Universiteit
Leiden
The Netherlands

Opleiding Informatica

Data-driven Estimation

of Consultation Time using Regression

Jody Liu (s1526952)

Informatica & Economie

Supervisors:

Matthijs van Leeuwen & Hugo Manuel Proença

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl

31/07/2017

Data-driven Estimation of Consultation Time using Regression

Jody Liu

Abstract

Doctors are able to give the right vaccines and inform customers about their travel destinations to ensure a prepared journey. To know how much time is exactly needed for these appointments, we can use historic data to perform analyses and make predictive models. Throughout the years data have been collected of appointments and GPS data. However, this data have not been used for estimating consultation time yet. The aim of this thesis is therefore to research how regression techniques can be used to accurately estimate appointment time based on historic data.

We look at which data can be used for the estimation. Data can then be interpreted to see what effects the attributes have on time duration. Also, we look which regression models can be used to accurately model the consultation time. For this, we introduce the use of (ensemble) regression trees and model trees. Different experiments are conducted to compare the regression models to decide the most suitable model. Based on these results, the model tree comes out as the most insightful model to make time predictions.

Moreover, this research also opens up the discussion in how much should be relied on models. It shows that human factors are important influencers in models and thus raises questions to what extent models can be used for human decision making. Data cannot explain every event and data-driven predictions may conflict with future policies.

Contents

Chapter 1

Introduction

A majority of Dutch households plans at least one holiday per year; according to an annual report of Nibud [4] the number of households that went on holiday was around 67% in 2016. Compared to the last 10 years, this percentage has returned to the same level as in $2010 - 2012$ and with this growth, there is an increased need to prevent ourselves from diseases in foreign countries. With the emergence of travelling to foreign places there is a higher risk of facing a variety of existing exotic infections (e.g., Hepatitis A, Hepatitis B or dengue); facing more evolved, severe viruses such as enterovirus 71, or being confronted with new diseases (e.g., transfusion-transmitted virus or Nipah virus) [13].

Not only will vaccines protect ourselves from these viruses, they can also protect our neighbouring circle from contracting the disease. Moreover, immunization can lead to protection of the future generation. Many viruses that have caused severe disablement or higher death rates years ago, are now less acute or have completely eradicated. For example, with the global vaccination program of smallpox, the eradication of this infectious disease was officially declared in 1980 by the WHO [11]. Thus, the chances of contracting smallpox now and in the future are quite slim. The last smallpox case in the US was in 1949.

To protect people from these viruses, the company Thuisvaccinatie offers an at-home service where doctors visit customers for vaccinations. Doctors are able to give the right vaccines and provide explanations and tips regarding the customer's destination, to ensure a prepared journey. With an annual customer growth due to a growing need for disease prevention and interest in their unique service, a challenge arises where the rate of vaccination appointments is growing at a fast pace. With this growing demand for vaccinations the present scheduling tool has to be revised in order to treat every client with the available resources. For this the company has asked to perform a data analysis on appointment data, to estimate consultation time based on different features.

Hence, the problem for this thesis is how we can use data to accurately estimate consultation time by using

GPS data and appointment data. With the use of data, descriptive and predictive analyses can be performed to gain a clearer understanding of the current appointment state of the company. With a more accurate estimation, I herein hope to contribute in finding a better balance between the growing demand for vaccinations and the available doctors, where the results can be used for a better route planning.

To properly estimate consultation time, a data analysis has been carried out where regression as main statistical technique has been used to adeptly model the consultation time. The proposed research question is therefore:

How can we use regression techniques to accurately model the consultation time based on historic data?

For a clear structure of this thesis a few sub questions have been posed to answer the research question in a correct manner. The sub questions are:

1. Which data (types) can be used for estimating the consultation time?
2. Which regression technique(s) can be used to accurately estimate the consultation time?
3. Which model can be used best to showcase the estimated consultation time?
4. Which attributes strongly correlate with the consultation time?

1.1 Thesis overview

This chapter is the introduction of this thesis. Chapter 2 of this thesis consists of a section where a more extensive problem definition is given and where regression will be explained on an abstract level. It also gives an overview of the approach for this thesis. Chapter 3 discusses different regression techniques to find the appropriate technique for our own consultation time estimate. Chapter 4 focuses on the data that have been acquired for this project, with an explanation of the necessary pre-processing steps prior to the data modelling process. In the following chapter, experiments and analyses are conducted, where visualizations and models are made and compared in order to find the most suitable model for the consultation time. Chapter 6 opens up a discussion related to this thesis and ends with a conclusion in Chapter 7, with subsequently a brief look of possible future work.

Chapter 2

Problem and approach

This chapter introduces the necessary background information, and explains regression, both on a higher abstract level and with respect to the problem. It also explains the approach, and materials and tools used for this thesis. Chapter 2 aims to provide an understanding of the problem and its scope.

2.1 Background information

The company Thuisvaccinatie provides a service in vaccination support where a doctor visits a customer's house (or any other preferred place) to give the right vaccinations, instead of the customer who needs to travel to a certain physical place. Doctors are available seven days a week, where every day is divided in four different time frames between 9AM till 10PM. This gives customers the flexibility in choosing their own time frame and day according to their personal preferences.

Because of the growing number of Thuisvaccinatie customers, a present dilemma is how to serve all the customers with the currently available doctors. Every appointment takes on average one hour and is evenly split between travel time of a doctor and consultation time. This means there is an average consultation time of thirty minutes. This estimate was based on data from previous years, but is currently not sufficient enough to further build upon. A thirty minute consultation is not applicable if every appointment has their own set of features that needs to be taken into account (e.g., type of vaccination or number of customers during an appointment). Also, based on experience there is an indication that more appointments can be scheduled within one time frame than what the current method calculates.

The provided real-world data consist of two files: GPS data (positional data and time duration of the doctors' cars) and appointment data (customer data), and have been primarily used for daily operations or daily decision making. The challenge for this thesis is to use this data for data analysis purposes and to gain insights in consultation time for customers. We will develop a more accurate model of the current consultation landscape where patterns in the model can be identified and where we can conclude which parties need more consultation time and which need less.

2.2 Regression

In statistics and machine learning there are many techniques that can be used for analyzing relationships between different variables. The subfield machine learning in computer science uses data in order to develop predictive models with the ability to learn from data without explicit programming [14]. This learning from data can be done either under supervision of humans or without. Based on the data and if the target variable is known, it can be divided into supervised learning, semi-supervised learning, unsupervised learning and reinforcement learning. Choosing the right technique depends in part on the type of variables that are given and what attribute type the target variable is. Since the current problem has labelled data with response variables for every instance, supervised learning techniques can be used for this research. Supervised learning techniques can be divided into classification and regression [23], and the choice between these two techniques depends on the type of the response variable. Because there is time duration as response variable, a regression analysis is used in this case. Regression has been widely used in many different fields (e.g., predicting stock market prices or predicting house prices) when it comes down to analyzing data with quantitative target attributes and explanatory variables [2]. Its goal is to find the most accurate function that is able to fit the data with the least error between the predicted target value and the actual value [23]. Such an analysis can be used for:

1. Studying the existence of associations between variables.
2. Identifying the measure of strength of these relationships by using correlation.
3. Formulating a regression equation to predict the target value based on explanatory variables.

A dataset can be seen as a table of data consisting of rows of observations with k columns of attributes. A model with k features can be described as a multiple regression equation $E(y)$ with $x_1, x_2, ...x_k$ variables.

***Definition* 2.2.1** (Multiple regression equation)**:**

$$E(y) = f(x_1, x_2, ...x_k) = \alpha + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_k x_k. \qquad \beta_i = \text{slope} \qquad \alpha = \text{y-intercept}$$

$E(y)$ is the expected value of y. In real-world conditions it cannot be assumed that for each x_i-value the corresponding y-value consistently occurs and it can therefore be said that $E(y)$ is a probabilistic model that allows variability in y for each x_i-value. Therefore, the probabilistic model can be denoted as the mean of a conditional distribution of y. Moreover, α (alpha) determines the height of intersection with the y-axis. β_i (beta) describes the change in y when there is a one-unit change in x_i when leaving all other x_j equal [2]. The slope does not indicate a strength or weakness of the association for x_i, but determines the direction of the association with x_i.

A corresponding equation for estimating the multiple regression equation is Definition 2.2.2.

***Definition* 2.2.2** (Prediction equation)**:**

$$\hat{y} = f(x_1, x_2, ...x_k) = a + b_1x_1 + b_2x_2 + ... + b_kx_k \qquad b_i = \text{slope} \qquad a = \text{y-intercept}$$

The prediction equation provides estimates of $\hat{y}$-values as response variable for any possible value for x_i. When modelling these equations care should be taken for regression outliers. An outlier can fall far from the fitted model and can cause a different trend when adding them to the model and inducing misleading patterns.

Both the multiple regression equation and the prediction equation can be used to fit a model and to estimate a corresponding model with the available observations. A residual, $y - \hat{y}$, can be described as the difference between the observed value and the predicted value, thus measuring the prediction error. A positive residual occurs when $y > \hat{y}$ and a negative residual is present when $y < \hat{y}$. With regression one of the objectives is to fit a model with an equivalent predictive model with the least error in residual. Thus, the smaller a residual the better the prediction fit the real values. Different statistical techniques can be used for calculating the least error and will be further explained in Chapter 5.

2.3 Formalizing the problem

For this problem, the equation can be formalized as:

- $E(y)$ = time estimation for an appointment.
- $\hat{y}$ = predicted time estimation for an appointment.
- k = number of attributes present in the model that are relevant for the estimation of $\hat{y}$ and y.
- β_i or b_i= change indicator for an attribute i.
- x_i = value input for attribute i.

Possible attributes for this problem can be region, numbers of customers in one appointment, type and/ or number of vaccinations.

By using this method, not only a regression equation to predict consultation time can be made. It also helps in structuring the approach to identify which attributes associate with each other and to find the key attributes that holds the strongest impact in the consultation time. Specifically, because the relationships between the attributes with time duration are still unknown, regression trees are used to find these relationships and to classify the time duration based on different attributes by constructing decision rules.

2.4 Approach

Before starting with the regression analysis, the data need to be converted in a correct format in order to develop an accurate model [1]. The whole process from turning raw data into a meaningful model can be described in six main steps:

1. Objective and scope

   Knowing what the objective is and scope can bring focus to the research. It can function as guideline by investigating the relevant attributes and to detect irrelevant data points for the scope of the problem.

2. Data collection

   After establishing a solid understanding of the objective, the available data can be collected. The collected data will be converted into one table and can be summarized based on type of variables, number of data points and its distribution based on datatype.

3. Data cleaning

   Data cleaning improves the data quality and can ensure a more accurate data model. In this phase, irrelevant data points can be excluded and variables that are most important for the data analysis will be identified for a proper analysis. Herein lies the exploratory data analysis to get a clear understanding of the data and to visualize the available dataset.

4. Data modelling

   This phase will use the available data to correlate it with the business objectives and to make recommendations by using regression models and statistics.

5. Evaluate Accuracy

   By evaluating accuracy, models can be shaped into better models to meet the business objectives.

6. Iterate

   Note that the five phases stated above are not sequential, but form an iterative and incremental process to ensure the most accurate model when making valuable conclusions.

2.5 Materials and tools

For this thesis, we use programming language Python 2.7 with iPython Notebook. For the pre-processing part of the data, we have Pandas. Visualizations of data are done with Matplotlib and machine learning algorithms are implemented by using Sci-kit Learn. A literature study is conducted for theories about regression and statistics. The Weka software is used for visualizing a prediction model by using model trees [12].

Chapter 3

Related Work

This chapter describes different regression techniques. Specifically, three types of regression trees are explained. These techniques will be used in the data modelling process to model the data.

3.1 Regression trees

Modelling can have 1) descriptive purposes, it provides a systematic structure of the data, and 2) prediction purposes, models can predict unobserved data. Decision trees are able to do both tasks [10]. Decision trees are used within supervised learning for classifying response values based on decision rules. Classification and Regression Trees (CART) have been first introduced by Breiman in 1984. With the classic regression techniques, the relationship between the response variable and its predictors has already been specified prior to the analysis (e.g., linear or exponential) and is able to confirm whether a relationship between these two exist [20]. However, a Regression Tree Analysis (RTA) does not define such a relationship and its initial focus is on developing a set of decision rules on the predictor variables [7] [18]. These decision rules are constructed by recursively partitioning the data into smaller subsets where binary splits are made based on the predictor variables [20]. Subsequently, it uses heuristic search to evaluate all the splits and to find the best split such that the tree moves vertically deeper in the tree. As for regression trees, the best split is the one where the two derived subgroups lie closest to the corresponding response variable. This algorithm will eventually construct a tree diagram with decision rules as branches and quantitative mean responses as leaves. To prevent the tree from overfitting, we need to have enough data available and should exclude irrelevant attributes in the model.

The right graph (b) of Figure 3.1 visualizes an example of a regression tree. The regression tree algorithm partitions possible target values into different regions based on the explanatory variables and establishes different split points. We see four split points with five leaves. Based on this regression tree a target variable classifies five possible regions $[R1, R2, R3, R4, R5]$ as we can see in left graph (a).

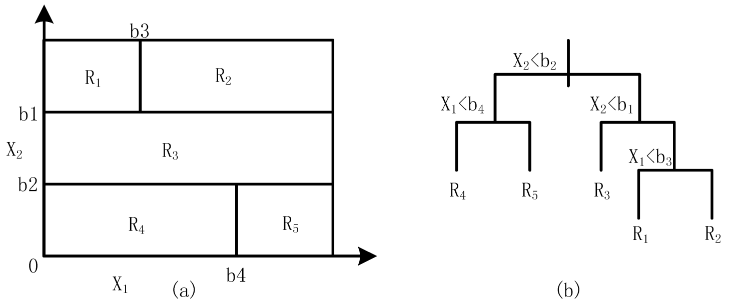


Figure 3.1: (a): A mapping of the response variable into regions $[R1, R2, R3, R4, R5]$ based on two attributes $X1, X2$. (b): Visualization of a regression tree based on attributes $X1, X2$ [24]

Most tree algorithms follow a recursive top-down method. A more formal outline of the main steps is explained below [23] [10]:

1. Start with root node r and assign all instances to r. $X := r$ expresses the traversed nodes.

2. If $X := \emptyset$, return tree with root node r and end algorithm.

3. Choose x, such that $X := X \setminus x$, to decide what the score $s^{old}(x)$ of node x is before the splitting task is performed. For regression trees this score can be the sums of squares about the mean or the sums of absolute deviations about the median.

4. If splitting is possible or necessary proceed to the next step. Otherwise return to step 2.

5. For all attributes $k \in K$, evaluate the splitting effect of that attribute. Choose attribute with highest result such that $s_k^{new}(x) - s_{old}(x)$ is maximized. This chosen attribute k can only occur once in the same path from the root.

6. Create node set of children Y. Add Y to X to make $X := X \cup Y$ and ensure that x is connected to all Y.

7. Associate the nodes in Y to the corresponding instances and go to step 2.

Regression trees can provide a clear hierarchical model that show the influence of an input variable on other input variables where variables higher in the tree can have a bigger impact on the result [19]. It is thus able to present the interactions between variables and the structure in the data. However, when dealing with linear response variables, it does not always produce the most accurate model and can have difficulties when modelling smooth lines [20] [7]. Also, trees can be highly sensitive to the data and its sample data can influence the manner in how the tree is split. It is therefore important that data are pre-processed and correct train- and test sets are made.

In the following subsections, different variants of regression trees that can be used for the modelling process are discussed.

3.1.1 Bagging trees

The word bagging is derived from *BootStrap Algorithm with Aggregation*. The idea behind bagging trees is that datasets are resampled with replacement from the original dataset [5]. With bootstrap resampling, around $1/3$ of the data is excluded from the sample and will be filled with other datapoints [20]. It then models for each sample a classifier and eventually aggregates all of these variances. When this is done, the average variance should create a lower variance compared to the variance of the original model. In theory it can be said that the variance of the aggregated classifier is reduced to 1/n of the original one.

The usage of bagging trees can lead to an aggregated model with a lower variance and can increase the accuracy of the model. A disadvantage of this method is that it requires $30 - 80$ trees to average [20]. Thus, interpreting one of these trees is difficult when all of these trees differ strongly from each other. A chosen individual tree with its relationships may be a possibility out of many, leading to a higher uncertainty of interpretation.

3.1.2 Random forest

The Random Forest algorithm collects numerous models for prediction and is used for accurate predictions that prevents the model from overfitting. The underlying idea is that an ensemble of multiple models can reduce the generalized error variance than using a single model for the prediction task [6]. Random forest uses a combination of bootstrapping and random feature selection. With the bagging method the trees are generated and at every node split a feature subset will be randomly chosen which means that the subset is independent from the previous node [18] [6]. Since a large number of trees are generated, there is a smaller generalized error which prevents the tree from overfitting and makes the features more useful for predictions [20]. Also, as with bagging trees, the trees are maximally grown without pruning and are aggregated by taking the average of the trees. Consequently, random forest can ensure a stronger prediction and also induce a durable variety among the trees [6]. With the random characteristic, the bias will be kept low because it diminishes correlations between unpruned trees, and at the same time a lower variance will occur because it ensembles the total number of unpruned trees.

3.1.3 Gradient boosted regression trees

Gradient Boosted Regression Trees are a combination of two algorithms: regression trees and boosting [10]. Boosting is a method to ensemble many models for an improved prediction. As with bagging trees and random forest, the idea is that combining multiple models together can result in a better prediction model than finding a most optimal single model. Boosting differs from other regression tree techniques because of its sequential, stage wise procedure. The boosted models fit the models to the training data by using methods to increasingly focus on the most difficult observations to predict in the existing trees. For regression problems, it can therefore be seen as a *functional gradient descent*. The algorithm starts with a regression tree that reduces the loss function maximally [10]. The second tree is built on top of that by fitting the tree on the residuals of the first tree and reduces the total error variance. The model then contains a combination of two trees where the combined residual is calculated. This process happens stage wise, meaning that the next tree will be made based on the previous model and leaves the previous model unchanged while the model becomes larger. Only the residual will be re-calculated to determine the contribution of the next new tree.

3.1.4 Model trees

The model tree algorithm was first introduced by Quinlan [21]. Quinlan's M5 algorithm constructs a decision tree with linear regression models as leaves. These linear models contain local regression coefficients of relevant attributes [17]. This local property means that attributes can be present in multiple leaves but have different independent coefficient values in each leaf. Having linear models as leaves instead of constant values, enables a more smoothing effect with less splitting occurrences. The splitting stops when the reached attribute values of a node vary little, or there are few instances remaining. According to Quinlan, the advantage of M5 over Breiman's CART algorithm is that it builds smaller regression trees with more accurate predictions [21]. Because M5 can classify many attributes for one leaf, less tree depth occurs. This makes the model less complex and provides an easier data interpretation.

3.2 Applications of regression

Research examples of the effectiveness of ensemble trees are studies in biomedical and clinical fields [3]. For many clinical analyses logistic regression is used for predicting binary outcomes; having either a disease or not. However, it would be even more beneficial if classification algorithms can be used for not only predicting these diseases but also provide insights for more targeted treatments of patients and improved assessments of a patient's diagnosis. A research with regards to this prospect indicates that there are benefits of using ensemble trees [3]. In this research, the three ensemble trees mentioned in Section 3.1 were used as well to make potential models. From results of these experiments, it came forward that logistic regression performs better than ensemble trees for predicting heart failures of patients. However, for analysis purposes ensemble trees gave good guidelines in classifying patients based on disease subtype.

A study more in line with our problem in estimating appointment times is the prediction of travel time [24] [25]. When predicting travel time a challenge is to incorporate uncontrollable factors that are influential for travel time prediction. There is sparseness of real-time traffic data and fluctuating interactions such as weather conditions, traffic incidents and different roadway circumstances. For these predictions different algorithms were used as well as the three ensemble trees mentioned in this chapter. From this research, ensemble trees were preferred over other techniques and in particular the gradient boosted trees. Since travel time deals with many fluctuating variables, the chances of having a high error rate are bigger. With gradient boosted trees it estimates many simple regression trees with low performances and by combining these together, it is able to correct these errors and improve the accuracy of the models. Moreover, because of its sequential character, it was able to reduce the error initially faster than the other algorithms.

Chapter 4

Data

This chapter describes the data available for this research and the necessary steps that were undertaken before starting with the data modelling process. As described in Chapter 2, these steps concern collecting data, cleaning data and performing an exploratory data analysis.

4.1 Data collection and data cleaning

The delivered data consisted of two files. One file contained GPS data of the doctors' cars with 14 attributes and $15,010$ entries. The other file contained data of the appointments with over 11 attributes and $7,453$ entries.

The objective is to estimate consultation time based on appointment attributes, where the appointment attributes and parts of the GPS file are explanatory variables and *stop duration* the response variable. The appointment attributes can be found in the appointment file. As for estimating the time duration of an appointment, the company does not use a clocking system of appointments and the only indication for consultation time needs to be derived from GPS data. Registered doctors use cars from the company for their appointments. The company records these travel routes, creating a data set for analysis. An available data attribute in this set is stop duration of a car and can thus indicate the duration of an appointment. If this stop duration can be linked to an appointment row then we know how long an appointment approximately was. Hence, the goal in the first step in the data analysis is to collect the two data sets and convert it into one single table with *stop duration* as response variable. To merge these two files a matching takes place where an appointment entry matches a GPS entry based on their time period, place and date. Because the data is application-driven and focuses on supporting daily operational decisions, the challenge in this project is to transform the data set into a data set for analysis purposes. This process ensures that the correct data can easily be extracted and transformed into a right format to perform data analyses. An explanation of the main attributes are listed in Table 4.1. The blue coloured attributes are key attributes for the matching process.

Attribute name	**Description**
GPS data file	
Ys code	Unique identifier
Date from	Date of car leave
Time only from	Time of car leave
Addr from	Address from car leave
Date to	Date of car arrival
Time only to	Time of car arrival
Addr to	Address to car destination
Odo from	Unknown
Odo to	Unknown
Event distance	Distance of car journey expressed in meters
Event duration	Duration of car journey expressed in seconds
Stop duration	Duration of car stop expressed in seconds
Distance private	Distance of private car use in meters
Keys in	Unknown
Appointment data file	
Reference	Unique identifier
Address	Address of customer and travel destination of doctor
Zipcode	Zipcode of customer and travel destination of doctor
Place	Place of customer and travel destination of doctor
Region	Region of customer and travel destination of doctor
Doctor	The assigned doctor for appointment
Date	Date of appointment
Time	Time indication of when appointment took place
Repeat	Indicator whether it is a first appointment or a repeat
Number of people	Number of people present during appointment
Vaccination name	An irregular set of columns indicating the vaccinations/ treatments needed per person for an appointment

Table 4.1: An explanation of the initial available data prior to pre-processing

Due to missing values and ubiquitous data entries, the files first need to be cleaned separately before merging them. This process is described in the next subsections.

4.1.1 Appointment data

Thuisvaccinatie offers 52 types of products and services to their customers. Upon receiving the data, the entries had an irregular number of columns because every appointment had a diverse number of products or services. To even the total columns out for every entry, we add the total offered products to the table and calculate the frequency of each product type for each product in one appointment.

Another change is to set the time frame in a correct format for every appointment. Appointments can be either in the morning, noon, afternoon or evening. The time frame attribute is expressed as a time stamp under attribute name *time*. E.g., interpreting an appointment on 9AM should mean an appointment on that time. However, based on an evaluation session with the company it is concluded that the stated time does not give an indication of the order between appointments and is not the actual appointment time. This means that these times are merely suggestions of a certain time frame and that doctors can deviate from these suggestions during the actual events. Since the GPS data will be partly merged based on time, this can lead to a misleading match. Thus, the attribute should be transformed into a time frame rather than a time stamp to prevent confusion. These time stamps changes to: morning, noon, afternoon or evening. For example, appointments between 9AM - 1PM are classified as a morning-value for attribute *time frame*.

Lastly, adding an extra column with the *occurrence* of a matching string serves as feature to eventually match the data with GPS data. This *matching* string is: Zipcode + time frame + date.

4.1.2 GPS data

The GPS data contain missing values for some locations or stop duration. Without known location or stop duration, it is impossible to use it for the matching process or time estimation. Since no linkage can take place without these values, it is better to exclude these from the data and to remove these entries.

Also, the doctors deliver medical examinations as a service that can take up to two hours and is separate from Thuisvaccinatie. It is therefore important that these entries are left out in the matching process, to ensure that appointments are not linked to misleading stop duration.

It happens that doctors use cars for private use. These are however no official appointments. Home locations of the doctors were therefore removed and duplicate stop locations have been deleted as well. A possible reason for a duplicate stop location is that the driver has found a better parking place a few minutes later and decided to relocate himself. However, it is of importance to choose the correct entry with the right stop location and stop duration to make an accurate prediction model. The duplicate entries with same stop location can have varied differences in stop duration from 5 minutes to 45 minutes.

Lastly, we can make the same columns as with appointment data with regards to *time frame*, *occurence* and *matching string*. With the *matching* string and *occurence* feature it is possible to merge the two tables. The occurrence feature tells the frequency of a matching string, indicating how frequent an appointment or destination point occurs at the same time, date and place.

4.1.3 Merging the data

After we clean the data separately, the matching can take place. This can be done by looking at the occurrence attribute and the matching string. Different scenarios can occur and we can use for every scenario a different technique to match the entries from both files. If the occurrences from both GPS entry and appointment entry are one, then immediate matching can take place. This means that there exists only one GPS entry with same place coordination and time frame as an appointment at a certain date. However, if one of these are at least bigger than one, then a few scenarios could have happened and are mapped out in Table 4.2.

Scenario	**Description**	**Matching solution**
Appointment occurrence == 1 GPS occurrence == 1	There is one GPS stop point in the area of an appointment at the same time and date.	The two data rows can be merged together.
Appointment occurrence >= 2 GPS occurrence == 1	There were multiple appointments in the area at the same time and date.	Option 1: By using positional data, calculate the distances for each appointment with corresponding GPS data point. Choose eventually the appointment with shortest distance with GPS point.
		Option 2: More than one appointment can take place in one area. Here, divide stop duration by appointment occurrence and distribute over appointments.
Appointment occurrence == 1 GPS occurrence >= 2	There were multiple stop points in the area at the same time and date for one appointment	By using positional data, calculate the distances for each GPS data point with corresponding appointment. Choose eventually GPS point with shortest distance with appointment*.
Appointment occurrence >= 2 GPS occurrence >= 2	There were multiple stop points and appointments in the area at the same time and date.	By using positional data, calculate the distances for each appointment with corresponding GPS data point. Choose eventually the appointment with shortest distance with GPS point*.

*It can occur that two or more GPS points have the same address and matches an appointment. In that case, the GPS point with the longest stop duration has been matched to an appointment.

Table 4.2: An explanation of possible matching scenario's when linking appointments with GPS data.

After the merging process, instead of two separate tables there is now one table with all data. We remove unnecessary attributes or duplicate attributes - in relation to the appointment data file - in the GPS data, since we are only interested in the stop duration of an entry. These attributes are for example *addr from, date from* and *time only from*. When collecting the data, there were GPS data of the doctors' cars with 14 attributes and $15,010$ entries, and an appointment file with over 11 attributes and $7,453$ entries. After cleaning and merging this to one table, there is a total of $6,615$ entries and 73 columns, from $2016-05-01$ till $2017-01-26$. Tables 4.3 and 4.4 are descriptions of the data, where vaccination products are named in Dutch.

For regression and classification algorithms there is the risk that categorical values are not well processed in a model when formatting categorical values into numerical values. If we take attribute as *region* and assign for every place a unique number instead of region name, then we cannot say that value *Amsterdam* with number 10 as value is greater than *Twente* with number 5. One-hot-encoding combats this drawback by making for every possible attribute value an individual binary column. This however also leads to extra columns and results in this case to 91 columns instead of 73 for the modelling process.

Attribute name	**Domain**	**Description**
Reference	$[0, ..., 6615]$	Appointment key for every individual entry
Address	String	Streetname of appointment
Place	String	Place of appointment
Region	String	Region of appointment
Doctor	String = {a1, ..., a14}	Name of doctor of an appointment
Date	Timestamp	Date of the appointment
Repeat	$\{0, 1\}$	A first appointment {0} or a second appointment {1}
Number of people	$[1, 8]$	Number of people during an appointment
DTP NVI	$\{0, 1\}$	A product type
DTP Revaxis	$\{0, 1\}$	A product type
Hepatitis A Senior	$\{0, 1\}$	A product type
Hepatitis A Junior	$\{0, 1\}$	A product type
Hepatitis B Senior	$\{0, 1\}$	A product type
Hepatitis B Junior	$\{0, 1\}$	A product type
Hepatitis B Indicatie	$\{0, 1\}$	A product type
Hepatitis A B Senior	$\{0, 1\}$	A product type
Twinrix	$\{0, 1\}$	A product type
Hepatitis A B Junior	$\{0, 1\}$	A product type
Buiktyfus	$\{0, 1\}$	A product type
Buiktyfus Indicatie	$\{0, 1\}$	A product type
Gele Koorts	$\{0, 1\}$	A product type
Gele Koorts Indicatie	$\{0, 1\}$	A product type
FSME	$\{0, 1\}$	A product type
FSME Indicatie	$\{0, 1\}$	A product type
Meningitis	$\{0, 1\}$	A product type
Meningitis Indicatie	$\{0, 1\}$	A product type
Rabies	$\{0, 1\}$	A product type
Rabies Indicatie	$\{0, 1\}$	A product type
Encefalitis	$\{0, 1\}$	A product type
Encefalitis Indicatie	$\{0, 1\}$	A product type
BMR	$\{0, 1\}$	A product type
BMR Indicatie	$\{0, 1\}$	A product type
HIB	$\{0, 1\}$	A product type
HIB Optioneel	$\{0, 1\}$	A product type
Pneumovax	$\{0, 1\}$	A product type
Pneumokokken	$\{0, 1\}$	A product type
Gammaquin	$\{0, 1\}$	A product type
Gammaquin Indicatie	$\{0, 1\}$	A product type

Table 4.3: Part 1 of description of available data after data cleaning

4.2 Exploratory data analysis

To gain a better understanding of the data, we conduct an exploratory data analysis to visualize the data and to investigate correlations between attributes.

Table 4.5 shows distributions of three features. We observe from this table that the mean of the *number of people* is 1.865004 and that the largest group of an appointment is 8. Also, on average there are 13 vaccination products per appointment, with a maximum of 78. A more interesting feature is *stop duration*. This shows that the maximum stop duration is $7,818$ seconds and equals 2.17 hours. This falls far above the average

Attribute name	**Domain**	**Description**
Gordelroos P1	$\{0,1\}$	A product type
Gordelroos P2	$\{0,1\}$	A product type
Consult	$\{0,1\}$	A product type
Herhaalconsult	$\{0,1\}$	A product type
Recept Malaria	$\{0,1\}$	A product type
Recept Overig	$\{0,1\}$	A product type
Cholera	$\{0,1\}$	A product type
Administratiekosten	$\{0,1\}$	A product type
Vaccinatieboekje	$\{0,1\}$	A product type
Rabies Uitleg	$\{0,1\}$	A product type
Buiktyfus Uitleg	$\{0,1\}$	A product type
Dengue Uitleg	$\{0,1\}$	A product type
Schistosomiasis Uitleg	$\{0,1\}$	A product type
Tuberculose Uitleg	$\{0,1\}$	A product type
Encefalitis Uitleg	$\{0,1\}$	A product type
Hoogteziekte Uitleg	$\{0,1\}$	A product type
Duikadvies Uitleg	$\{0,1\}$	A product type
Malaria Uitleg	$\{0,1\}$	A product type
FSME Uitleg	$\{0,1\}$	A product type
Deet Lotion	$\{0,1\}$	A product type
Deet Extra	$\{0,1\}$	A product type
NoShow	$\{0,1\}$	A product type
Event Duration	$[5,...,8721]$	Travel duration of a doctor to be at the appointment
Stop Duration	$[110,...,7818]$	Stop duration of an appointment
Time_y	Timestamp	Actual time of an appointment based on GPS data
time frame_y	String = {morning, noon, afternoon, evening}	Actual time frame of an appointment based on GPS data

Table 4.4: Part 2 of description of available data after data cleaning

	Number of people	**Number of vaccinations**	**Stop duration**
Mean	1.865004	12.845654	1671.978987
Std	1.157825	10.226579	921.724399
Min	1.000000	0.000000	110.000000
25%	1.000000	6.000000	992.000000
50%	1.000000	10.000000	1467.000000
75%	2.000000	17.000000	2144.000000
Max	8.000000	78.000000	7818.000000

Table 4.5: Data distribution based on number of people, number of vaccinations and stop duration

consultation time of an appointment. However, the table also indicates a mean of $1,671.978987$ seconds ($\approx$ 27.87 minutes). This average is in line with the current consultation time calculation where on average every appointment takes thirty minutes for consultation. Though, there is a standard deviation of 921.724399 ($\approx$ 15.36 minutes), illustrating differences in appointments as well. It would therefore be interesting to find attributes that influence these deviations in the average consultation time of thirty minutes.

In the next subsections, different visualizations are given for certain attributes to describe how they correlate with the response variable. In every boxplot figure, the right most boxplot are calculations based on the total appointments.

4.2.1 Analysis based on region

Figures 4.1 and 4.2 show boxplots of the time duration based on region of the appointment.

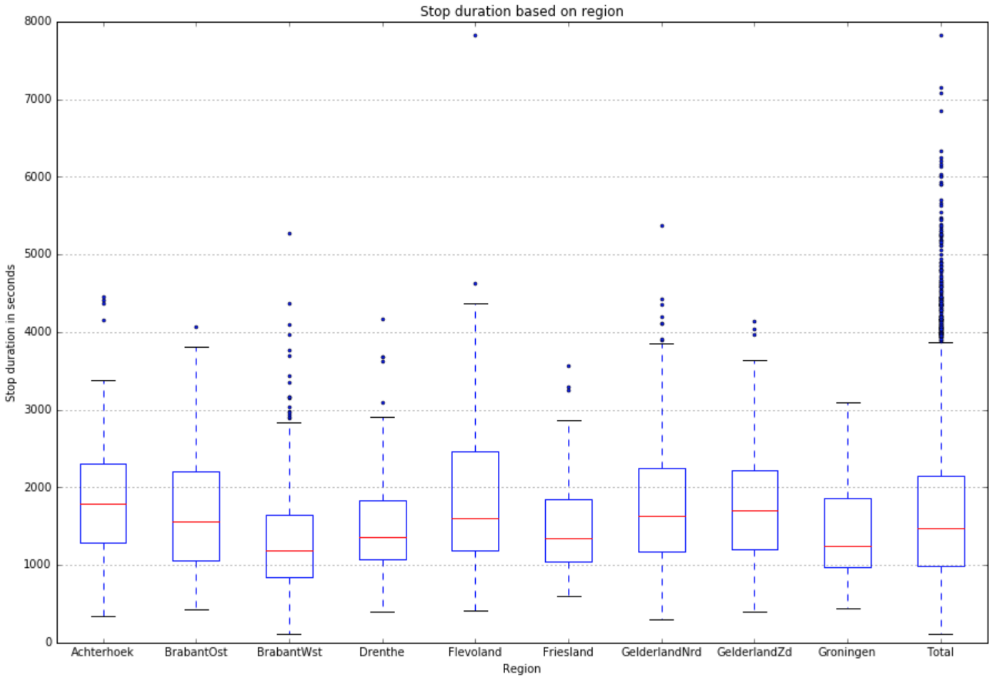


Figure 4.1: Boxplot of time duration based on region part 1 (in seconds).

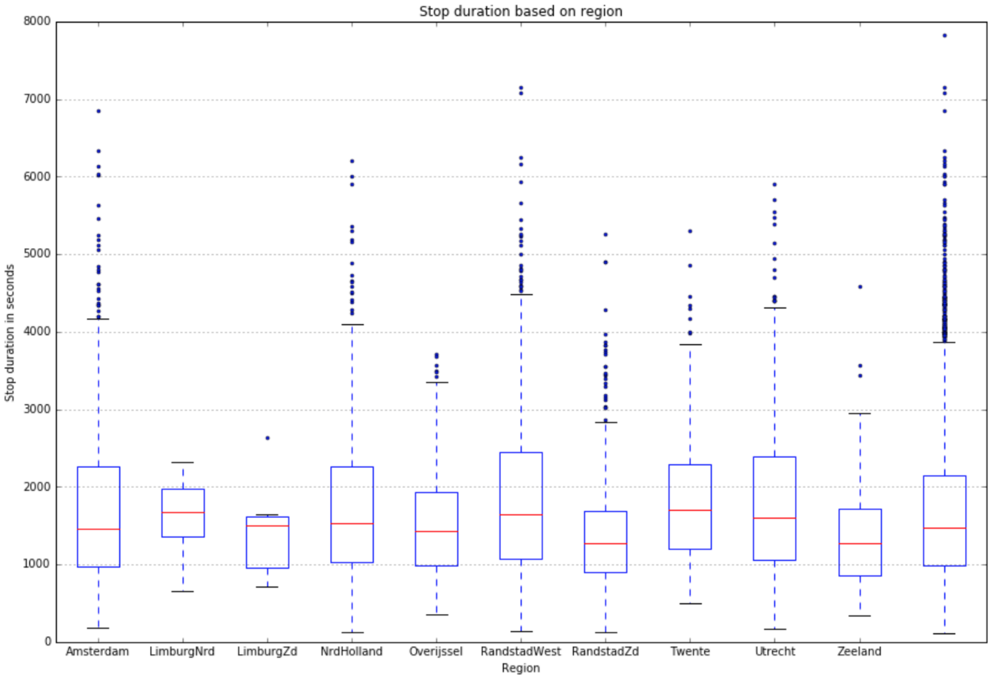


Figure 4.2: Boxplot of time duration based on region part 2 (in seconds).

It can be seen that there is an overall median between 1000 and 2000 seconds. There are no big differences compared to the regions, although the distribution of the stop durations per region can be different. For example, around Randstad West, Noord-Holland, and Amsterdam the minimum stop durations are lower compared to the other regions. However, for these regions there are also more data points with a longer stop duration. This distribution difference between sparseness and expansion can also be due to the amount of available data in these regions. Figure 4.3 shows a histogram to visualize the frequency of appointments of over $3,600$ seconds per region. This indeed shows that there are more appointments in Amsterdam, Noord-Holland and Randstad West, and can be the reason for such outliers in the boxplots.

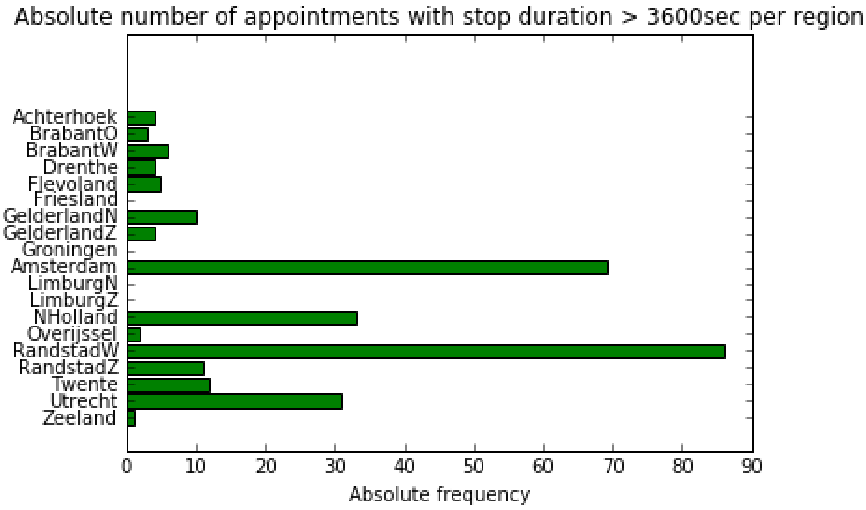


Figure 4.3: Histogram of absolute appointment frequency with stop duration > 3,600 seconds

4.2.2 Analysis based on doctor

Figure 4.4 shows the distributions of stop duration based on doctor. It should be assumed that every doctor is equally skilled and no evident differences are present in stop duration based on doctor. A boxplot helps to test if this assumption is true and that no stop duration differences are present between doctors. With Figure 4.4 we observe that there are a few differences between doctors. E.g., doctor a8 and doctor a2 have medians lying above the average doctor. Because there are differences, the doctor attribute should be included in the predictive model.

It should be noted that this does not neccessarily imply a direct causal relationship between doctor and stop duration. The longer stop duration can also be due to a larger group size that a certain doctor frequently needs to vaccinate or the type of region (e.g., in Amsterdam or Randstad West an appointment takes longer).

4.2.3 Analysis based on group size and repeat of appointment

Another boxplot visualization is Figure 4.5 where distributions of stop duration are on the y-axis with the group size on the x-axis. An upward trend is present and aligns with the assumption that the larger the group, the longer an appointment takes.

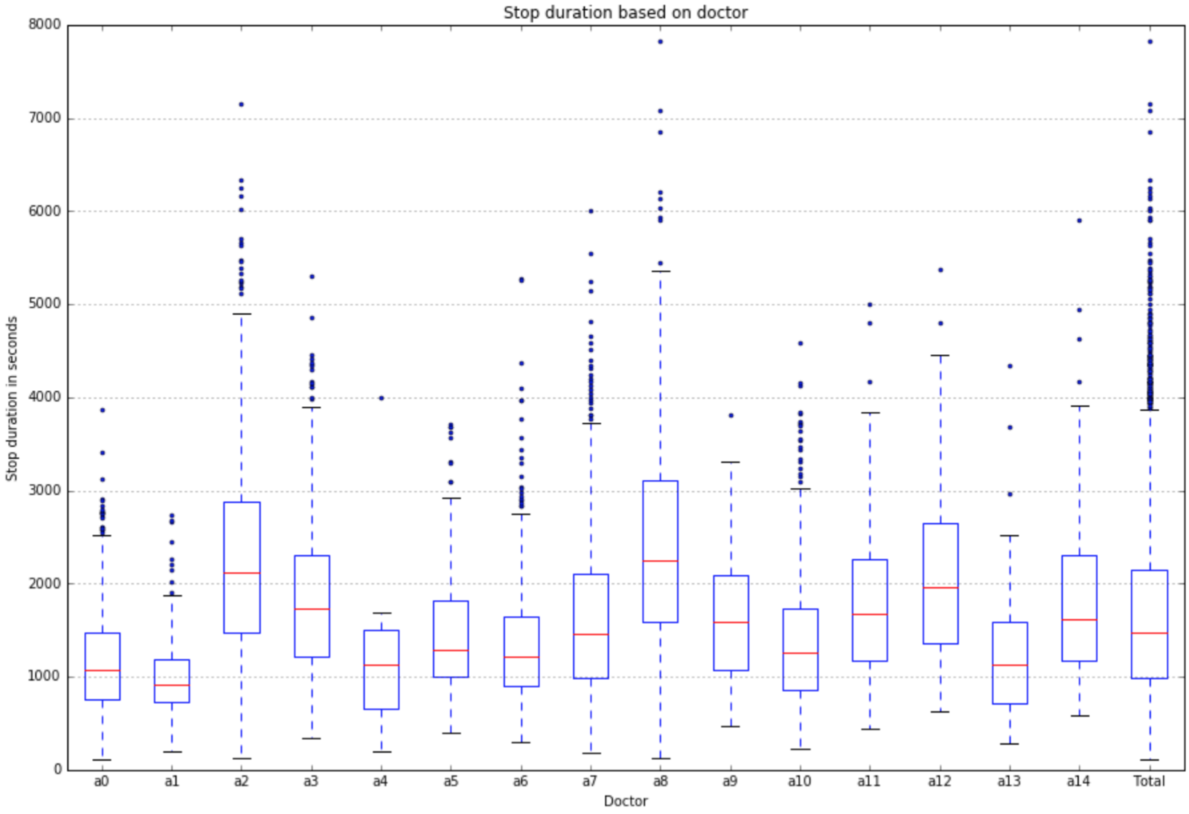


Figure 4.4: Boxplot of time duration based on doctor in seconds

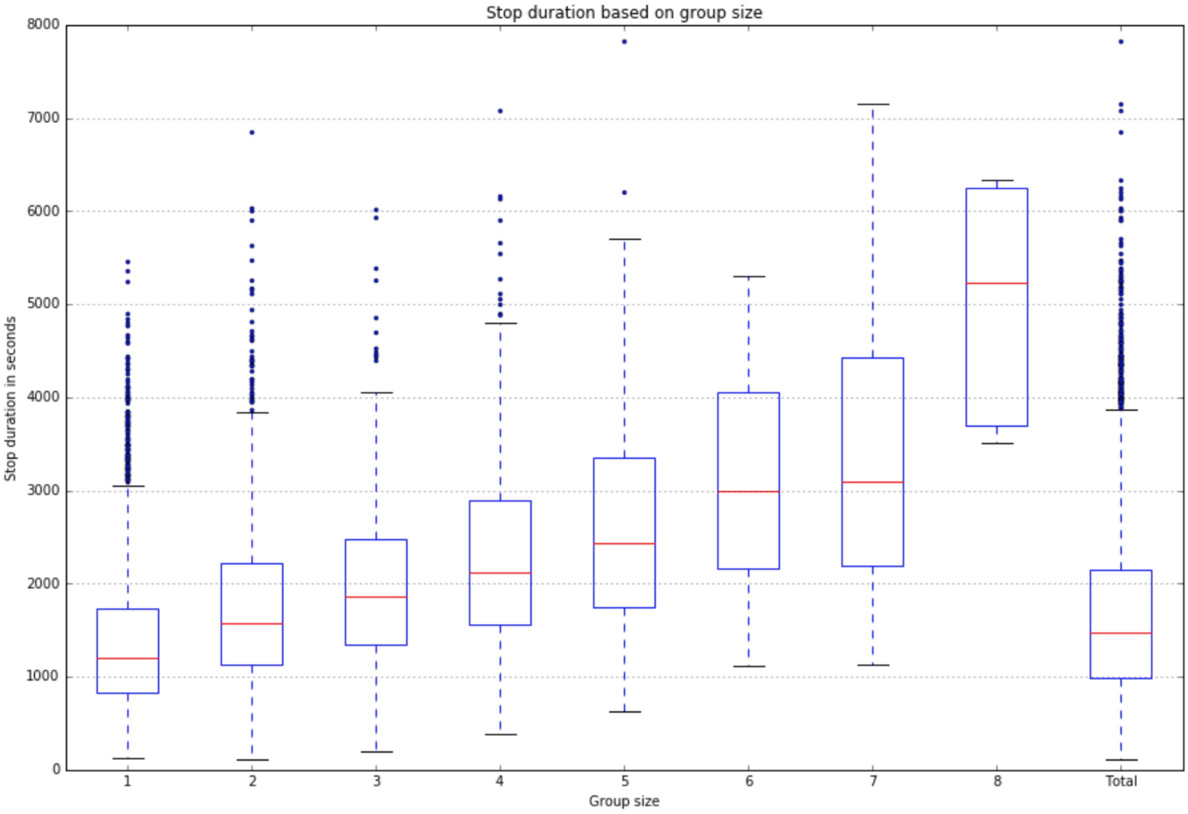


Figure 4.5: Boxplot of time duration based on group size in seconds

It is possible that a customer requires multiple sequential appointments before a vaccination yields the full effect. Figure 4.6 represents the differences in time between the two appointment types. A first appointment would take in general longer than a repeating appointment, since it covers a longer introduction from the doctor and requires a longer explanation regarding the vaccinations. Also, customers may have more questions during their first session. This notion is also reflected in the time duration of an appointment between these two types. The boxplot of the first appointment lies higher than the second appointment.

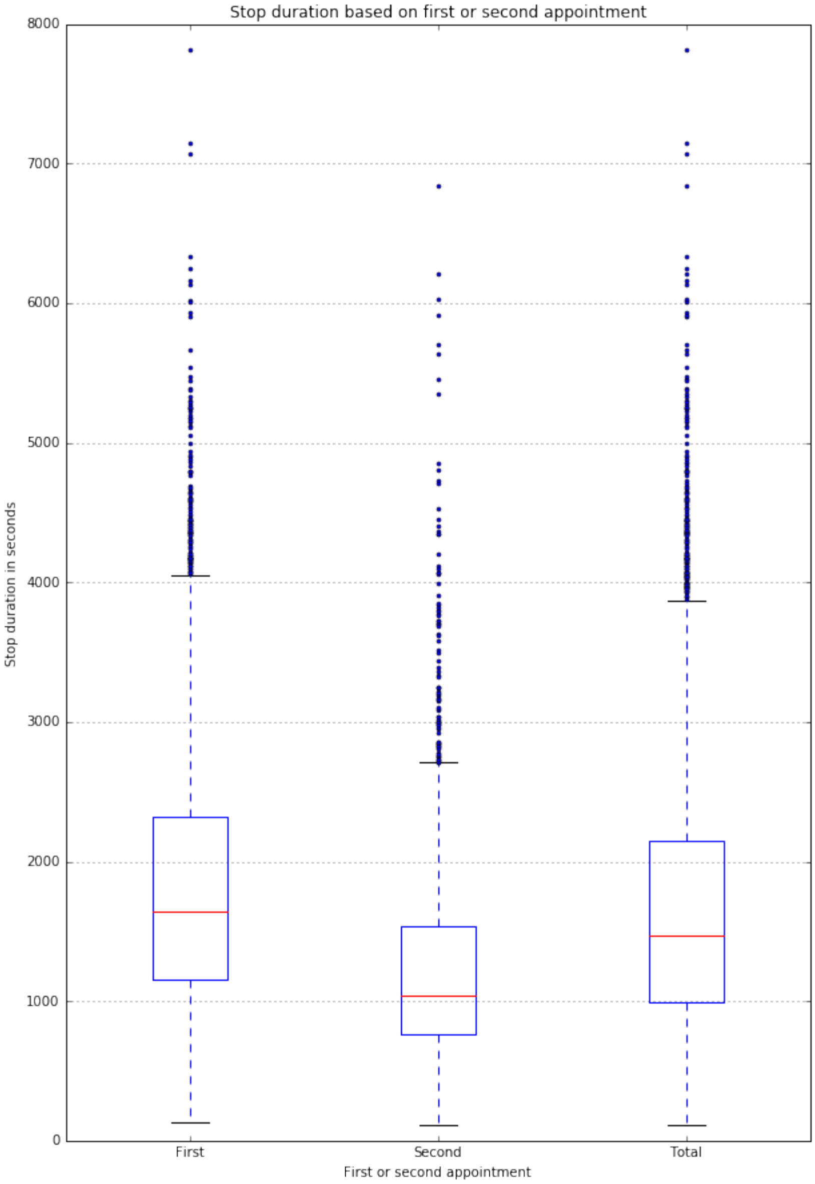


Figure 4.6: Boxplot of time duration based on first or second appointment in seconds

4.2.4 Correlations

Figure 4.7 represents a correlation matrix of different attributes. For the calculation, we use the *Pearson Correlation Coefficient* (PCC). The PCC measures the linear correlation between two continuous variables X and

Y. A commonly used Greek sympol for the PCC is ρ (rho) that divides the covariance of (X, Y) by the product of standard deviation X and standard deviation Y [2] [8]. The covariance can also be rewritten in terms of mean and expectation.

***Definition* 4.2.1** (Covariance)**:**

$$\rho(X, Y) = \frac{cov(X,Y)}{\sigma X \sigma Y} = \frac{E[(X-\mu X)(Y-\mu Y)]}{\sigma X \sigma Y}$$

A correlation is a ratio that takes a value between $+1$ and -1. $+1$ indicates a positive linear correlation, 0 indicates no linear correlation and -1 a negative linear correlation. The correlation is used to analyze how two variables are associated with each other. For example, a positive correlation means that two variables are in the same direction such that an increase of variable X means an increase of variable Y. A correlation between the same two variables for X and Y always results in 1. Table 4.6 is the same correlation matrix but expressed in numbers.

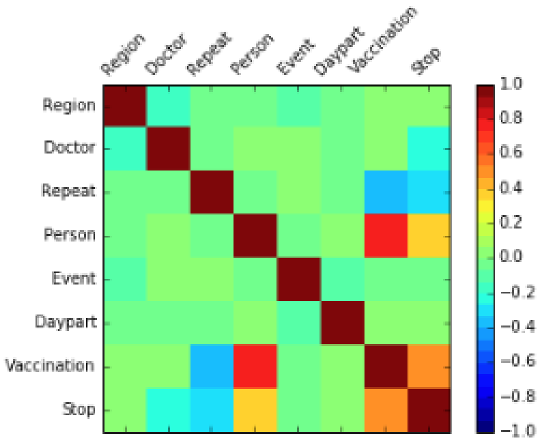


Figure 4.7: Correlation matrix of attributes

	Region	**Doctor**	**Repeat**	**People**	**Event**	**Daypart**	**Vaccinations**	**Stop**
Region	-	-0.177180	-0.010858	-0.023799	-0.083257	-0.054908	0.004419	0.064833
Doctor	-0.177180	-	-0.032628	0.031341	0.008883	-0.002052	0.013382	-0.224716
Repeat	-0.010858	-0.032628	-	-0.057334	0.027300	-0.001337	-0.360510	-0.274447
People	-0.023799	0.031341	-0.057334	-	-0.023142	0.030350	0.746105	0.391760
Event	-0.083257	0.008883	0.027300	-0.023142	-	-0.070426	-0.047213	-0.002799
Daypart	-0.054908	-0.002052	-0.001337	0.030350	-0.070426	-	0.022648	0.006246
Vaccinations	0.004419	0.013382	-0.360510	0.746105	-0.047213	0.022648	-	0.528887
Stop	0.064833	-0.224716	-0.274447	0.391760	-0.002799	0.006246	0.528887	-

Table 4.6: Correlations between attributes expressed in numbers

A negative correlation can be seen between *number of vaccinations* and *repeating appointment* $(-0,60510)$, indicating that a second appointment yields a decrease in number of vaccinations. This can be explained because only a subset of vaccinations requires a second appointment, meaning that the number of vaccinations that a person needs decreases in the second appointment. A strong positive correlation is between *number of vaccinations* and *number of people* $(0,746105)$. A possible explanation is that when the group becomes larger, more vaccinations are needed to cover the whole group. There are also correlations between variables and stop duration in the correlation matrix. It shows a positive correlation between *number of people* and *stop duration*

(0.391760), but also *number of vaccinations* and *stop duration* (0.528887). Both positive values are reasonable because the larger a group or the more vaccinations are required, the longer an appointment would take.

Chapter 5

Evaluation

This chapter explains the experiment setup with theories of performance indicators to evaluate the predictive models. It also explains the steps of the modelling process and evaluation of the experiments. By comparing the different regression models, we can choose the most suitable model that accurately estimates consultation time.

5.1 Experiment setup

After the data have been properly cleaned and pre-processed, different models can be developed for descriptive and predictive purposes. Since the response variable is numeric, we use regression. Specifically, regression trees will be used to model the data. These are classic regression tree, the three ensemble types of regression trees and the M5P model tree. To investigate which model suits the data best, we use R^2 and the *Mean Square Error* (MSE) to evaluate the models.

5.1.1 Bias and variance

A data set can be divided in training data and test data. Training data consist of observational data for the algorithm to learn. With this experience the test data can be used to evaluate the performance of the model that has been made with the algorithm and training data [14]. It is important that no observations from the training set are included in the test set. When these observations are included in the test set it will be hard to identify its actual performance and to know if the model has generalized or has memorized the data. If a model has generalized well enough, it is able to perform its task well on a new data set. However, if the model has only memorized the tasks based on the observations, it will perform poorly and will not be able to predict the correct response values for a new data set [23] [15]. This is also called overfitting. An overfitted regression tree might perform well on the trained data, but can be a complex model that performs incompetently on unseen observations. It is therefore important that not too many attributes are selected and that splitting stops when there is little improvement to be gained.

Besides overfitting, there is also a problem of underfitting where a model is too generic and does not capture all the patterns present in the data. The bias-variance trade-off is a trade-off that continuously needs to be made in statistics and machine learning. A high bias misses the relevant patterns between the response variable and explanatory attributes (underfitting), whereas a high variance can cause the model to be too sensitive to variations in training data (overfitting). When modelling we want to make an accurate model that fits the training data well, but also performs well on unseen instances. It is therefore important to use the right performance metrics to find the right balance between these two. Measuring the performance is relevant for evaluating the trustworthiness of the model and for comparing different approaches [23].

As for the bias-variance trade-off, *Forward Feature Selection* or *Backward Feature Elimination* is a method to decrease the variance by using less attributes than the initial attribute input to simplify the model [23]. Also, another method is to use a large training set to decrease variance. Another way for this trade-off is to use ensemble learning [15]. Boosting trees combine models with a high bias such that an ensemble of these trees leads to a lower bias than the individual models. Similarly, bagging trees combine models with a high variance and ensembles them leading to a lower variance than taking the individual models separately.

This subsection finishes with the Occam's Razor principle. This principle states - referenced to Englishman William of Ockham in the 14th-century - that **"one should not increase, beyond what is necessary, the number of entities required to explain anything"** [23]. This showcases that the simplest model should be chosen to explain data and concerns the balance between overfitting and underfitting. Hence, in statistics and machine learning there has to be a balance made between fitness, generalization, precision and simplicity.

5.1.2 Quality of models

Train/test split and *k*-fold cross-validation

An evaluation technique to build a model is train/test split. This method split the data set in a training set and test set. An algorithm learns on a training set and with the resulting model the accuracy will be calculated based on test data. This accuracy can be of different forms depending on type of learning algorithm. For example, with classification there are measures such as *precision, recall* and *F1 score*. For linear regression there are *Mean Square Error* (*MSE*) or *R-squared* (R^2) as measures. A pitfall when using train/test split is that a single performance indicator does not tell much about its actual reliability of the result [23]. An accuracy of 0.91 would not guarantee an equivalent accuracy score on another data set with the same model. It then suffices to calculate a confidence interval over a single performance indicator. Confidence intervals can only be calculated with the use of multiple measurements and one possibility to do this, is by using *k*-fold cross-validation.

k-fold cross-validation is often used when there are few data instances available or when a performance indicator can only be used over a set of data instances instead of a single instance [23]. A data set will then be divided in k equal subsets. Each subset will be used as a test set with $k-1$ training sets. This repeats k times and leads to k results. It is then possible to analyze one of the individual test sets or to take an average of k test sets. There can be two advantages to use *k*-fold cross-validation. One is that the data set is used as training set and test set leading to a more robust model [23] [15]. Second, it is possible to get k test sets instead of one. Especially with a limited data set, having k test results instead of one can give us more insights into the model reliability. The results are however not completely independent since the k test sets overlap within the *k*-folds as well.

Formalizing quality indicators

The Mean Square Error is a risk metric [23] and is an average of the squared error; this error is a difference between the observation and its predictor, the residual. A variation in the difference can occur due to noise in the data or the predictors do not hold enough information to make a correct prediction of an observation. Because of the exponent a MSE cannot be negative and the closer its value is to zero the better [16]. The MSE and R^2 can be written as Definitions 5.1.1 and 5.1.2 respectively.

***Definition* 5.1.1** (Mean Square Error (MSE))**:**

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2$$

***Definition* 5.1.2** (R^2)**:**

$$R^2 = \frac{TSS-SSE}{TSS} = \frac{\sum(y-\overline{y})^2-\sum(y-\hat{y}^2)}{\sum(y-\overline{y}^2)}$$

where TSS = *Total Sum of Squares* and shows the error of $\overline{y}$ when predicting for y without using information of x. SSE = *Sum of Squared Errors* and shows the error of $\hat{y}$ when predicting for y using information of y.

The R^2 can therefore be seen as the proportional reduction error when the prediction equation, $\hat{y}$, is used to predict y instead of the mean, $\overline{y}$ [2]. This indicator shows how much error $\hat{y}$ is able to proportionally reduce when the predictor equation is used instead of $\overline{y}$. When there is a strong association between the variables x and y then the prediction equation $\hat{y}$ performs better than $\overline{y}$.

5.2 Comparing regression tree models

Machine learning algorithms can be used for predictive analyses. Performance scores and run time complexity are measures for choosing an appropriate algorithm. A factor prior to the algorithm choice is to measure whether data is dependent through time. Since the appointments are scheduled over a time period and the data are ordered based on date and time, we need to ensure that the model does not perform well because of improvements over time. Hence, we would like to have a model with attributes that are independent throughout time. A way to ascertain this time dependency is to perform the modelling with k-fold cross-validation and train/ test split. With train/test split the data set is split into training set and test set in an orderly manner over time, whereas k-fold cross-validation does not compromise this order and is independent of time of the data. For this thesis, the train/ test split ratio is 80/20. When the results of both techniques are comparable, then we know that there is independence between time and data, and no specific time order is essential in the analysis. The performance results of both techniques are shown in Table 5.1 and Figures 5.1 and 5.2

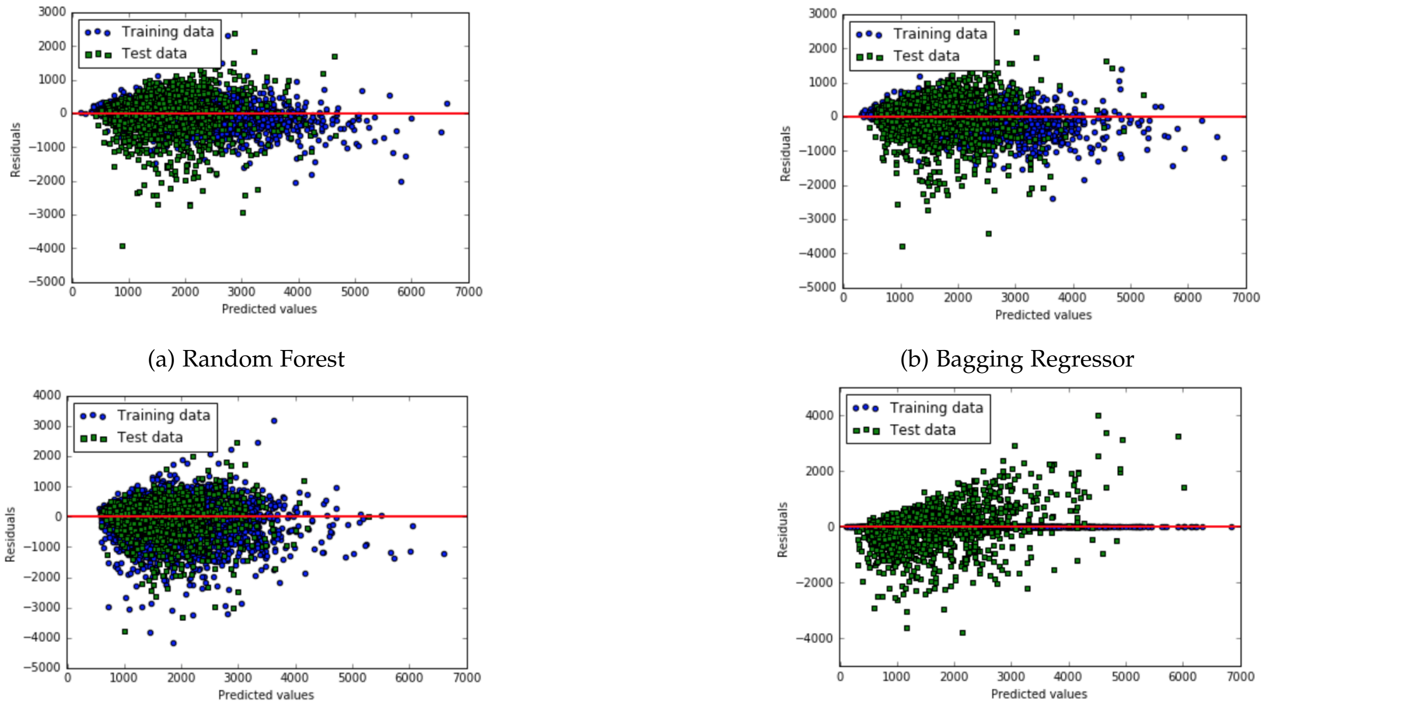
The performance measurements are MSE and R^2 for each model. Table 5.1 shows that all models except for gradient boosting performs well on training data with a high R^2 score of 0.92 for random forest and bagging regressor, and even a R^2 of 1.00 for decision regressor. However, it also appears that a difference is present compared to the prediction on test data for random forest, bagging regressor and decision regressor with a score of $0.53, 0.52$ and 0.23 respectively. The same pattern is seen when using MSE for calculations. This indicates an overfitted model meaning that the models are too complex and performs good on training data but does not generalize well on new data. The scores in test data for gradient boosting show a slightly better test score compared to the other ensemble trees, namely a R^2 of 0.56 and MSE of $355, 217.06$.

Aside from this information, Figure 5.1 presents for each algorithm a graph that visualizes residuals on training data and test data. As explained in Chapter 2 the residual equals the difference between a prediction and an observation. An optimal model has residuals of zero, which in this case displays with a red line for each residual graph. The decision regressor has perfect training residuals of 0 which is in line with the MSE and R^2 scores. Nevertheless, there is a poor score on test data which displays a strongly overfitted model. Random forest and bagging regressor show a more evenly spread test error with a few outliers, whereas gradient boosting has stronger residual differences with more data points below the red line. This implies that gradient boosting predicts appointment times longer than the actual time.

	MSE		**R^2**		**Runtime**
Method	*train*	*test*	*train*	*test*	*in seconds*
Random forest	67,296.45	384,423.94	0.92	0.53	0.487
Bagging regressor	69,577.24	389,658.65	0.92	0.52	0.643
Boosted gradient regressor	288,292.89	355,217.06	0.66	0.56	1.201
Decision regressor	0.00	623,559.29	1.00	0.23	0.083

Table 5.1: Performance scores of four regression models by looking at MSE, R^2 and runtime. These models are made by using train/ test split.

There are also differences in runtime, where the decision regressor comes out as a slightly faster algorithm; 0.083 seconds to fit the model. This can be explained due to the modelling of a single tree, whereas the ensemble algorithms develop multiple estimators to make their predictions and can thus take longer to make the model. When comparing the three ensemble trees, gradient boosting has a longer runtime. A possible explanation for this is its sequential, stage wise property where every next tree is build on top of the previous



(a) Random Forest

(b) Bagging Regressor

(c) Boosted Gradient Regressor

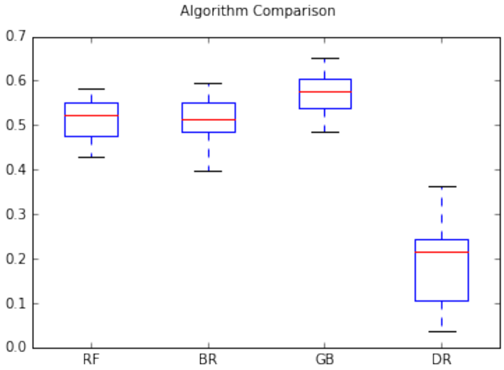
(d) Decision Regressor

Figure 5.1: Residuals on training and test set by using train/test split on four types of regression tree models

ones, meaning that for every tree new re-calculations need to be made whereas the other ensemble trees happen more randomly and are less dependent on the previous trees.

A different modelling technique is to use *k*-fold cross-validation. Figure 5.2 are performance results of using 10-fold cross-validation with computation times. The same pattern can be derived from these results as with train/test split. Decision regressor performs poorer according to R^2 than the ensemble trees and a possible cause for this is the development of a single tree. In general, gradient boosting has a higher score than the other two ensemble trees, but requires a longer runtime and is twice as long than other ensemble trees. Next to these performance scores, the figure also showcases standard deviations. Boxplots are shown in the left graph that visualizes the same information. These standard deviations show not too substantial fluctuations when generating new models with the same algorithm. Even though the performance scores with *k*-fold cross-validation are lower in comparison to train/ test split, the differences are not too conclusive to choose train/test split over *k*-fold cross-validation when looking at the performance scores and standard deviations. Since *k*-fold cross-validation can give us more reliable models, we choose to use this technique to work with.

Based on the experiment results the models are too overfitted to perform a good predictive model, such that there are too many features than the available data. The trees are too deep and too complex, resulting in low training residuals but high test residuals. An option to reduce the overfitness is by using fewer attributes to counterweight for the depth and complexity of the trees. Comparing the four models result in random forest as machine learning technique for optimizing the model. The figures show an overall faster runtime of random



	Performance		*Runtime*
Method	R^2	*std*	*in seconds*
RF	0.494117	0.058	6.039
BR	0.505692	0.064	5.631
GB	0.573261	0.052	12.656
DR	0.168796	0.121	0.979

Figure 5.2: Model performances by using 10-fold-cross-validation. The left figure shows boxplots of R^2 for random forest, boosted regressor, gradient boosted regressor and decision regressor. The right table shows scores expressed in numbers. Std = standard deviation of the R^2-mean.

forest in comparison to the other two ensemble trees and performs better than a single decision regressor. Regardless of a lower performance score than gradient boosting, random forest is able to model a better model with a training score of 0.92. A next step is to optimize the model such that the model is able to perform well on test data too. Section 5.3 covers this next step in more detail. The goal in optimizing the model is to achieve at least the same performance score as the results above but with fewer attributes and more simplicity.

5.3 Random forest model optimization

The R^2 score of random forest without any reduction in the number of features is 0.49 with k-fold cross-validation and has a runtime of 6.039 seconds. This model can hopefully be less overfitted with fewer attributes for more simplicity.

Figure 5.3 illustrates the first 34 relative important features, with the most important ones ranking higher in the figure. The relative importance of $feature_i$ is determined by taking the error reduction of $node_i$, weighted by the sample proportion that reaches that node split [22] [7]. An interesting observation is that doctors play a more influential role in determining appointment time than vaccination type. The variable a_i indicates a doctor's id and is in alignment with the boxplots of time duration based on doctor in Figure 4.4. As explained in Chapter 4 the figure of the descriptive analysis on doctors shows fluctuating medians in appointment time and reveals the influence that doctors have on appointments, similar to the relative importance histogram. With the use of this histogram we can perform forward feature selection to construct an optimized model.

The approach in reducing the number of attributes works by looking at the feature importance of Figure 5.3 and to start building a model with the most important feature first. Based on the ranking in the figure, we then add one feature at a time until the score of the model is around 0.49. Implementing this method results in a model with the first 34 important features and is a reduction in comparison to the total data input of 91 features. The random forest model with 34 attributes has the following cross-validated performance score:

```
R^2 score                                   0.49

Root mean squared error                     686.201136694 seconds

Mean absolute error                         501.206894758 seconds

Runtime                                     3.069

Total number of instances                   6615
```

The performance indicators show that this improved model has a similar performance score with fewer attributes and a two times smaller runtime than the previous random forest model.

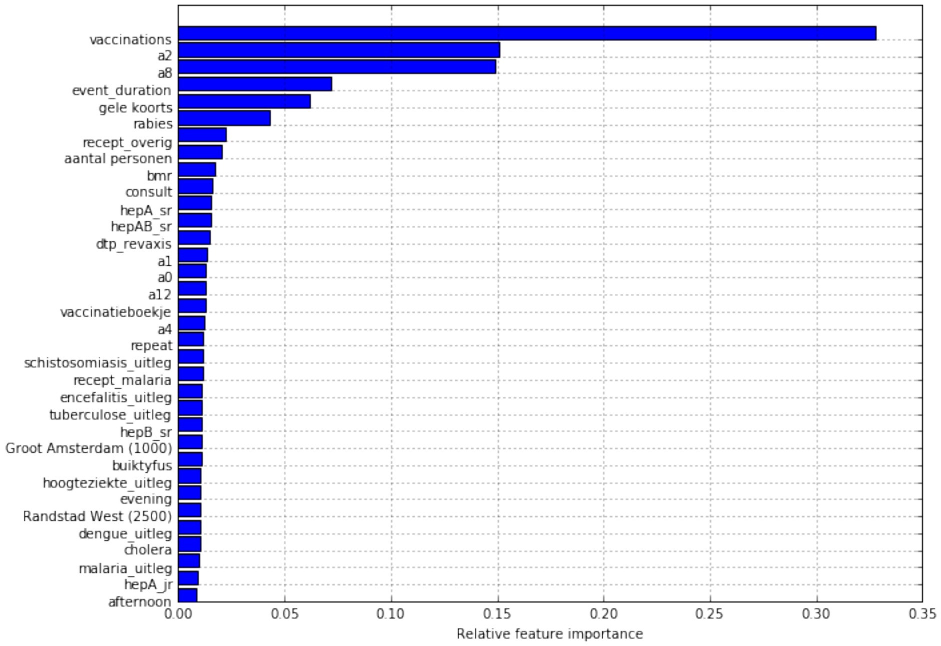


Figure 5.3: Top 34 important relative features based on random forest

5.4 Ensemble tree versus model tree

A drawback of using ensemble trees is its black-box property; the visualization of a regression tree is rather complex. When taking random forest models, the algorithm makes many random estimators and takes the average of these estimators as end result. A possibility is to extract one of these estimators for interpretation. This however does not guarantee that it represents the average overall case and represents a possible state out of many regression estimators. This contributes to the notion that in machine learning there is a trade-off between accuracy of a model and the interpretability of a model. The experiments in Sections 5.2 and 5.3 focus mainly on improving the quantitative accuracy of a prediction model. Section 5.4 provides insights into the visualization of appointment time predictions. A model tree algorithm makes a decision tree with multiple regression equations as leaves. This ensures a less complex regression tree than the other algorithms and makes it easier to interpret for end users. With the use of Weka [12] the algorithm M5P runs on the data and results in the visualization and performance of Figure 5.4. The M5P algorithm is a reconstruction of

Quinlan's algorithm [9], introduced in Chapter 3. This model illustrates a tree with one node and two leaves. Furthermore, Table 5.2 illustrates the performance of M5P in comparison to random forest.

	M5P	**Random forest**
R^2	0.618	0.49
Mean absolute error (in seconds)	403.2736	501.2069
Root mean squared error (in seconds)	569.7166	686.2011
Runtime (in seconds)	9.16	3.069

Table 5.2: Performance comparison of M5P and random forest, using 10-fold cross-validation

Having two different models, namely random forest and M5P, we are able to compare both results and derive if the results are in line with one another. Based on Table 5.2 M5P has a longer runtime than random forest. However, even without feature selection M5P is able to perform better with lower mean absolute error than random forest. The algorithm also has a higher R^2 score and is able to fit the data better with the observations.

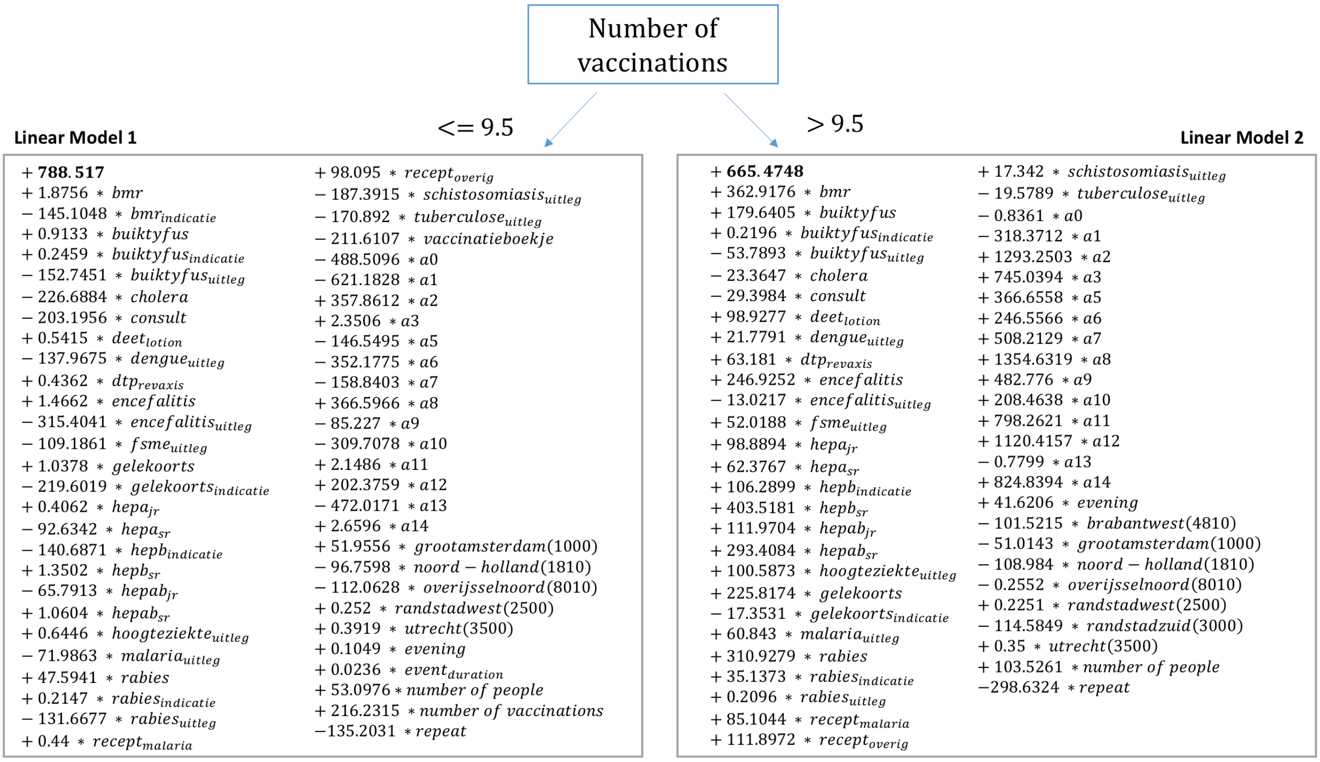
Overall, both models include the same attributes with some deviations in the number of attributes. M5P uses more attributes to build the model because it does not perform feature selection. Also, from the random forest results we derive that, next to the number of vaccinations, doctors a2 and a8 have two of the highest relative feature importances. With M5P we are able to look more closely on this impact in seconds. From this model we see that they have two of the highest coefficients on both leaves. It shows that the number of vaccinations are dependent of how much extra appointment time is needed when doctor a2 or a8 performs the consultation. For example, with vaccinations fewer than 9 a consultation requires 366.5966 seconds extra whereas vaccinations of more than 10 leads to an extra time of $1,354.6319$ seconds. Table 5.3 compares two M5P models including and excluding doctors in the model. The results show that doctors are indeed an important factor when modelling the data. Excluding doctors result in a higher error rate, but also increases complexity of the model with $2,855$ rules instead of 2 rules.

	With doctor	**Without doctor**
R^2	0.618	0.368
Mean absolute error (in seconds)	403.2736	538.5737
Root mean squared error (in seconds)	569.7166	738.7341
Number of rules	2	2,855
Runtime (in seconds)	9.16	5.83

Table 5.3: Performance comparison of M5P including the doctor attribute and M5P excluding doctor. The models are made by using 10-fold cross-validation.

Interestingly, the number of vaccinations in the left leaf influences the appointment time, whereas in the right leaf the number of vaccinations does not hold an impact on the eventual appointment time. It shows that at a certain point the number of vaccinations has no further influence on the time estimation.

When comparing the same attributes from both leaves with one another, we see that attributes on the right with negative values have lower values. The same is with positive attributes where the right attributes present higher coefficient values than their left equivalent. This implies that an increase in number of vaccinations yield an extra increase in time for each of these attributes.



Linear Model 1

$+\mathbf{788.517}$
$+1.8756 \ast bmr$
$-145.1048 \ast bmr_{indicatie}$
$+0.9133 \ast buiktyfus$
$+0.2459 \ast buiktyfus_{indicatie}$
$-152.7451 \ast buiktyfus_{uitleg}$
$-226.6884 \ast cholera$
$-203.1956 \ast consult$
$+0.5415 \ast deet_{lotion}$
$-137.9675 \ast dengue_{uitleg}$
$+0.4362 \ast dtp_{revaxis}$
$+1.4662 \ast encefalitis$
$-315.4041 \ast encefalitis_{uitleg}$
$-109.1861 \ast fsme_{uitleg}$
$+1.0378 \ast gelekoorts$
$-219.6019 \ast gelekoorts_{indicatie}$
$+0.4062 \ast hepa_{jr}$
$-92.6342 \ast hepa_{sr}$
$-140.6871 \ast hepb_{indicatie}$
$+1.3502 \ast hepb_{sr}$
$-65.7913 \ast hepab_{jr}$
$+1.0604 \ast hepab_{sr}$
$+0.6446 \ast hoogteziekte_{uitleg}$
$-71.9863 \ast malaria_{uitleg}$
$+47.5941 \ast rabies$
$+0.2147 \ast rabies_{indicatie}$
$-131.6677 \ast rabies_{uitleg}$
$+0.44 \ast recept_{malaria}$
$+98.095 \ast recept_{overig}$
$-187.3915 \ast schistosomiasis_{uitleg}$
$-170.892 \ast tuberculose_{uitleg}$
$-211.6107 \ast vaccinatieboekje$
$-488.5096 \ast a0$
$-621.1828 \ast a1$
$+357.8612 \ast a2$
$+2.3506 \ast a3$
$-146.5495 \ast a5$
$-352.1775 \ast a6$
$-158.8403 \ast a7$
$+366.5966 \ast a8$
$-85.227 \ast a9$
$-309.7078 \ast a10$
$+2.1486 \ast a11$
$+202.3759 \ast a12$
$-472.0171 \ast a13$
$+2.6596 \ast a14$
$+51.9556 \ast grootamsterdam(1000)$
$-96.7598 \ast noord-holland(1810)$
$-112.0628 \ast overijsselnoord(8010)$
$+0.252 \ast randstadwest(2500)$
$+0.3919 \ast utrecht(3500)$
$+0.1049 \ast evening$
$+0.0236 \ast event_{duration}$
$+53.0976 \ast number\ of\ people$
$+216.2315 \ast number\ of\ vaccinations$
$-135.2031 \ast repeat$

Linear Model 2

$+\mathbf{665.4748}$
$+362.9176 \ast bmr$
$+179.6405 \ast buiktyfus$
$+0.2196 \ast buiktyfus_{indicatie}$
$-53.7893 \ast buiktyfus_{uitleg}$
$-23.3647 \ast cholera$
$-29.3984 \ast consult$
$+98.9277 \ast deet_{lotion}$
$+21.7791 \ast dengue_{uitleg}$
$+63.181 \ast dtp_{revaxis}$
$+246.9252 \ast encefalitis$
$-13.0217 \ast encefalitis_{uitleg}$
$+52.0188 \ast fsme_{uitleg}$
$+98.8894 \ast hepa_{jr}$
$+62.3767 \ast hepa_{sr}$
$+106.2899 \ast hepb_{indicatie}$
$+403.5181 \ast hepb_{sr}$
$+111.9704 \ast hepab_{jr}$
$+293.4084 \ast hepab_{sr}$
$+100.5873 \ast hoogteziekte_{uitleg}$
$+225.8174 \ast gelekoorts$
$-17.3531 \ast gelekoorts_{indicatie}$
$+60.843 \ast malaria_{uitleg}$
$+310.9279 \ast rabies$
$+35.1373 \ast rabies_{indicatie}$
$+0.2096 \ast rabies_{uitleg}$
$+85.1044 \ast recept_{malaria}$
$+111.8972 \ast recept_{overig}$
$+17.342 \ast schistosomiasis_{uitleg}$
$-19.5789 \ast tuberculose_{uitleg}$
$-0.8361 \ast a0$
$-318.3712 \ast a1$
$+1293.2503 \ast a2$
$+745.0394 \ast a3$
$+366.6558 \ast a5$
$+246.5566 \ast a6$
$+508.2129 \ast a7$
$+1354.6319 \ast a8$
$+482.776 \ast a9$
$+208.4638 \ast a10$
$+798.2621 \ast a11$
$+1120.4157 \ast a12$
$-0.7799 \ast a13$
$+824.8394 \ast a14$
$+41.6206 \ast evening$
$-101.5215 \ast brabantwest(4810)$
$-51.0143 \ast grootamsterdam(1000)$
$-108.984 \ast noord-holland(1810)$
$-0.2552 \ast overijsselnoord(8010)$
$+0.2251 \ast randstadwest(2500)$
$-114.5849 \ast randstadzuid(3000)$
$+0.35 \ast utrecht(3500)$
$+103.5261 \ast number\ of\ people$
$-298.6324 \ast repeat$

Figure 5.4: Visualization of the predictive model with the use of M5P

5.5 Main findings

From the experiments it is clear that ensemble trees perform better than a single decision regressor. An explanation for this difference is its ensemble property, where ensemble trees take many single decision regressors, thus requiring longer computation time. By averaging the performance of these tree results, it will in general perform better and is more robust than a single tree. However, the models - regardless of algorithm - showcase overfitted models that perform well on training data, whilst displaying poor test performance due to the complexity of the models. When choosing a regression tree algorithm that suits best with the data set, it can be said that random forest comes out as optimal model with a test score of 0.49 and a short runtime. To further optimize the model, we reduce the number of attributes, while receiving the same performance score as before. With relative feature importance we are able to reduce the number of attributes to increase the simplicity of the model, leading to a model with 34 input attributes instead of 91.

When comparing the random forest with model tree M5P, we conclude that M5P performs better with a lower error rate and is a simpler model for interpretation. Regardless of algorithm choice, it can be said that the number of vaccinations holds the strongest influence. Both in random forest and M5P this attribute ranks highly in both relative terms (Figure 5.3) and in absolute terms (Figure 5.4). Also, doctors play, with similar coefficients, an important part in time duration determination. This also leads to higher complexity when excluding this attribute from the model. Thus, M5P can be best used for predicting the model since it performs better and is able to visualize the influence of attributes in absolute values. This enables end users to immediately act on it.

Chapter 6

Discussion

The analyses of the data were based on GPS data. The raw GPS data represented pure numbers of a car's stop duration, but did not account for human action. It is unknown what exactly took place during a stop duration. Different events could occur from parking place to the customer's physical place and these are unexplainable factors, meaning that there is always noise in the data. This in turn might not represent the actual time influence of the available attributes. E.g., did a Hepatitis B Senior vaccine require an extra 403.5181 seconds, and did an appointment in Groot Amsterdam really result in an extra 51.9556 seconds, or was there a delay in travelling? These are hard questions to know and to measure. It does however highlight the notion that raw data are not able to explain every hard detail and that there is always an unpredictable human factor that plays an important part in events. This also means that certain associations do not imply causal relationships necessarily. There can be alternative reasons for these associations that cannot be derived from the available data.

It should also be noted that the models are highly dependent on the doctors. This shows that time duration depends on skill or doctor's practices, and not only on type of vaccination or group size. This also makes the model less secure for future use. That is, if doctors decide to leave the company, then the model would be less useful for time estimation. It is therefore of importance that clear choices should be made when implementing this model for future use within the company. For example, we can question if we should give doctors enough time for consultation based on their needed work time. Or give every doctor the same work time regardless of his work effort.

Thus, data-driven models should therefore be regarded with care and should be seen as a tool. Regardless of the models that are made, it should be our own human domain knowledge that calls the action. In the end, human decisions are made by human actions. We cannot explain everything that happened based on data only and data-driven predictions may conflict with desirable future policies.

Chapter 7

Conclusions

The aim of this thesis is to achieve a more accurate estimation for consultation time. Based on historic data from Thuisvaccinatie the appointment time is estimated using different regression techniques. The available data for this project contained data regarding customers' appointment data and doctors' GPS data. By combining these two sets we are able to create a setting where stop duration from GPS data as time can be explained from customers' appointment data.

Since time is a numeric continuous response variable, it was favorable to use regression techniques. More precisely, different experiments were conducted with the use of regression trees and the ensemble trees random forest, bagging trees and gradient boosted trees, and model tree. Based on these experiments, it was visible how challenging the trade-off was between model complexity and model interpretation. From the outcomes of the experiments on regression trees it came forward that random forest performed better. By reducing the number of data attributes from 91 to 34 we were able to make a model with a similar $R^2 = 0.49$ with feature reduction. However, due to its black-box character we were able to optimize the model but at the expense of interpreting the model. To gain results from both aspects we have made a visualization by making a model tree on the data set. With these two approaches there is a focus on optimizing quantitative performance on a model and on interpreting a visualized model based on the same data set. Based on these results, we concluded that in terms of interpretation and performance the model tree came out as best.

From these approaches and descriptive analyses there are several conclusions that can be made. There are correlations between time duration and repeat, and between time duration and group size. These two observations are logically in line with the current practices of Thuisvaccinatie. A more engaging observation is the strong association between time duration and doctor, where there seems to be some differing fluctuations in time based on doctor. In general, however, the appointments take on average thirty minutes with a standard deviation of fifteen minutes. This means that the current method of assigning thirty minutes for every appointment is quite accurate. With a standard deviation of fifteen minutes, it does leave room for more precise time estimation with the use of the M5P prediction model. With M5P we are able to reduce the standard deviation from fifteen minutes to seven minutes, creating a model that is able to estimate the consultation time more accurately.

7.1 Future work

The purpose of a time estimation model is to make future calculations of appointment duration more accurate. There are several extensions to work on to optimize the model and to gain a better understanding of customers.

- **Parameter tuning** - A technique not implemented in the model optimization is parameter tuning [23]. Since we mainly focus on the predictive power of the attributes, parameter tuning is out of the scope but can help in further optimizing the model. We can ensure that pruning takes place earlier in the ensemble trees, making the model less complex and overfit.
- **Availability of more diverse attributes** - It can be insightful to add more attributes of customers that are more based on the group composition to make more precise calculations. For example, age of customers with differences in children or adolescence, or travel destination to see which countries require longer consultation time.
- **Clocking system of doctors** - Appointment time is derived from GPS data of the doctors. However, the GPS data do not explain the reason for travelling, meaning that there are data points in the set that does not imply an appointment. Also, these stop duration of GPS include the walking process to and from an appointment. This makes the stop duration not representative for the time estimate of the available attributes in the data. With the use of a clocking system for doctors, we are able to make more accurate time estimations that are mainly focused on an appointment. Also, these observations can be more easily linked to the appointment data entries.

Bibliography

[1] A. Afifi and S. Azen. *Statistical analysis A Computer Oriented Approach*. Academic Press, 111 Fifth Avenue, New York, New York 10003, 1 edition, 1972.

[2] A. Agresti and B. Finlay. *Statistical Methods for the Social Sciences*. Prentice Hall, Inc., Upper Saddle River, New Jersey 07458, 4 edition, 2009.

[3] P. Austin, J. Tu, J. Ho, D. Levy, and D. Lee. Using methods from the data-mining and machine-learning literature for disease classification and prediction: a case study examining classification of heart failure subtypes. *Journal of Clinical Epidemiology*, 2016.

[4] J. Bos and G. Schonewille. Vakantiegeld-enquête 2016. Technical report, Nationaal Instituut voor Budgetvoorlichting, 2016.

[5] L. Breiman. Bagging predictors. *Machine Learning*, (26):123–140, 1996.

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

[7] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth International Group, Belmont, California 94002, 1 edition, 1984.

[8] N. Chok. Pearson's versus Spearman's and Kendall's correlation coefficients for continuous data. *University of Pittsburgh*, 2008.

[9] C. Deepa, K. SathiyaKumari, and V. Sudha. Prediction of the compressive strength of high performance concrete mix using tree based modeling. *International Journal of Computer Applications*, 2010.

[10] J. Elith, J. Leathwick, and T. Hastie. A working guide to boosted regression trees. *Journal of Animal Ecology*, (77):802–813, 2008.

[11] F. Fenner. Smallpox and its eradication. *WHO*, 1988.

[12] E. Frank, M. Hall, I. Witten, and C. Pall. *The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques"*. Morgan Kaufmann, 2016.

[13] J. Galama. Opkomende virusinfecties. *Nederlands Tijdschrift voor Geneeskunde*, 2001.

[14] G. Hackeling. *Mastering Machine Learning with Scikit-learn*. Packt Publishing Ltd., Livery Place, 35 Livery Street, Birmingham B3 2PB United Kingdom, 1 edition, 2014.

[15] G. James, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning*. Springer-Verlag New York Inc., New York, 1 edition, 2013.

[16] E. Lehmann and G. Casella. *Theory of Point Estimation*. Springer, New York, 2 edition, 1998.

[17] D. Malerba, F. Esposito, M. Ceci, and A. Appice. Top-down induction of model trees with regression and splitting nodes. *Transactions on pattern analysis and machine intelligence*, 2004.

[18] J. Mendes-Moreira, A. Jorge, J. Sousac, and C. Soares. Comparing state-of-the-art regression methods for long term travel time prediction. *Intelligent Data Analysis*, (16):427–449, 2012.

[19] D. Moore, S. Davey, and B. Lees. A new method for predicting vegetation distributions using decision tree analysis in a geographic information system. *Journal of Environmental Management*, (15):59–71, 1991.

[20] A. Prasad, L. Iverson, and A. Liaw. Newer classification and regression tree techniques: Bagging and random forests for ecological prediction. *Ecosystems*, (9):181–199, 3 2006.

[21] J. Quinlan. Learning with continuous classes. *World scientific*, 1992.

[22] Scikit-Learn. 1.13. Feature selection. `http://scikit-learn.org/stable/modules/feature_selection.html`, 2010.

[23] W. van der Aalst. *Process Mining*. Springer, 2 edition, 2016.

[24] F. Zhang, X. Zhu, T. Hu, W. Guo, C. Chen, and L. Liu. Urban link travel time prediction based on a gradient boosting method considering spatiotemporal correlations. *International Journal of Geo-Information*, 2016.

[25] Y. Zhang. *Uncertainty Associated with Travel Time Prediction: Advanced Volatility Approaches and Ensemble Methods*. PhD thesis, University of Maryland, 789 East Eisenhower Parkway, 2015.