.2 Matching

If we want to analyse the effect of a treatment, we could look at the Average Treatment Effect (ATE) which is defined as:

$$\text{ATE} = E(y_1|x, T = 1) - E(y_0|x, T = 0), \tag{4.13}$$

where y_1 is the outcome for the treatment group and y_0 is the outcome for the control group, x is the vector of covariates and T is a binary variable indicating treatment. While this might work fine in a random experiment, in our case this will not suffice, since the assignment to treatment is based on the features of the client and not at random. For an example of why this will not work, see Section 3.2.

A better way to analyse the treatment effect is to look at the Average Treatment Effect on the Treated (ATET):

$$\text{ATET} = E(y_1|x, T = 1) - E(y_0|x, T = 1). \tag{4.14}$$

The second term is counter-factual and can never be observed. Therefore we have to estimate the second term by matching every observation from the treatment group to a similar observation from the control group. Then the difference in outcome can be attributed to the treatment.

In this section we will discuss different matching procedures to find for each observation from the treatment group a similar observation from the control group, resulting in an increased balance between the treatment and control group.

When using a matching method all covariates that influence the outcome or the assignment to treatment must be present in the model. However this leads to a difficulty in our case.

Time is important in the assignment of treatment, because some treatments are only assigned after some time in social security. Not including time as a covariate in the matching might lead to an unfair matching. Consider a treated observation T with failure time x_T and treatment start time t_T which is matched to an observation from the control group C with failure time x_C (see Figure 4.2).

Suppose that x_C is smaller than t_T. This results in an unfair matching since C did not get the chance to participate in the treatment, because he exited the social security before t_T.

However including time is not fair either since it is also part of the outcome (time to event). Including time would increase the balance of the outcome between the treatment and control group, which leads to a smaller difference between the groups.

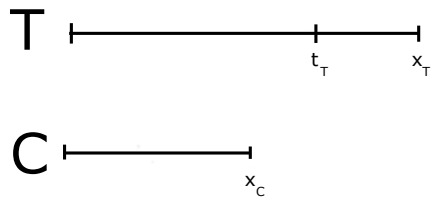


Figure 4.2: An unfair matching between a controlled observation C and a treated observation T.

Our approach is to use a customized matching where for every treated observation T a match can only be made to an observation from the control group C if $x_C >= t_T$ (see Listing 4.1).

Listing 4.1: Pseudo code for the matching. Calculating the distance can be done using PSM or some other distance measure.

```
for(i in 1:row(treatment)) {

    # Control has to have a longer survival time
    # than start of treatment
    pos_matched <- control_group[time >= start_time_voorz[i]]

    # Filter out matches already made
    pos_matched <- pos_matched[!(ID %in% control_matched$ID)]

    best_match <- min(distance(treatment[i], pos_matched))

    if(row(best_match) > 1)
        arbitrarily break ties

    control_matched <- combine(control.matched, best.match)
}
```

The matching can be divided into two steps:

1. Calculate the distances between the treatment group and the control group (e.g. propensity score, Mahalanobis distance)
2. Use a matching method to find a control group that closely matches the treatment group (Nearest Neighbour, Caliper matching)

For the matching phase we will use nearest neighbour matching. Note that this leads to a matched control group that has the same size as the treatment group. We could also have used a matching with replacement which would lead to a smaller control group or match every treatment unit to two or more control units which leads to a larger control group. In the next subsections the different distance metrics will be discussed.

4.2.1 Propensity score matching

One of the most popular matching procedures is propensity score matching (PSM)[23].

The propensity score is defined as the chance that a client receives treatment:

$$p(\mathbf{X}) = Pr(T = 1|\mathbf{X}), \tag{4.15}$$

where $\mathbf{X}$ is the vector of covariates and T is a binary variable that indicates if the client has received treatment. The Propensity score can be estimated using a logistic regression model, where the outcome variable indicates participation in the treatment. This step effectively reduces the dimensions to one and the distance is calculated as follows:

$$d(\mathbf{x}, \mathbf{y}) = |p(\mathbf{x}) - p(\mathbf{y})|. \tag{4.16}$$

A downside of this method is that if two clients have the same propensity score, they don't necessarily have the same covariate values. Furthermore as mentioned in Section 3.2 PSM approximates a random experiment as opposed to the stronger fully blocked experiment, which might increase imbalance.

4.2.2 Normalized distance matching

Another approach is to use a distance metric such as the Euclidean distance. First the data is scaled so that every covariate has the same mean and standard deviation. If we do not scale than age for example would be more important in the matching than a binary covariate, since the difference in age between two clients can easily be greater than the difference in some binary covariate. Therefore scaling makes sure that every covariate is equally important in the matching. The distance after scaling can then be calculated as follows:

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}, \tag{4.17}$$

where $\mathbf{x}$ and $\mathbf{y}$ represent two observations and x_i and y_i represent the i'th covariate for both observations. The distance can then be used in the matching phase to find a closest match.

4.2.3 Weighted distance matching

Instead of using just the Euclidean distance we can also use different weights for every covariate. To calculate the optimal weights for every covariate a genetic algorithm is used. Then the weights can be used to calculate a weighted distance:

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{n} w_i \times (x_i - y_i)^2}, \tag{4.18}$$

where w_i is the weight for the i'th covariate. For the genetic algorithm the function *genMatch* from the *Matching* library is used with a population size of 100 and a maximum of 100 iterations.

4.2.4 Mahalanobis distance matching

Instead of Euclidean distance it is common to use the Mahalanobis distance[19]. The Mahalanobis distance is calculated as follows:

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T \Sigma^{-1} (\mathbf{x} - \mathbf{y})}, \tag{4.19}$$

where $\mathbf{x}$ and $\mathbf{y}$ are the two observations and Σ is the covariance matrix, which contains the covariance between all pairs of covariates. Instead of just looking at the differences between the observations it also takes the covariance of the different variables into account. Note that this requires that the covariance matrix can be inverted. When the covariance matrix is the identity matrix then the Mahalanobis distance reduces to the Euclidean distance. Note that scaling prior to calculating the Mahalanobis distance is not needed, since the Mahalanobis distance scales all covariates based on the covariance matrix.

4.3 Missing data

As discussed in Section 2.2.1 we use imputation to fill in missing values. When dealing with missing data we can distinguish three patterns of missingness[24]:

- Missing at Random (MAR)
- Missing Completely at Random (MCAR)
- Missing Not at Random (MNAR)

MAR denotes that the distribution of the missingness does not depend on the missing parts, but it can depend on the observed part. MCAR denotes that the missingness does not depend on either the observed or the missing parts. If the distribution depends on the missing part as well then it is MNAR. If the missingness pattern is MNAR then imputing missing values becomes very difficult and we need to incorporate the reason of missingness into our model.

For multiple imputation the missingness must be MAR or MCAR. So we have to prove that our missingness pattern is not MNAR. In the ideal situation we would contact the clients that have missing data to find out if it the missingness depends on the missing data itself. However this is infeasable in our case.

However we argue that data is missing when a consultant either forgets to administrate it or if the data is not useful for their social security and intervention instruments. In that case the missingness depends either on the probability of a consultant forgetting to fill it in (MCAR) or on their social security and intervention instruments (MCAR).

4.3.1 Multiple imputation

To impute the missing data the MICE (Multiple Imputations by Chained Equations) algorithm is used.[8] The MICE algorithm works for data that is MAR or MCAR.

Multiple imputation is a technique where multiple imputed datasets are created from an incomplete dataset. Then the analysis will be done on every imputed dataset and finally the results are pooled. This decreases the uncertainty when dealing with imputed values.

MICE uses separate imputation models for every covariate using a technique called chained equations. It follows the following steps for every imputation[2]:

1. Replace the missing values with the mean for every covariate as placeholder
2. For every covariate:
   (a) Remove the placeholder and set the missing values to NA
   (b) Use a regression model (e.g. logit/probit) to impute the missing values using the other covariates
3. When every missing value is imputed, repeat step 2 for some cycles, commonly ten times to improve the imputation

In our case the missingness of some covariates is substantial. Some of the values from Competensys have a missingness of 61%. Therefore as mentioned before in Section 2.2.1 we first remove the rows of clients that have no known Competensys data and we remove some covariates that are not important. The resulting missingness can be found in Table 4.1.

Covariate	**Missingness**
Number of children	0.2539
Working experience	0.0989
Werk_afgelopen_jaar	0.0632
Etnicity	0.0368
Participatie_trede	0.0295
Neighbourhood	0.0250
Dutch nationality	0.0006

Table 4.1: Missingness table of covariates before the imputation. Covariates not mentioned in this table have no missing values.

Although the missingness of the number of children is still a bit high, the rest of the covariates all have a missingness under ten percent. Also the total percentage of missingness in the entire dataset is only 1.539%.

In this paper we will use the MICE algorithm with only one imputation, since the focus of this research is not on the imputation but on the analysis and matching steps. Furthermore the total missingness is still relatively low, so we expect that the uncertainty introduced by using single imputation is not very high. Using only one imputation makes the analysis a lot easier since we only have one imputed dataset and we do not have to pool the results.

Chapter 5

Experiments

In this section the results of the experiments will be discussed. In Section 5.1 the results from the Shiny tools will be discussed. In Section 5.2 the results from the classification models will be presented. In Section 5.3 we discuss the result from the Survival Analysis models and in Section 5.4 we will look at the effect of the intervention programs.

5.1 Analysis using Shiny-tool

To effectively receive feedback on the used models and the data, two tools are used that are written in R using the Shiny package[1]. This creates a web-interface that can automatically update plots and other objects when the input changes. The first tool uses classification trees to predict the outflow given client features. The second tool uses the Kaplan-Meier estimator to plot the survival functions for different groups.

These tools make it easier explain how these models work to domain experts. Additionally since these tools can be used to dynamically update the output we can interactively find interesting results together with the domain experts.

5.1.1 Classification trees

The first tool that was built is based on classification trees. This model is chosen because it is easy to explain how it works and it can show the effect of different covariates. The response variable is a binary variable that indicates either outflow or no outflow. The user can select which outflow reasons count as outflow and which reasons should be ignored. See Figure 5.1 for a screenshot of the tool.

To avoid categorizing social security records that have just began as the 'no outflow' category, all records that are still running for less than half a year are filtered out (this period can be altered in the tool). Furthermore multiple filters can be used to filter the dataset. Filters can be applied to the *Loonwaarde*, the age and duration of the social security, but also to the intervention instruments. In the last case clients who did not receive the selected treatment are filtered out to create a tree from clients that have followed a specific treatment. This might help to find groups of clients for which an intervention could work or not.

It is impossible to summarize all results from this tool, since it can produce a nearly infinite number of classification trees. Some interesting trees can be found in Appendix A.1.

[1] `http://shiny.rstudio.com/`

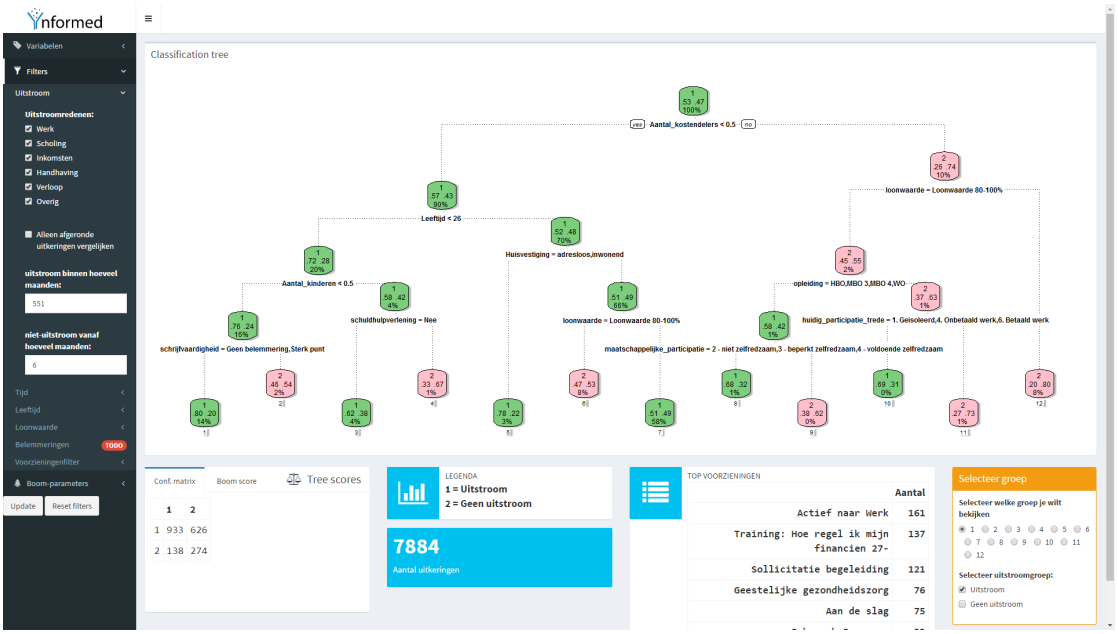


Figure 5.1: Screenshot from the classification tree tool. To the left a sidebar is shown with many possible filters and plotting options, in the center the classification tree is shown and below the tree information about the tree and intervention instruments in the leaf nodes can be found.

Some of the features that improved or deteriorated the chance of leaving the social security system were already known, however there were also some surprises. As expected a low age (< 28) and a high *Loonwaarde* were important characteristics, but more surprisingly having one or more children or one or more cost divisors significantly reduces the chance of outflow.

Interestingly when filtering on intervention instrument, two classification trees were different from most of the other trees. For *Intervention E* the most important feature to predict outflow was if someone had a Dutch nationality. For *Intervention B* two of the most important features where if someone had worked the previous year and part-time income. This might be logical since this intervention instrument is focused on actively assisting clients to get a job.

5.1.2 Survival analysis

The second tool plots the survival curves based on the Kaplan-Meier estimator. The user can customize which outflow reasons count as an event. The outflow reasons that are not selected are considered censored. Furthermore it is also possible to filter the dataset using filters comparable to the classification tree tool. See Figure 5.2 for a screenshot of the tool.

As is the case with the previous tool it can create many different interesting curves. Some of the curves can be found in Appendix A.2.

Some interesting results from the tool were:

- Males exit the social security faster and the proportion of males to exit the social security is much bigger than females
- Clients that have part-time income have a lower survival rate (except for the group with the highest *Loonwaarde*)

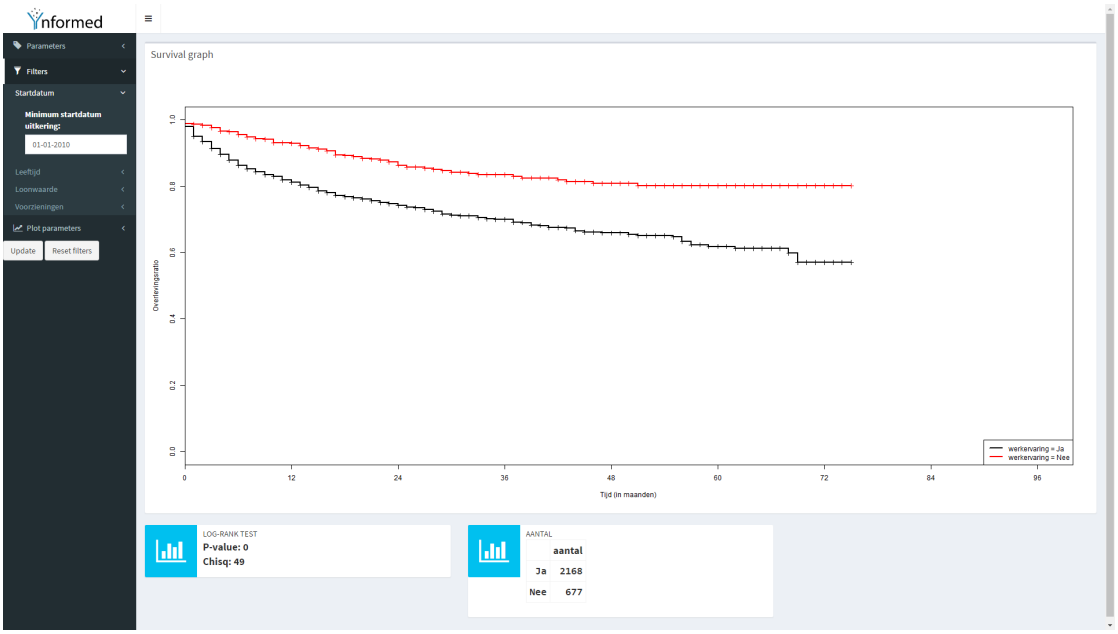


Figure 5.2: Screenshot from the Survival analysis tool. To the left the sidebar with filters and plotting options is shown, in the center the Survival function is plotted using the Kaplan-Meier estimator. Beneath the Survival function, the log-rank score and number of observations can be found.

- The biggest difference in curves was due to *Werk_afgelopen_jaar* equals 'yes', which significantly reduced the time to outflow

5.2 Classification models

For consultants, who help clients with their reintegration, it might be useful to know which clients have a high probability of exiting the social security within six years. Therefore we also trained some classification models to predict the chance of outflow.

The models used were classification tree, conditional inference tree and Naive Bayes, since the implementation of those models in R can easily handle unknown values. We assume that the reader is familiar with Naive Bayes and classification trees. Conditional inference trees are based on the framework proposed by Hothorn et al.[14]. They test for independence between the covariates and the response variable to find the covariate with the strongest association to the response variable before finding an optimal split in that covariate. According to the authors this prevents the variable selection bias and over-fitting of traditional classification trees.

Among the models Naive Bayes performed the best. In Figure 5.3 the accuracy, precision and recall are plotted against different thresholds. The plots for the other two models can be found in Appendix B.1.

This model can be used to automatically give a score to every new client that enters the social security. The consultants then can use this score to help people with a low (or medium) score first instead of clients that are bound to exit the social security faster.

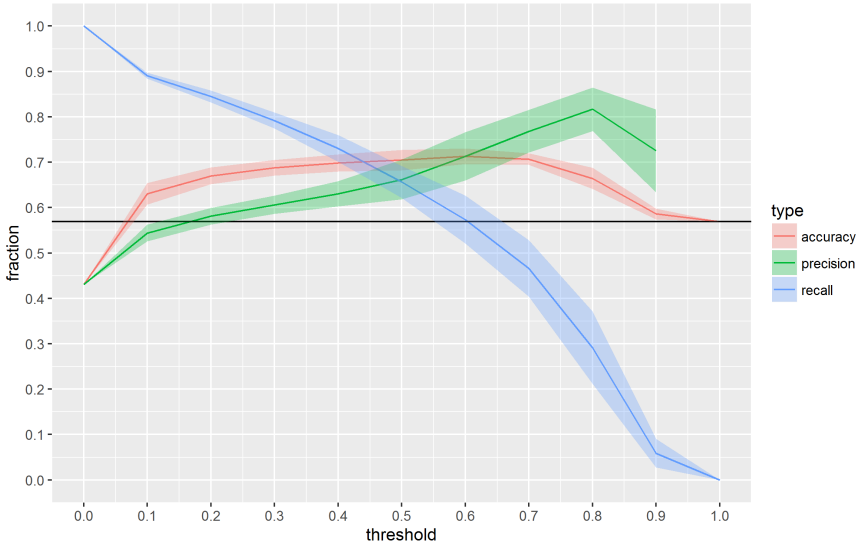


Figure 5.3: Performance of the Naive Bayes classifier. The horizontal black line indicates the baseline accuracy when we classify all cases as the majority class (no outflow).

5.3 Survival analysis models

This section describes the different survival analyses that were performed. In the remainder of this chapter the data is first imputed using the following procedure.

From all the Competensys data the covariate with most known values is found. All clients that have no known value for this covariate are removed. Covariates that are not informative or that contain too many unknown values are removed. The remaining data is imputed using MICE with one imputation and a maximum of 5 MICE iterations.

5.3.1 Cox proportional hazards model

Cox proportional hazards model is the first survival model we tested. The event is in this case outflow to a full-time job, other outflow reasons are again considered censored. The resulting output can be found in Appendix C.1.

The most significant covariates are age, gender, part-time income, *Loonwaarde*, *Participatie_trede* and *Werk_afgelopen_jaar*. The covariate with the most positive coefficient (shorter survival time) is the highest level of *Loonwaarde*, while the covariate with most negative coefficient is not having worked the past year.

The coefficients can be interpreted as follows: Consider the covariate *Werk_afgelopen_jaar* equals 'no'. The coefficient of -1.15 means that if someone has not worked the past year then the log hazard ratio decreases by 1.15. A negative coefficient indicates a positive effect on the expected survival time and thus a negative estimated effect on the hazard.

The concordance index of the model is 0.811, when the model is trained on the complete dataset and 0.791 on average with 100 bootstrap samples.

When using the Cox proportional hazards model, we should first check if the proportional hazards assumption holds. This can be tested using a statistical test. The output of this test can be found in Appendix C.2.

As we can see the global p-value is lower than 0.05, which indicates a likely violation of the proportional hazard. Both part-time income and the number of children have a p-value below 0.05, however simply removing those covariates can make the model perform worse.

In Figure 5.4 the value of the coefficients of part-time income and children is shown over time. It seems that the influence of part-time income on the log hazard hazard ratio increases in the first year and the influence of children decreases slightly.

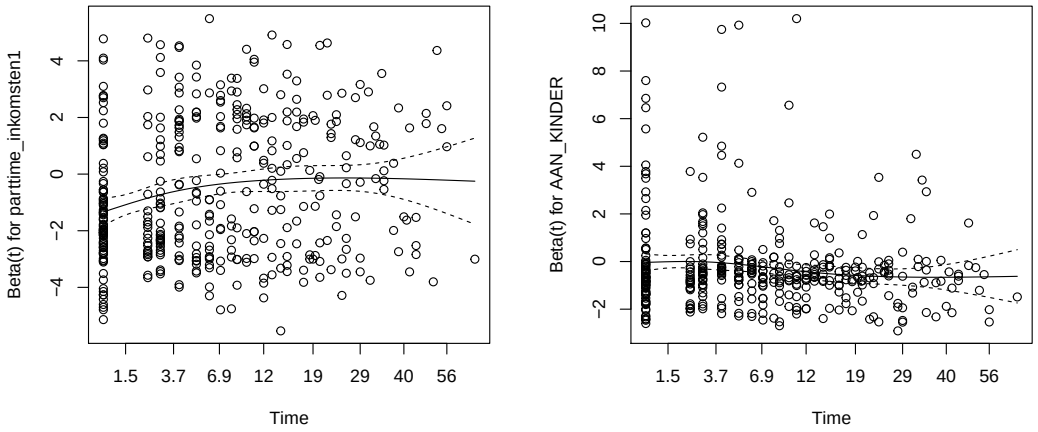


Figure 5.4: Coefficients of part-time income (left) and children (right) plotted over time. If the plotted curve is completely horizontal, then the effect of the covariate is independent over time.

Note that a violation of the proportional hazards does not make the model useless, but the underlying statistical model will most likely not hold.

5.3.2 Parametric survival models

Since the proportional hazards assumption is probably violated we also look at parametric survival models. In this case we consider the exponential, Weibull and log-logistic models. The output of the log-logistic model can be found in Appendix C.3.

The concordance index for the models trained on the entire dataset can be found in Table 5.1.

Model	**Apparent**	**Bootstrapped**
Exponential	0.810	0.789
Weibull	0.811	0.792
Log-logistic	0.813	0.792

Table 5.1: Concordance indices for the parametric survival models trained on the entire dataset (apparent) and average of 100 bootstrap samples.

As we can see the concordance index is practically the same for all the different methods. The scores for the Cox proportional hazards model are also comparable to these scores.

Also the coefficients and the significance of the covariates are practically the same as the output from the Cox proportional hazards model. Note that the sign of the coefficients from the parametric models is opposite to the Cox proportional hazards model.

5.3.3 Survival trees

As is the case with the normal classification trees, we can make many different survival trees depending on different filters and outflow. However one of the most interesting survival trees is the one for outflow to work (see Figure 5.5). This analysis can also be very useful for visualizing the effect of the covariates, since it combines insights from the Kaplan-Meier curves and classification trees.

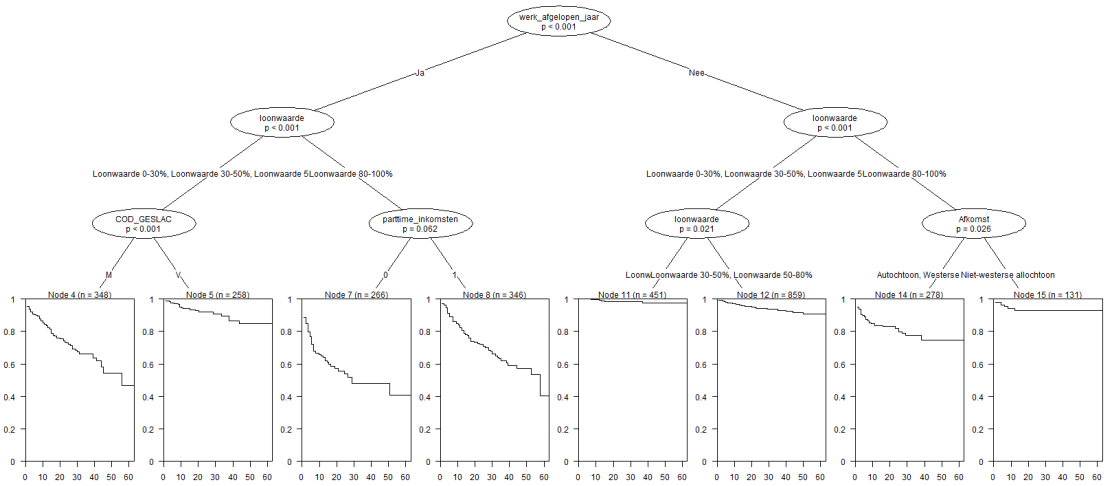


Figure 5.5: Plot of a survival tree for outflow to work. At every node the p-value of the split is shown and the splitting criterion. At the leaf nodes the Kaplan-Meier estimators are shown for every group.

In Figure 5.5 the survival tree is displayed. We can see that *Werk_afgelopen_jaar* is the most important covariate, with a p-value < 0.001. The group with lowest survival rate is the third leaf node, which has almost four times as many events as the total group. This group has worked last year, has the highest *Loonwaarde* and no part-time income. This is an interesting result since part-time income is a positive feature overall, however for the group that has worked last year and has the highest *Loonwaarde* it is a negative feature. Note that the third and fourth leaf node have similar survival ratios at the end of the study period, however the third leaf node has a steeper survival curve indicating a shorter survival time.

The group with the highest survival rate is fifth leaf node, which consists of clients that have the lowest *Loonwaarde* and who have not worked last year. Almost no one in this group has found a job in the study period.

5.3.4 Survival random forest

While a survival tree is an interesting model to visualize survival data, it has its drawbacks, such as over-fitting. A random forest mostly outperforms classification trees, so instead of using a

survival tree, we could also use a survival forest to model the survival data. The error rate and importance of the covariates can be found in Figure 5.6.

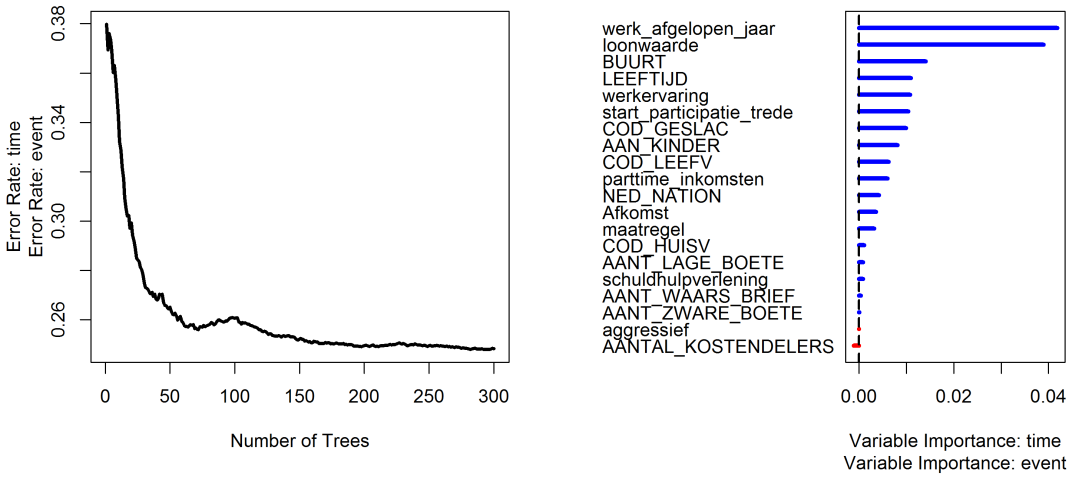


Figure 5.6: The left plot shows the out-of-bag (OOB) error rate depending on the number of trees used. The right plot gives the variable importance of all the covariates.

To evaluate model we again use the c-index and compare the survival forest to the Cox proportional hazards model (see Figure 5.7). The c-index is again calculated using bootstrapping with 100 samples.

As we can see, the Cox proportional hazards model performs slightly better than the survival forest at all times, even though the proportional hazards assumption will likely not hold. The random forest was built using 200 trees.

5.4 Analysis on intervention programs

Up until now we have looked at the effect of client features on the outflow, but not at the effect of the intervention instruments. The classification tree tool can filter on intervention instruments, but that will not tell us how the different programs compare to each other.

If we want to look at the effect of the intervention instruments we ideally want a treatment group of clients who have only participated in that kind of treatment and a control group who has not participated at all. Unfortunately it is almost impossible to find a treatment group that only has had one treatment and since everyone receives some intervention instrument there also is no such control group.

The next best thing we can do is compare a group that has received among other treatments one particular intervention instrument and a group that has received other treatments but not this particular intervention instrument. In this case we compare the effect of an intervention instrument against all other intervention instruments.

This analysis requires two steps; First we must make sure that the treatment and control group differ in their treatment, while the rest of the covariates are equal (as much as possible). Then we can use survival analysis to plot the differences using Kaplan-Meier curves between the two groups and check the log-rank test, to see if the curves differ significantly.

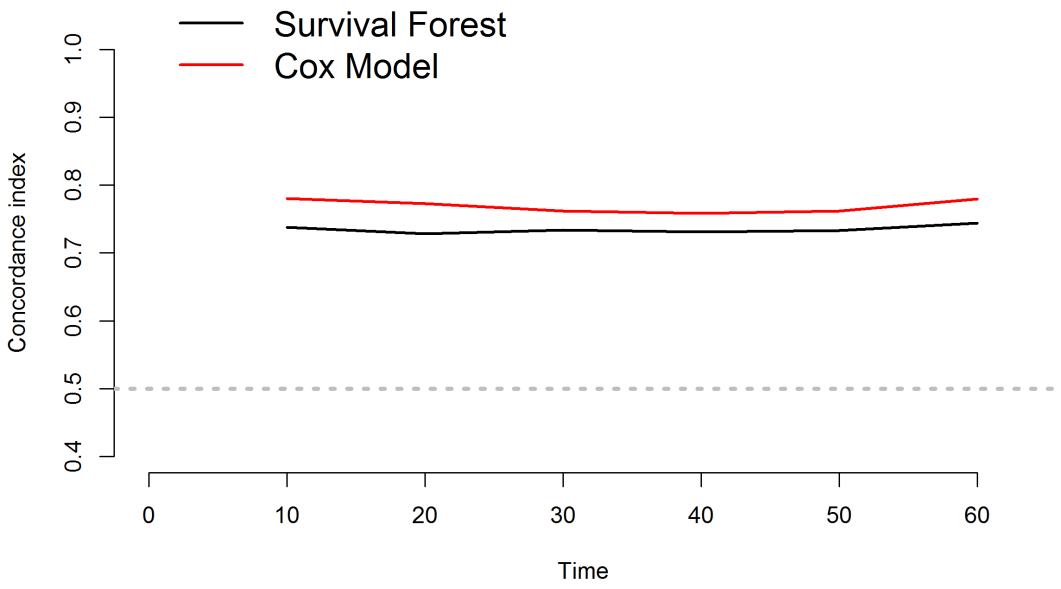


Figure 5.7: The c-index between the Cox proportional hazards model and the survival forest. At each timepoint the c-index is calculated using only those pairs where one of the event times is known to be earlier than this timepoint.

5.4.1 Matching

For creating the control group we use propensity score matching, genetic matching, Euclidean distance matching and Mahalanobis distance matching. To evaluate the effect of the matching we use the Standardized Mean Difference (SMD), which in this case is calculated as follows:

$$\text{SMD}_i = 100\% \times \frac{\text{mean}(T_i) - \text{mean}(C_i)}{\text{sd}(T_i)}, \tag{5.1}$$

where T_i indicates the i'th covariate of the treatment group and C_i the i'th covariate of the control group. To evaluate the result of the matching we can look at the difference of the SMD before and after the matching. See Tables 5.2, 5.3, 5.4 and 5.5 for an overview of the average SMD over all covariates for the intervention instruments *Intervention A*, *Intervention B*, *Intervention C* and *Intervention D*.

As we can see Mahalanobis distance and PSM usually work best on reducing the overall SMD. However different matching methods may differ in their SMD for important covariates. So the overall SMD does not say everything and we should use multiple matching techniques.

5.4.2 Survival analysis

Now that we have a matched control group, for every intervention instrument we can use the Kaplan-Meier estimator to plot the survival function for the treatment and control group.

	Before			After		
Matching	**mean**	**min SE**	**max SE**	**mean**	**min SE**	**max SE**
PSM	10.225%	8.066%	12.385%	4.149%	3.480%	4.818%
Gen. Match	10.225%	8.066%	12.385%	6.776%	5.542%	8.010%
Norm. Match	10.225%	8.066%	12.385%	5.269%	3.994%	6.544%
Mahalanobis	10.225%	8.066%	12.385%	4.158%	3.322%	4.994%

Table 5.2: Standardized mean difference for *Intervention A* over all covariates, before and after the matching and the standard error on both sides.

	Before			After		
Matching	**mean**	**min SE**	**max SE**	**mean**	**min SE**	**max SE**
PSM	21.025%	17.385%	24.664%	3.945%	3.380%	4.509%
Gen. Match	21.025%	17.385%	24.664%	9.522%	7.961%	11.083%
Norm. Match	21.025%	17.385%	24.664%	5.729%	4.443%	7.015%
Mahalanobis	21.025%	17.385%	24.664%	3.536%	2.963%	4.109%

Table 5.3: Standardized mean difference for *Intervention B* over all covariates, before and after the matching and the standard error on both sides.

	Before			After		
Matching	**mean**	**min SE**	**max SE**	**mean**	**min SE**	**max SE**
PSM	19.025%	15.846%	22.205%	2.528%	2.165%	2.891%
Gen. Match	19.025%	15.846%	22.205%	7.727%	6.467%	8.988%
Norm. Match	19.025%	15.846%	22.205%	5.399%	4.005%	6.794%
Mahalanobis	19.025%	15.846%	22.205%	3.574%	2.812%	4.335%

Table 5.4: Standardized mean difference for *Intervention C* over all covariates, before and after the matching and the standard error on both sides.

	Before			After		
Matching	**mean**	**min SE**	**max SE**	**mean**	**min SE**	**max SE**
PSM	14.417%	12.447%	16.387%	8.500%	7.466%	9.534%
Gen. Match	14.417%	12.447%	16.387%	6.945%	5.752%	8.138%
Norm. Match	14.417%	12.447%	16.387%	5.873%	4.433%	7.312%
Mahalanobis	14.417%	12.447%	16.387%	5.329%	4.305%	6.354%

Table 5.5: Standardized mean difference for *Intervention D* over all covariates, before and after the matching and the standard error on both sides.

Since Mahalanobis distance and PSM score the best in their overall standard mean difference we use those two methods for the Kaplan-Meier curves. In Table 5.6 the log-rank statistics for the different Kaplan-Meier curves can be found.

As we can see only for *Intervention B* and *Intervention C* the treatment and control group have significantly different KM-curves according to most matching algorithms.

Intervention	PSM	Gen. Match	Norm. Match	Mahalanobis
Intervention A	0.611	0.274	0.416	0.974
Intervention B	0.000	0.001	0.005	0.008
Intervention C	0.094	0.000	0.011	0.380
Intervention D	0.983	0.303	0.518	0.297

Table 5.6: Log-rank statistics score on Kaplan-Meier estimators on the matched dataset, using different distance metrics.

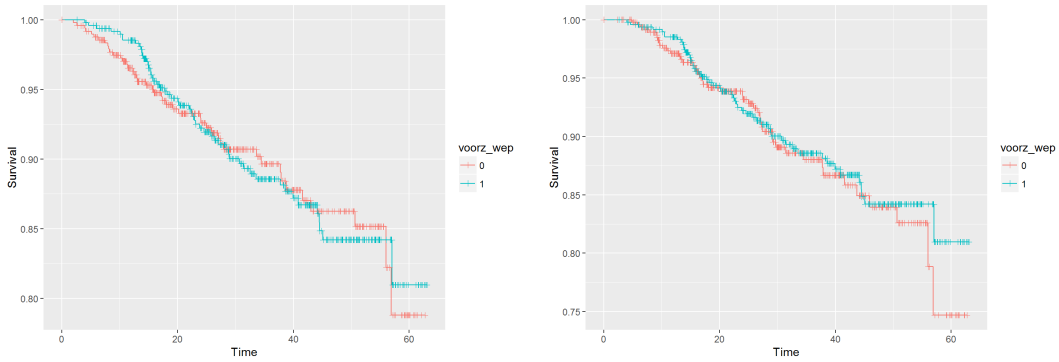


Figure 5.8: The Kaplan-Meier estimator for *Intervention A* on the matched dataset using the Mahalanobis distance matching (left) and PSM (right).

Intervention A

For the instrument *Intervention A* we can see that the curves do not differ that much, however the treatment group seems to have a slower start but in the end it catches up to the control group (see Figure 5.8). This might be due to the nature of the intervention instrument, which requires participants to work at a certain place to get experience. During that phase they cannot easily get another job. However if we look at the log-rank statistic, the curves do not differ significantly, so this intervention instrument does not work better than other instruments for these clients.

Intervention B

For the instrument *Intervention B* we can see a clear difference in the curves (see Figure 5.9). Not only does the curve for the treatment group lie completely beneath the control curve, the curves also move further apart, indicating an increasing effect. As we have seen in the previous section the difference before matching is also biggest in this program. This is because this program is intended for clients that have a bigger chance of getting a full-time job. To do this, this intervention instrument actively assists clients to get a job. As we can see in the log-rank score these curves also differ significantly according to all matching methods.

Intervention C

The Kaplan-Meier curves for *Intervention C* look similar to those of *Intervention B* (see Figure 5.10). This might be because the target audience for this intervention is comparable to that of *Intervention B*. According to the genetic and normalized matching the Kaplan-Meier curves differ significantly, but according to the Mahalanobis distance they do not differ significantly. In

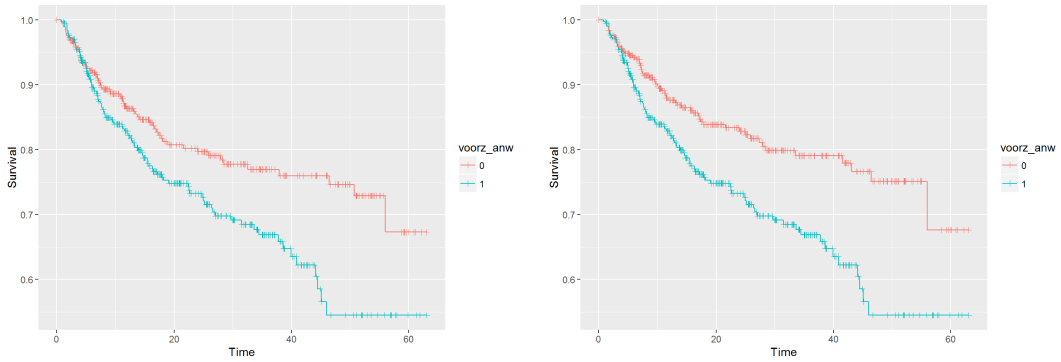


Figure 5.9: The Kaplan-Meier estimator for *Intervention B* on the matched dataset using the Mahalanobis distance matching (left) and PSM (right).

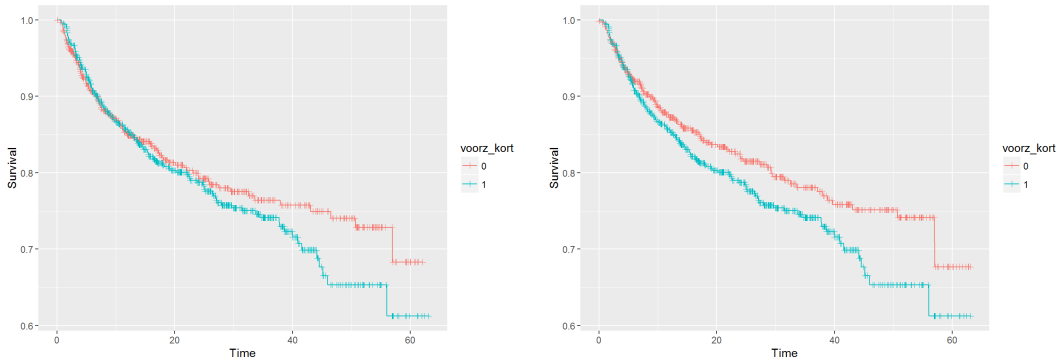


Figure 5.10: The Kaplan-Meier estimator for *Intervention C* on the matched dataset using the Mahalanobis distance matching (left) and PSM (right).

the plot from the Mahalanobis distance both curves are almost the same in the beginning, they only begin to differ after 3-4 years. This might indicate the effect is largest for clients that have been in the social security for a longer time.

However PSM gives the best match and in the right plot we can see a slight difference between the curves. The p-value of the log-rank score is in this case 0.094 which is significant at $\alpha = 0.1$.

Intervention D

For the last intervention *Intervention D*, the Kaplan-Meier curves are a lot harder to interpret (see Figure 5.11). This is because the number of treatments in this group is a lot lower en the different training programs that make up this intervention might differ. According to the log-rank score the curves do not differ significantly, which means that this intervention might not work better than other intervention instruments for these clients, however it is also probably not worse than other intervention instruments for these clients.

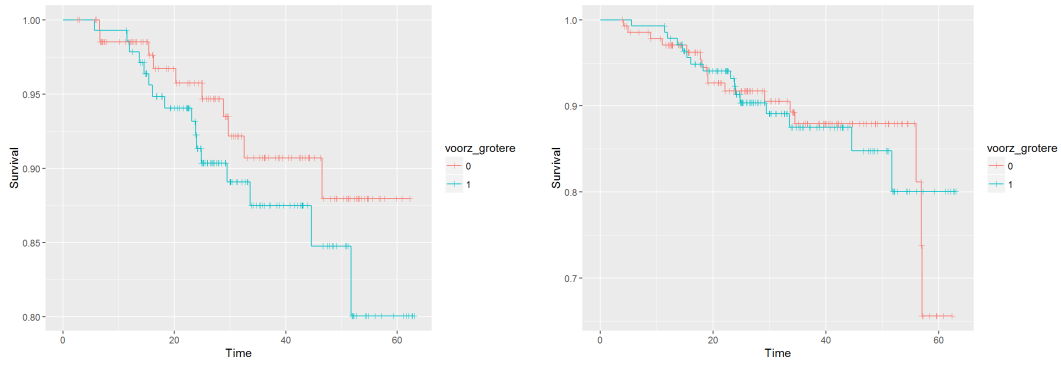


Figure 5.11: The Kaplan-Meier estimator for *Intervention D* on the matched dataset using the Mahalanobis distance matching (left) and PSM (right).

Chapter 6

Conclusion

From the analysis follows that there are some personal features that influence the chance of outflow given the current intervention assignment policy, such as *Loonwaarde*, *Participatie_trede*, *Werk_afgelopen_jaar*, age, gender and part-time income. The importance of this covariates can be found in both the classification trees and the survival analysis. The classification trees however also suggest that the number of children and cost divisors are important features. Using these analyses it is possible to predict if an individual has a high chance of outflow. These models tend to work well, but will probably not be used in the near future, since they require a standardized central dataset that can be easily updated.

Using survival analysis models we can better model the time it takes until someone exits the social security. Even though the proportional hazards assumption will probably not hold for some of the covariates, Cox proportional hazard model still performs well and can be used to find the importance and significance of the different covariates, while survival trees can be used to partition the space on log-rank score to create interesting new insights.

Finally using matching and survival analysis we found that both *Intervention B* and *Intervention C* worked better than other intervention instruments for clients that typically received those interventions, while *Intervention A* and *Intervention D* did not perform significantly better or worse than other interventions. This might indicate that it is easier to help clients that are more likely to find a job in the first place and that helping them further improves their speed of exiting the social security.

In Section 1.1 we discussed the sub-questions:

1. Can client features be used to predict the outflow from social security?
2. Can client features be used to predict the outflow from social security for clients that have participated in a specific intervention?
3. Which intervention(group) has a higher outflow rate than other intervention(group)s?

Both the classification trees and survival analysis models can be used to answer the first sub-question. It appears that client features can be used to predict both the survival time in the social security and if someone exits because of a specific reason.

To answer the second sub-question we can again look at the Shiny tools. In both tools we can filter on the intervention instrument to look at the influence of client features on the outflow within an intervention group. Furthermore the classification tree tool can also give for every leaf-node a list with the top 6 interventions followed by the clients in that leaf node.

Finally to answer the third sub-question we used an approach that consists of a matching phase and Kaplan-Meier curves.

Which brings us to the main research question: 'How can we use Data Mining techniques on social security data to improve the efficiency of reintegration?'

It seems there are a lot of different techniques useful in the domain of social security, where survival analysis is a specifically good technique.

6.1 Discussion

The municipality where this project was executed was enthusiastic about the results. Currently they are still discussing what to do with the insights gained from this research. One of the results will probably be the deployment of the Shiny tools on a laptop. These tools can then be used to look for additional patterns and features in the dataset that we have not found yet. The tools will most likely be used by the statistics department of the municipality.

Furthermore a presentation about the results was also given to the management team that handles the social security. The reactions were very positive on the results. Still it can be difficult to apply the results to their policy, since it requires a different way of handling data.

Even though we are very careful to precisely formulate the assumptions made during the matching, there can still be a problem with the quality of the matching, either because we have too little data or because the balance is too low. Interpreting the quality of the balance is difficult and highly depends on the context and imbalance before the matching. In our case having no difference between the treatment and control groups is infeasible. However since the balance has been greatly increased by the matching we conclude that the matching is good enough.

Another problem related to the matching is that the treatment group also participates in other interventions. Therefore we do not know for certain that the effect is only thanks to the intervention instrument that is currently investigated.

Yet another problem is that of implementation. For this project the data was manually extracted from different databases from the municipality and many preprocessing steps were performed before the analysis could be done. However these kind of analysis are more useful when they can be done more than once. Currently this is not easily possible since the preprocessing and manual extraction are a large part of this project. Also using prediction models that can continuously predict and update its model requires a more standardized dataset for example in the form of a data warehouse.

We must always keep in mind that the results found using the different models signify correlation but not necessarily causation. Even though outflow always occurs after the features are known, there might still be hidden features that also play an important roll.

6.2 Future work

While this project was successful in presenting some ways where data science can help to improve the efficiency of reintegration there are always data science techniques which could also be useful in this field. We experimented briefly on using clustering which seems like a promising technique on finding target audiences for intervention instruments. Also sequential pattern mining might be a good technique to analyse the order of different intervention instruments that clients can follow.

Another way to look at the imbalance between the treatment and control group is to include both the intervention instrument and the covariates in a survival model such as the Cox pro-

portional hazards model. While this was briefly tried in this project, more research would be required to evaluate the usefulness of such a model.

Also it might be better to include time-varying variables in the Cox regression model, because some covariates will definitely change over time, such as age.

Another improvement would be to use multiple imputation instead of single imputation and fit the models on every imputed dataset before pooling the results together.

Finally analysing this data more than once could lead to valuable insights, especially if the municipality decides to change their policy on the basis of this analysis. Then it would be good to analyse the result of such a policy change.

Chapter 7

Bibliography

[1] Paul D. Allison. *Multiple regression: A primer*. Pine Forge Press, 1999.

[2] Melissa J. Azur, Elizabeth A. Stuart, Constantine Frangakis, and Philip J. Leaf. Multiple imputation by chained equations: what is it and how does it work? *International journal of methods in psychiatric research*, 20(1):40–49, 2011.

[3] J.L. Binet, A. Auquier, G. Dighiero, Cl. Chastang, H. Piguet, J. Goasguen, G. Vaugier, G. Potron, P. Colona, F. Oberling, et al. A new prognostic classification of chronic lymphocytic leukemia derived from a multivariate survival analysis. *Cancer*, 48(1):198–206, 1981.

[4] R.W.B. Blonk, M.W. van Twuijver, H.A. van de Ven, and A.M. Hazelzet. Quickscan wetenschappelijke literatuur gemeentelijke uitvoeringspraktijk. Technical report, TNO, 2015.

[5] J. Bolhaar, N. Ketel, and B. van der Klaauw. Onderzoek naar effectiviteit inzet reintegratieinstrumenten DWI. Technical report, Vrije Universiteit Amsterdam, 2014.

[6] Irmgard Borghouts, Ronald Dekker, Charissa Freese, Shirley Oomens, and Ton Wilthagen. *Het werkt niet vanzelf: Over loonprikkels als instrumenten in de Participatiewet*. Celscus juridische uitgeverij, Amersfoort, 2015.

[7] Imad Bou-Hamad, Denis Larocque, Hatem Ben-Ameur, et al. A review of survival trees. *Statistics Surveys*, 5:44–71, 2011.

[8] Stef Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in R. *Journal of statistical software*, 45(3), 2011.

[9] Longbing Cao. Social security and social welfare data mining: An overview. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 42(6):837–853, 2012.

[10] David R. Cox. Regression models and life table. *Journal of the Royal Statistical Society*, 34:187–220, 1972.

[11] Peter Dalgaard. *Introductory statistics with R*. Springer Science & Business Media, 2008.

[12] Frank E. Harrell, Robert M. Califf, David B. Pryor, Kerry L. Lee, and Robert A. Rosati. Evaluating the yield of medical tests. *Jama*, 247(18):2543–2546, 1982.

[13] Arjan Heyma. Re-integratiedienstverlening in de ww: Wat werkt voor wie en wanneer? Technical report, SEO economisch onderzoek, 2015.

[14] Torsten Hothorn, Kurt Hornik, and Achim Zeileis. Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical statistics*, 15(3):651–674, 2006.

[15] Edward L. Kaplan and Paul Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481, 1958.

[16] Alenka Kavkler, Daniela-Emanuela Danacica, Ana Gabriela Babucea, Ivo Bicanic, Bernhard Bohm, Dragan Tevdovski, Katerina Tosevska, Darja Borsic, et al. Cox regression models for unemployment duration in Romania, Austria, Slovenia, Croatia and Macedonia. *Romanian Journal of Economic Forecasting*, 10(2):81–104, 2009.

[17] Gary King and Richard Nielsen. Why propensity scores should not be used for matching. Working paper, February 2016.

[18] David G. Kleinbaum and Mitchel Klein. *Survival analysis: a self-learning text.* Springer Science & Business Media, 2006.

[19] Prasanta Chandra Mahalanobis. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2:49–55, 1936.

[20] Nathan Mantel. Evaluation of survival data and two new rank order statistics arising in its consideration. *Cancer chemotherapy reports. Part 1*, 50(3):163–170, 1966.

[21] Seamus McGuinness, Philip J. O'Connell, and Elish Kelly. The impact of training programme type and duration on the employment chances of the unemployed in ireland. *The Economic and Social Review*, 45(3, Autumn):425–450, 2014.

[22] Bruce D. Meyer. Unemployment insurance and unemployment spells. *Econometrica*, 58(4):757–782, 1990.

[23] Paul R. Rosenbaum and Donald B. Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

[24] Joseph L. Schafer and John W. Graham. Missing data: our view of the state of the art. *Psychological methods*, 7(2):147, 2002.

[25] Colin Shearer. The CRISP-DM model: the new blueprint for data mining. *Journal of data warehousing*, 5(4):13–22, 2000.

[26] Robert H Somers. A new asymmetric measure of association for ordinal variables. *American sociological review*, pages 799–811, 1962.

[27] Terry M. Therneau. *A Package for Survival Analysis in S*, 2015. version 2.38.

[28] Laine Thomas and Eric M Reyes. Tutorial: survival estimation for cox regression models with time-varying coefficients using sas and r. *Journal of Statistical Software*, 61, 2014.

[29] Stef Van Buuren, Hendriek C. Boshuizen, Dick L. Knook, et al. Multiple imputation of missing blood pressure covariates in survival analysis. *Statistics in medicine*, 18(6):681–694, 1999.

[30] Bram van Dijk. Treatment effect of job-training programmes on unemployment duration in Slovakia. *Statistica Neerlandica*, 60(1):57–72, 2006.

[31] Wei Xu, Ziang Li, Cheng Cheng, and Tingting Zheng. Data mining for unemployment rate prediction using search engine query data. *Service Oriented Computing and Applications*, 7(1):33–42, 2013.

[32] Huaifeng Zhang, Yanchang Zhao, Longbing Cao, Chengqi Zhang, and Hans Bohlscheid. Customer activity sequence classification for debt prevention in social security. *Journal of Computer Science and Technology*, 24(6):1000–1009, 2009.

Appendix A

Shiny Tool

A.1 Classification trees

Below are some interesting classification trees that were extracted from the classification tree tool.

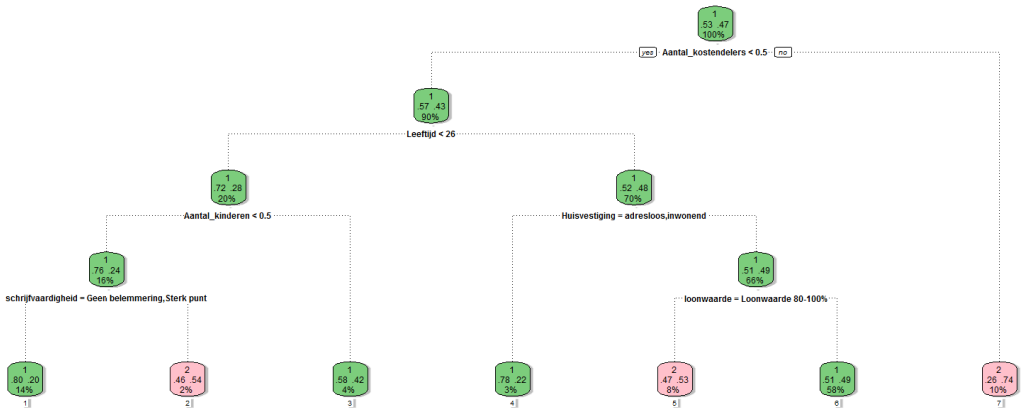


Figure A.1: Classification tree on all outflow reasons

In Figure A.1 we can see a classification tree that classifies all outflow reasons as outflow. The first split is on the number of cost-divisors. Only 26 percent of all people with cost divisors exit the social security as opposed to 57 percent of the people who do not have cost-divisors. The group with the largest probability of outflow consist of clients with no cost divisors, an age below 26, no children and no problems with writing.

Figure A.2 classifies only work, schooling and income as outflow reasons. These reasons might be seen as positive outflow reasons, since the client flows out of social security with less financial problems. We have a similar group with high probability of outflow as in the previous figure, but now there is also a small group of older clients with high *Loonwaarde* who can handle pressure and have no financial problems. This group has a high ratio of outflow but is also small.

Instead of looking at outflow within the study period (five years) Figure A.3 classifies the same outflow reasons as the previous figure but within one year. There is only one group with a high probability of outflow within one year. This group consists again of young clients with no

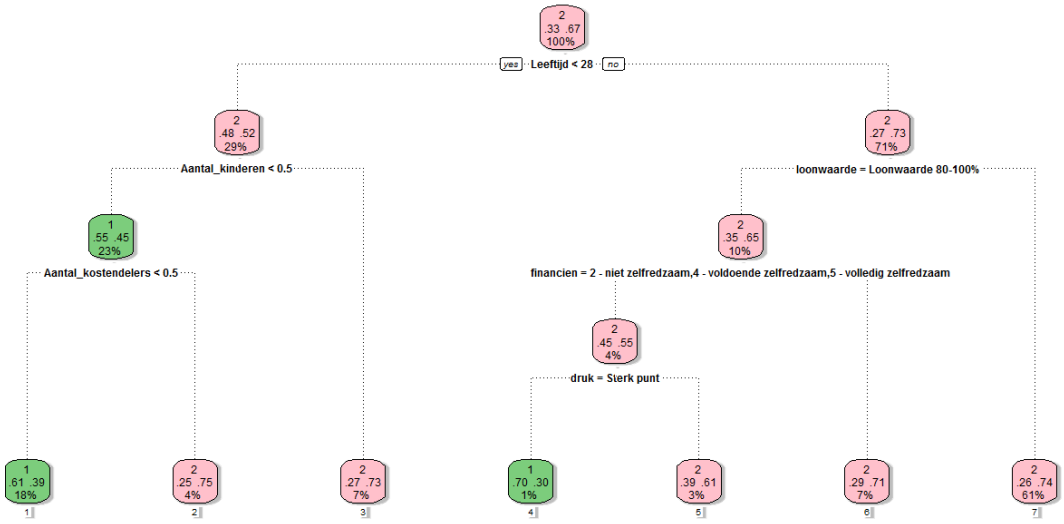


Figure A.2: Classification tree for outflow reasons: work, schooling, income

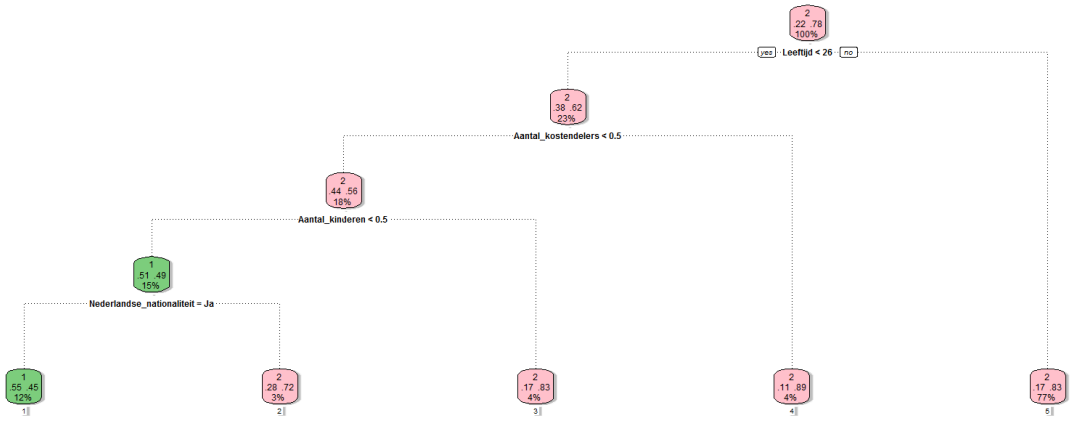


Figure A.3: Classification tree for outflow within a year

children or cost-divisors and additionally they have a Dutch nationality.

Figure A.4 classifies outflow after a year (using the same outflow reasons as the previous two trees). In this tree other covariates appear. One very important one is having part-time income. There is also a small group with a high probability of outflow, namely clients without part-time income, who have worked the past year and have either a low or high *Participatie_trede* and live in a certain neighbourhood.

Figure A.5 classifies outflow within the group of clients that have received *Intervention E*. Interestingly the most important covariate in this case seems to be if a client has a Dutch nationality. Again age, the number of children and how a client handles pressure are important covariates.

Finally Figure A.6 classifies outflow within the group of clients who receive *Intervention*

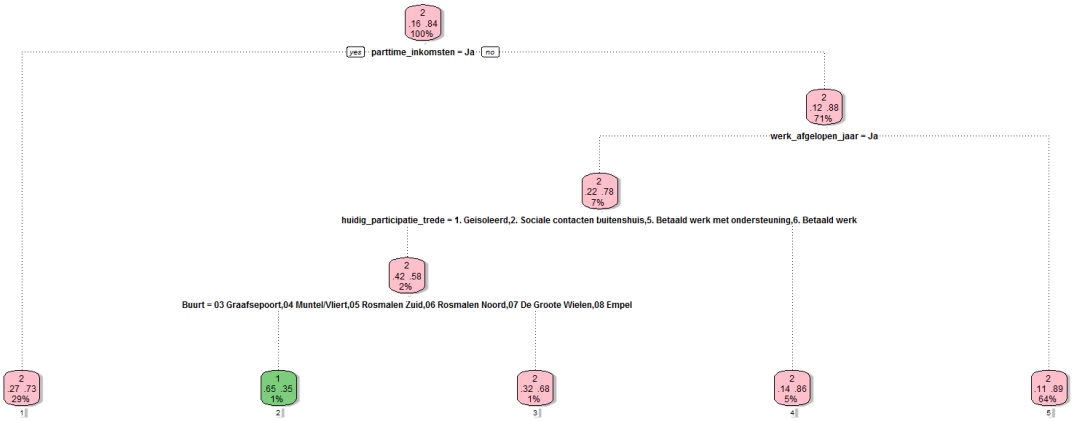


Figure A.4: Classification tree for outflow after a year

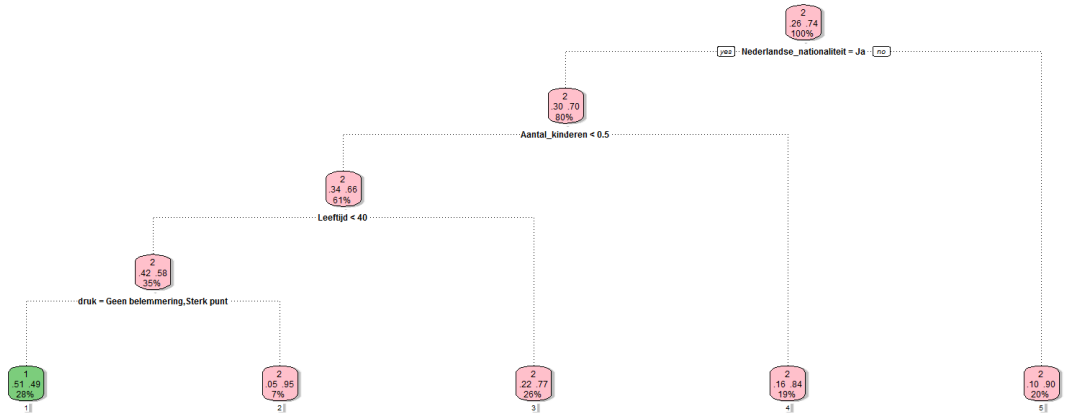


Figure A.5: Classification tree for intervention *Intervention E*

B. As we can see the probability of outflow within *Intervention B* is higher than on average. Furthermore there are more positive groups then in the previous figures. Since *Intervention B* focusses on getting clients a job as fast as possible *Werk_afgelopen_jaar* is the most important covariate.

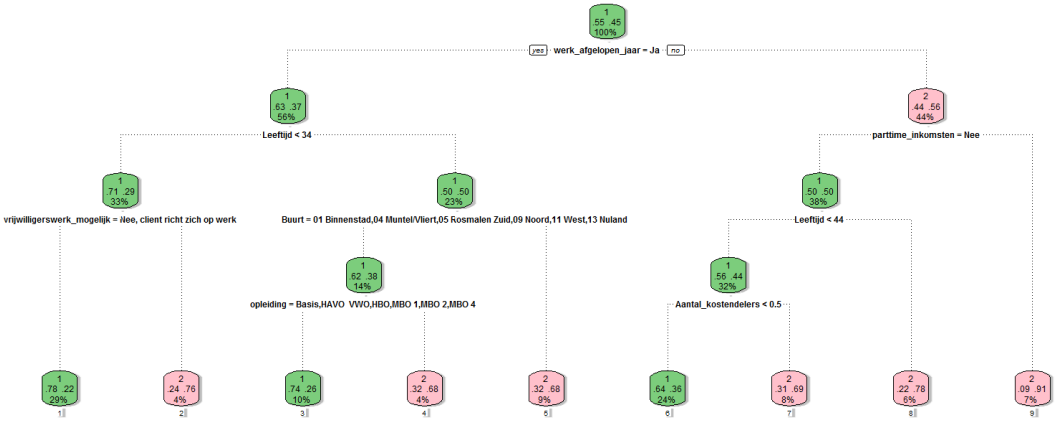
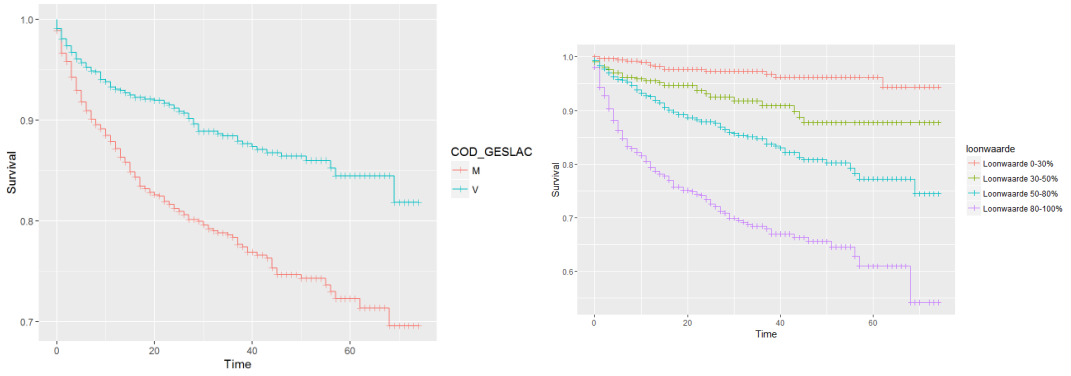


Figure A.6: Classification tree for intervention *Intervention B*

A.2 Kaplan-Meier curves

Below are some interesting Kaplan-Meier curves from the Survival Analysis tool. In all curves we only look at the outflow reason 'work'.

Figure A.7: Kaplan-Meier curves with stratified gender and Loonwaarde

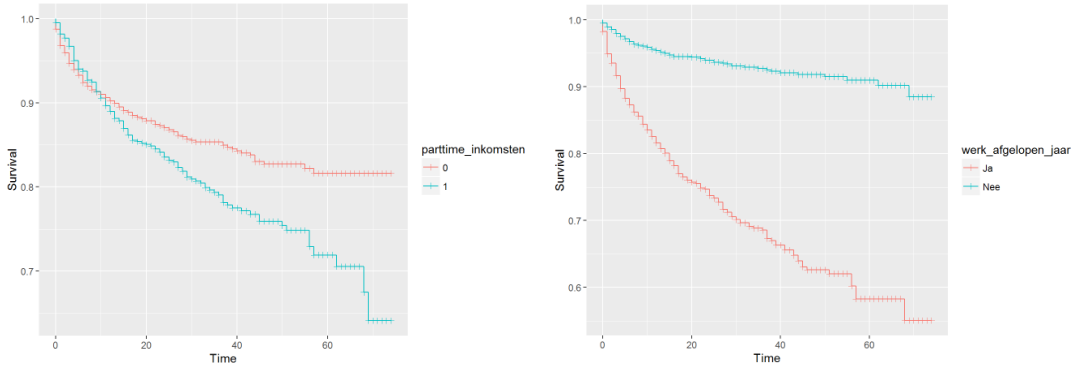


(a) Gender

(b) Loonwaarde

In Figure A.7 we can see the Kaplan-Meier curves stratified on gender (left) and *Loonwaarde* (right). As we can see a higher portion of males flow out of the social security then females. Furthermore the outflow seems to correspond with the expected *Loonwaarde*, where a high *Loonwaarde* indicates a high chance of outflow (to work). The p-value of the log-rank score is in both cases 0, indicating a significant difference between the curves.

Figure A.8: Kaplan-Meier curves with stratified part-time income and *Werk_afgelopen_jaar*



(a) Part-time income

(b) Werk_Afgelopen_Jaar

In Figure A.8 the Kaplan-Meier curves stratified on part-time income (left) and *Werk_afgelopen_jaar* (right) are displayed. The p-value of the log-rank score for is 0.01 for part-time and 0 for *Werk_afgelopen_jaar*, again indicating a significant difference. For both curves the difference in

outflow tends to get bigger over time and for *Werk_afgelopen_jaar* the difference between the two curves is exceptionally large indicating a big advantage for people with recent working experience.

Appendix B

Classification models

B.1 Performance of tree models

Below the performance is given for the other classification models that were tested. The horizontal line indicates the baseline accuracy of the majority class (no outflow).

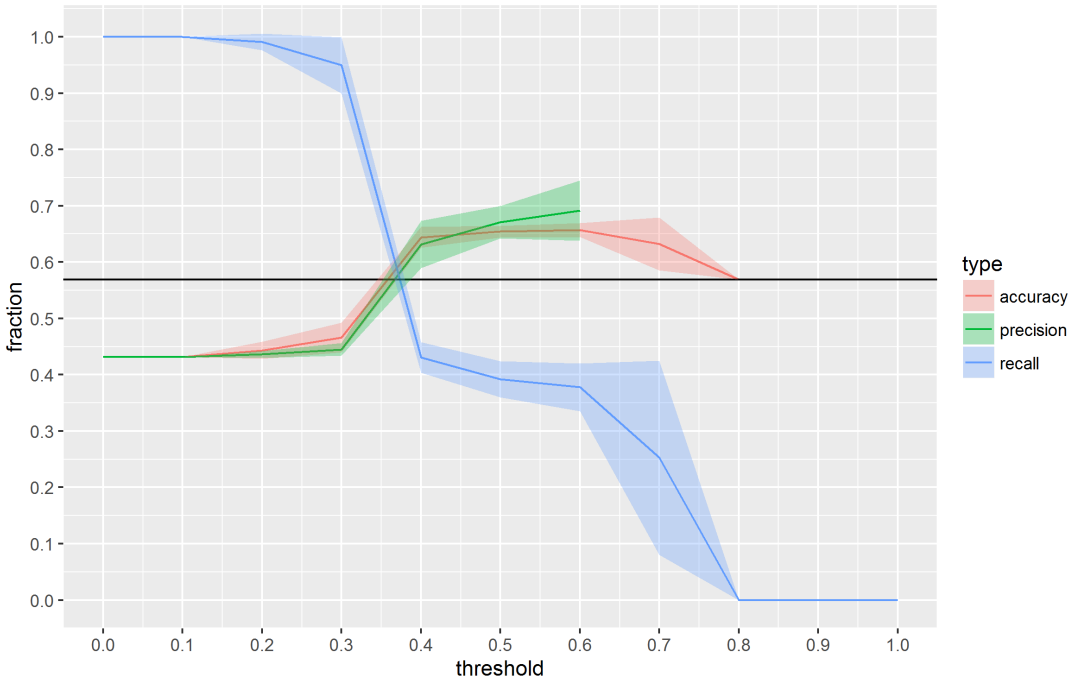


Figure B.1: Performance of the classification tree

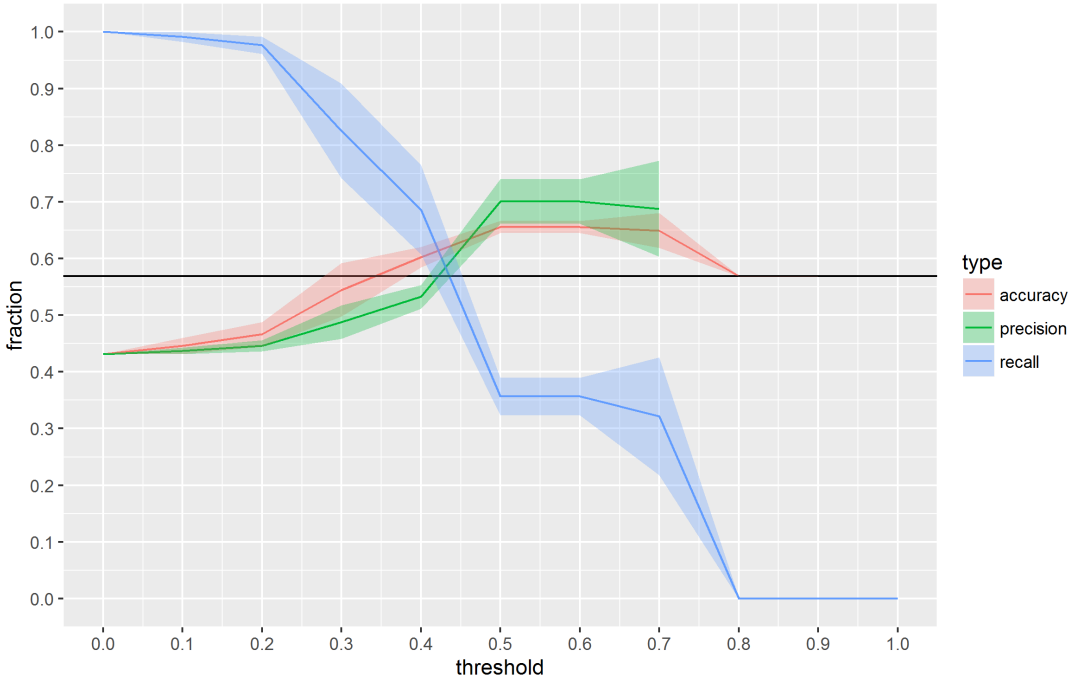


Figure B.2: Performance of the conditional inference tree

Appendix C

Survival analysis

C.1 Output Cox proportional hazards model

Below is the output of the Cox proportional hazards model.

```
Call:
coxph(formula = func, data = x)

  n= 2937, number of events= 391

                                    coef exp(coef)  se(coef)      z            Pr(>|z|)
LEEFTIJD                       -0.021292  0.978934  0.004626 -4.602       0.000004177450 ***
NED_NATION1                     0.314563  1.369660  0.214615  1.466             0.142729
AfkomstNiet-westerse allochtoon -0.403875  0.667728  0.154352 -2.617             0.008882 **
AfkomstWesterse allochtoon      0.349842  1.418843  0.301188  1.162             0.245422
COD_GESLACV                    -0.664408  0.514578  0.125313 -5.302       0.000000114553 ***
COD_LEEFVao                    -0.316419  0.728754  0.331024 -0.956             0.339133
COD_LEEFVge                     0.240601  1.272014  0.152700  1.576             0.115108
COD_HUISVeigenaar               0.022863  1.023126  0.565328  0.040             0.967741
COD_HUISVhuurder                0.056943  1.058596  0.465537  0.122             0.902648
COD_HUISVinrichting            -0.922717  0.397438  1.117131 -0.826             0.408821
COD_HUISVinwonend               0.287528  1.333128  0.622005  0.462             0.643895
AAN_KINDER                     -0.266533  0.766030  0.092513 -2.881             0.003964 **
BUURT02 Zuidoost                0.052570  1.053977  0.204035  0.258             0.796675
BUURT03 Graafsepoort            0.097722  1.102656  0.199777  0.489             0.624732
BUURT04 Muntel/Vliert           0.132627  1.141825  0.220581  0.601             0.547664
BUURT05 Rosmalen Zuid          -0.021025  0.979194  0.384363 -0.055             0.956376
BUURT06 Rosmalen Noord          0.356279  1.428006  0.296089  1.203             0.228868
BUURT07 De Groote Wielen        1.000439  2.719476  0.318008  3.146             0.001655 **
BUURT08 Empel                   0.177605  1.194354  0.475476  0.374             0.708753
BUURT09 Noord                  -0.166118  0.846946  0.205223 -0.809             0.418254
BUURT10 Maaspoort               0.400239  1.492181  0.231269  1.731             0.083519 .
BUURT11 West                   -0.055625  0.945894  0.181544 -0.306             0.759301
BUURT12 Engelen                -0.323508  0.723607  0.728404 -0.444             0.656947
BUURT13 Nuland                  0.363992  1.439063  0.531120  0.685             0.493136
BUURT14 Vinkel                  0.614395  1.848538  0.613312  1.002             0.316457
schuldhulpverlening1           -0.368782  0.691576  0.171396 -2.152             0.031425 *
parttime_inkomsten1            -0.606127  0.545460  0.117264 -5.169       0.000000235461 ***
AANTAL_KOSTENDELERS            -0.189736  0.827178  0.073313 -2.588             0.009652 **
aggressief1                    -0.188973  0.827809  0.719065 -0.263             0.792702
maatregel                      -0.131779  0.876535  0.050871 -2.590             0.009585 **
EERDER_UITK                     0.099394  1.104501  0.085709  1.160             0.246184
loonwaardeLoonwaarde 30-50%     0.871476  2.390435  0.352355  2.473             0.013388 *
```

```
loonwaardeLoonwaarde 50-80%      1.123851  3.076680  0.312353  3.598             0.000321 ***
loonwaardeLoonwaarde 80-100%     1.968323  7.158661  0.307642  6.398       0.000000000157 ***
werkervaringNee                 -0.702737  0.495228  0.225097 -3.122             0.001797 **
start_participatie_trede2        0.613061  1.846074  0.240956  2.544             0.010950 *
start_participatie_trede3        0.611466  1.843131  0.246894  2.477             0.013263 *
start_participatie_trede4        0.721380  2.057271  0.256956  2.807             0.004994 **
start_participatie_trede5        0.892436  2.441068  0.268354  3.326             0.000882 ***
start_participatie_trede6        0.470612  1.600973  0.476364  0.988             0.323189
werk_afgelopen_jaarNee          -1.151484  0.316167  0.129315 -8.905 < 0.0000000000000002 ***
AANT_LAGE_BOETE                 -0.035100  0.965509  0.093134 -0.377             0.706269
AANT_ZWARE_BOETE                -0.361462  0.696657  0.174820 -2.068             0.038675 *
AANT_WAARS_BRIEF                -0.346543  0.707128  0.235284 -1.473             0.140786
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Concordance= 0.811  (se = 0.016 )
Rsquare= 0.16   (max possible= 0.864 )
Likelihood ratio test= 511.6  on 44 df,   p=0
Wald test            = 392.1  on 44 df,   p=0
Score (logrank) test = 484  on 44 df,   p=0
```

C.2 PH assumption

Below is the output of the proportional hazards assumption check. A p-value < 0.05 indicates a likely violation of the proportional hazards assumption.

```
                                      rho     chisq        p
LEEFTIJD                         0.056794  1.160161 0.281432
NED_NATION1                     -0.051964  1.143363 0.284943
AfkomstNiet-westerse allochtoon -0.028630  0.349188 0.554573
AfkomstWesterse allochtoon      -0.013337  0.069531 0.792021
COD_GESLACV                      0.023044  0.220894 0.638359
COD_LEEFVao                      0.021952  0.184295 0.667708
COD_LEEFVge                     -0.022311  0.165082 0.684520
COD_HUISVeigenaar               -0.021641  0.194358 0.659314
COD_HUISVhuurder                -0.032414  0.425573 0.514169
COD_HUISVinrichting              0.052943  1.053870 0.304617
COD_HUISVinwonend                0.053545  1.165975 0.280230
AAN_KINDER                      -0.113438  5.686781 0.017093
BUURT02 Zuidoost                 0.001498  0.000883 0.976290
BUURT03 Graafsepoort            -0.000704  0.000196 0.988837
BUURT04 Muntel/Vliert            0.013838  0.076120 0.782627
BUURT05 Rosmalen Zuid            0.074543  2.175492 0.140225
BUURT06 Rosmalen Noord           0.061148  1.478422 0.224022
BUURT07 De Groote Wielen        -0.028025  0.298180 0.585026
BUURT08 Empel                   -0.008665  0.030731 0.860841
BUURT09 Noord                    0.024507  0.241407 0.623192
BUURT10 Maaspoort                0.095026  3.753470 0.052698
BUURT11 West                     0.037984  0.572549 0.449248
BUURT12 Engelen                 -0.029824  0.359889 0.548568
BUURT13 Nuland                  -0.027882  0.313895 0.575300
BUURT14 Vinkel                  -0.007861  0.026138 0.871565
schuldhulpverlening1             0.067487  1.936045 0.164099
parttime_inkomsten1              0.160574 10.931530 0.000945
AANTAL_KOSTENDELERS              0.077278  2.619224 0.105576
aggressief1                      0.021678  0.185656 0.666557
maatregel                        0.060512  1.289240 0.256188
EERDER_UITK                      0.006178  0.014095 0.905494
loonwaardeLoonwaarde 30-50%     -0.055574  1.218310 0.269693
loonwaardeLoonwaarde 50-80%     -0.006230  0.015838 0.899852
loonwaardeLoonwaarde 80-100%    -0.049586  0.991626 0.319345
```

```
werkervaringNee                  0.058394  1.286888 0.256622
start_participatie_trede2        0.045586  0.823197 0.364247
start_participatie_trede3        0.076141  2.261820 0.132598
start_participatie_trede4        0.061571  1.494804 0.221473
start_participatie_trede5        0.037805  0.575226 0.448190
start_participatie_trede6        0.077150  2.347668 0.125471
werk_afgelopen_jaarNee          -0.086019  3.184026 0.074361
AANT_LAGE_BOETE                  0.015216  0.092980 0.760423
AANT_ZWARE_BOETE                 0.062382  1.073292 0.300203
AANT_WAARS_BRIEF                 0.073896  2.301996 0.129208
GLOBAL                                 NA 69.008847 0.009384
```

C.3 Output parametric survival model

Below is the output of the log-logistic parametric survival model.

```
Parametric Survival Model: Log logistic Distribution

psm(formula = func, data = x, dist = "loglogistic")
                    Model Likelihood     Discrimination
                       Ratio Test           Indexes
Obs       2937    LR chi2     542.50    R2       0.213
Events     391    d.f.            44    Dxy      0.626
sigma 1.063044    Pr(> chi2) <0.0001    g        0.032
                                        gr       2.036

                                 Coef    S.E.   Wald Z Pr(>|Z|)
(Intercept)                       4.8266 0.7525  6.41  <0.0001
LEEFTIJD                          0.0271 0.0059  4.58  <0.0001
NED_NATION=1                     -0.3882 0.2567 -1.51  0.1304
Afkomst=Niet-westerse allochtoon  0.5361 0.1880  2.85  0.0043
Afkomst=Westerse allochtoon      -0.3919 0.3757 -1.04  0.2969
COD_GESLAC=V                      0.8518 0.1555  5.48  <0.0001
COD_LEEFV=ao                      0.3263 0.3851  0.85  0.3968
COD_LEEFV=ge                     -0.4121 0.1942 -2.12  0.0339
COD_HUISV=eigenaar               -0.0467 0.6685 -0.07  0.9442
COD_HUISV=huurder                 0.0654 0.5372  0.12  0.9030
COD_HUISV=inrichting              1.5316 1.3086  1.17  0.2418
COD_HUISV=inwonend               -0.1498 0.7441 -0.20  0.8405
AAN_KINDER                        0.3258 0.1104  2.95  0.0032
BUURT=02 Zuidoost                -0.1715 0.2573 -0.67  0.5051
BUURT=03 Graafsepoort            -0.2188 0.2503 -0.87  0.3820
BUURT=04 Muntel/Vliert           -0.2874 0.2776 -1.04  0.3004
BUURT=05 Rosmalen Zuid            0.0157 0.4775  0.03  0.9737
BUURT=06 Rosmalen Noord          -0.4921 0.3789 -1.30  0.1941
BUURT=07 De Groote Wielen        -1.3437 0.4120 -3.26  0.0011
BUURT=08 Empel                   -0.0778 0.5963 -0.13  0.8961
BUURT=09 Noord                    0.1526 0.2514  0.61  0.5439
BUURT=10 Maaspoort               -0.4779 0.2903 -1.65  0.0997
BUURT=11 West                    -0.0381 0.2252 -0.17  0.8658
BUURT=12 Engelen                  0.2497 0.8930  0.28  0.7798
BUURT=13 Nuland                  -0.4421 0.6418 -0.69  0.4909
BUURT=14 Vinkel                  -1.2103 0.8312 -1.46  0.1454
schuldhulpverlening=1             0.4587 0.2073  2.21  0.0269
parttime_inkomsten=1              0.8201 0.1464  5.60  <0.0001
AANTAL_KOSTENDELERS               0.2314 0.0875  2.64  0.0082
aggressief=1                      0.3956 0.9966  0.40  0.6914
maatregel                         0.1863 0.0631  2.95  0.0031
EERDER_UITK                      -0.1453 0.1049 -1.39  0.1660
loonwaarde=Loonwaarde 30-50%     -0.9829 0.3974 -2.47  0.0134
loonwaarde=Loonwaarde 50-80%     -1.2823 0.3511 -3.65  0.0003
```

loonwaarde=Loonwaarde 80-100%	-2.4121	0.3489	-6.91	<0.0001
werkervaring=Nee	0.7904	0.2605	3.03	0.0024
start_participatie_trede=2	-0.7279	0.2867	-2.54	0.0111
start_participatie_trede=3	-0.7513	0.2959	-2.54	0.0111
start_participatie_trede=4	-0.8607	0.3088	-2.79	0.0053
start_participatie_trede=5	-1.0852	0.3235	-3.35	0.0008
start_participatie_trede=6	-0.3492	0.5898	-0.59	0.5539
werk_afgelopen_jaar=Nee	1.4191	0.1614	8.79	<0.0001
AANT_LAGE_BOETE	0.0602	0.1116	0.54	0.5894
AANT_ZWARE_BOETE	0.4491	0.2128	2.11	0.0348
AANT_WAARS_BRIEF	0.4754	0.2867	1.66	0.0973
Log(scale)	0.0611	0.0422	1.45	0.1472