FH JOANNEUM
University of Applied Sciences

CALCULUS OF LENGTH OF STAY ON THE BASIS OF THE GERMAN DRG DATA USING CART AND K-NEAREST NEIGHBOUR

A master thesis by

VICENT JOSEPH ANDREU TRAWICK

advised by

DI Dr. Robert Mischak, MPH

and submitted to the

Institute for eHealth

of the

Graz University of Applied Sciences

in partial fulfillment of the
requirements for the degree of a
Master of Science (MSc)

July 2019

"Science knows no country, because knowledge belongs to humanity, and is the torch which illuminates the world."(Louis Pasteur)

Acknowledgements

First of all, I would like to thank my thesis advisor DI Dr. Robert Mischak, MPH of the Institute of eHealth at Graz University of Applied Sciences, whose expertise helped me formulate my research topic and methodology of this paper. He was more than willing to help me whenever I had a question with my research.

I would also like to thank my parents, for their support and encouragement throughout the process of researching and writing this thesis. You are always there for me.

Thank you.
Vicent Andreu

Abstract

The length of stay (LOS) is defined as the duration of a single episode of hospitalization. It is used as an indicator of how efficient a hospital utilizes its resources. LOS is also linked to hospital related side effects. A higher stay tends to increase both the infections acquired in the hospital and the medication side effects. An effective methodology is needed in order to handle large datasets to optimise prediction accuracy. In the present thesis machine learning models were used to predict the LOS extracted from the electronic health records of a German hospital. The variables identified to predict the LOS were the age, gender, ICD code, number of procedures and the day of the week that the patient was accepted. The first objective was to classify between a long-term patient and a short-term patient. The classifiers used were the classification tree and the k-nearest neighbour. It was found that the classification tree was the best classifier for this dataset, with an AUC of the ROC of 0,96, an accuracy of 97%, a precision of 74% and a recall of 64%. Afterwards, a regression tree model was trained to predict the exact length of stay in days of the patients. The regression tree model has a MAE of 3,57, an RMSE of 10,47 and an R-squared of 0,56.

Zusammenfassung

LOS ("length of stay") bezieht sich auf die Aufenthaltsdauer der Patienten in einem Krankenhaus und es ist ein guter Indikator für wie effizient ein Krankenhaus die vorhandenen Ressourcen benutzt. Die jeweilige Krankenhausatmosphäre spielt eine wichtige Rolle in dem Wohlbefinden der Patienten und kann dementsprechend die Aufenthaltsdauer (LOS) beeinflussen. Ein längerer Krankenhausaufenthalt kann zu Infektionen und erhöhten Nebenwirkungen von Medikamenten führen. Um die Vorhersagegenauigkeit von großen Datenmengen zu optimieren, wird eine effektive Methode benötigt. In der vorliegenden These wurden elektronische Patientenakten von einem Krankenhaus in Deutschland mit Hilfe von Modellen und maschinellem Lernen analysiert, um die Aufenthaltsdauer von Patienten vorherzusagen. Einbezogen sind die Variablen Alter, Geschlecht, ICD Code, Anzahl der Krankenhausbehandlungen, und der Wochentag an dem ein Patient ins Krankenhaus aufgenommen wurde. Das erste Ziel war, einen Patienten als Langzeitpatient oder Kurzzeitpatient einzustufen. Als Klassifikatoren wurden ein Klassifikationsbaum und der k-nächste-Nachbarn Algorithmus benutzt. Der Klassifikationsbaum war für diesen Datensatz am besten geeignet mit einem AUC von 0,96, einer 97% Genauigkeit, einer 74% Präzision, und einem Trefferquote von 64%. Danach wurde ein Regressionsbaum Modell angepasst um die exakte Dauer eines Krankenhausaufenthaltes (in Tagen) zu prognostizieren. Das Regressionsbaum Modell hat ein MAE von 3,57, ein RMSE von 10,47 und ein R-quadrat von 0,56.

Contents

List of Figures

List of Tables

Chapter 1

General information of predictive analytics in the healthcare system

1.1 Introduction

The healthcare system faces several problems, such as high costs and variable performance [Amarasingham et al., 2014]. It could potentially benefit from predictive analytics and big data using a large quantity of information in the form of routine data available in electronic medical records. These big pools of data can be used to help clinical decision making and health system operations. The goal is to analyze and compare the clinical data in order to recognize patterns and to extract meaningful features. Recently, the term "data mining", has been used frequently in the medical literature. It can be defined as the procedure of "selecting, exploring and modeling large amounts of data in order to discover unknown patterns or relationships which provide a clear and useful result to the data analyst" [Bellazzi and Zupan, 2008]. Data mining has been used in many other fields, including marketing, engineering, crime analysis, web mining, and many more [Chen et al., 2006]. One of the potential benefits of having patient data in electronic health records is that the information can be updated instantly and the data from different medical departments or even hospitals can be gathered in a single place.
New models of predictive analytics can be extracted from disease specific data in medical records, both with the predictors of a disease and its outcome variables. Predictive analysis is really useful in the beginning of a clinical encounter, due to the high uncertainty of prognostics with limited information, and the limited resources available. These techniques could guide in the decision making process, making healthcare more cost-effective and more responsive to patients' needs [Janke et al., 2016]. Due to advancements in computational power, more complicated decision tools are able to be used. This could lead to the discovery of new relationships between variables.

The German diagnostic-related group (G-DRG) system [Geissler et al., 2011] is based on the Australian refined diagnosis related-groups (AR-DRG). Treatment cases are assigned a DRG code based on the following: medical diagnosis and procedures, patient characteristics, length of stay, duration of ventilation, reason for hospital admission and discharge. Each DRG code is linked to a fixed cost calculated by the Institute for the Hospital Remuneration System (InEK). The G-DRG system applies to all hospitals and patients in Germany except rehabilitation, psychiatric, psychosomatic or psychotherapeutic patients and facilities.

1.2 Motivation

Length of stay (LOS) is defined as the duration of a single episode of hospitalization. It is calculated as the difference between discharge and admission dates. If the patient is admitted in the hospital and then leaves on the same day, the lenght of stay for this specific patient is 0. LOS in this thesis was defined as the patient staying overnight. Each day that the patient stayed after midnight in the hospital is counted as one day in the lenght of stay. LOS serves as an important factor that indicates the efficiency of the hospital management, the use of the hospital's resources, in addition to the patients' quality of care [Baek et al., 2018]. A reduced length of stay in a hospital can lower the risk of hospital related side effects, such as infections acquired in the hospital or medication side effects [Bueno et al., 2010]. An increase in the age of the population will result in an increase and frequency of length of stay in a hospital. Being aware of the factors that cause a greater length of stay will allow for better planning by the hospital. In conclusion, the healthcare system can benefit by knowing the length of stay because they will be able to manage bed stay in a more efficient way.
Several scientific studies have been conducted in the topic of management of LOS in hospitals. The majority dealt with determining the LOS based on a specific department such as predicting of the LOS of elderly people in institutional long-term care [Xie et al., 2005], in psychiatric wards [Sharma et al., 2015], or in the intensive care unit [Kapadohos et al., 2017]. The other studies on the topic of LOS, were based on a specific condition, predicting the LOS based on patients with a total knee replacement [Carter and Potts, 2014], patients with heart failure [Foraker et al., 2014], or patients with hip fractures after undergoing surgery [Neuman et al., 2014]. Care must be taken when analyzing the LOS of patients with the same disease or within the same department due to complex individual differences or variations of organization within a specific department [Baek et al., 2018]. For this reason, all departments and all activities within the hospital will be taken into

account in this thesis. Data will be extracted from electronic health records (EHR) to determine what factors contribute to an increase of LOS within a hospital.

1.3 Research questions

The aim of this thesis is to identify a suitable machine learning model based on classification and regression trees (CART) and on k-nearest neighbour in order to predict the LOS of a patient. Additionally, the factors and variables which contribute to an increase of LOS will be identified. In this master's thesis, the following research questions will be addressed:

- Using routine data extracted from hospital electronic health records, can effective predictors of an individual patient be used in order to predict whether a patient will be a long-term patient (staying in the hospital for 30 days or longer) or a short-term patient(staying less than 30 days) with the method of classification trees or with the algorithm of k-nearest neighbour?
- Is it possible to use a regression tree in order to calculate the exact length of stay in days using routine data extracted from hospital electronic health records?
- Which techniques are the most effective for choosing an appropriate training dataset for the prediction of length of stay, and how is it possible to solve overfitting problems related to machine learning algorithms?

1.4 Material and methods

Research subjects were extracted from a database of patients admitted to a hospital in Germany between April 2012 and December 2015. Patients were analyzed with process pattern analysis using the technique of a decision tree and with k-nearest neighbour to predict whether the LOS will be short or long-term. Afterwards, a regression tree will be made in order to predict the length of stay in days. The program Matlab with its machine learning routines will be used for this thesis to create the models and validate them.

1.5 Structure of the thesis

The thesis is structured as follows: Chapter 2 provides the theory behind the machine learning models k-nearest neighbour, Support Vector Machines, Classification and Re-

gression Trees and Neural Networks in addition to a literary review on the application of predictive models in medicine. Chapter 3 presents a description of the data extracted from the electronic health records used to create the models. Afterwards the preprocessing of the extracted data is explained, along with the statistical analysis in order to understand from multiple perspectives what could affect the LOS in the hospital. In chapter 4 the results of the different machine learning models are presented. Chapter 5 provides the conclusions of the thesis and a discussion section.

Chapter 2

Literary review of commonly used Machine Learning Techniques in medicine

With the exponential growth of medical data placed in databases, appropriate models are needed to extract meaningful data out of it. The purpose of machine learning is to "optimize a performance criterion based on previous experience" [Larranaga et al., 2006]. It uses statistical theory to create models based on predictions made from previous findings. Pattern recognition refers to the discovery of regularities in information through computer algorithms and later classifying the data in categories [Bishop, 2006]. Pattern recognition tries to use information processing to solve problems that encompass a wide variety of topics, such as speech recognition, classification of handwritten characters and medical diagnosis. Humans tend to solve this in an effortless way, but the application in computers has been a challenge [Bishop et al., 1995]. This chapter will describe the most common machine learning techniques used for prediction in medicine.

2.1 k-nearest neighbour

The model of k-nearest neighbour is a method used for classification, pattern recognition and for regression. k-nearest neighbor is a supervised learning algorithm. It works well when each decision boundary is very irregular, such as handwritten digits or satellite image scenes. This classifier uses the cases themselves to classify instances and does not need to be fit into a model [Cooper et al., 1997]. It is considered a memory based learning algorithm because it accumulates the training instances in a lookup table and interpolates from these [Islam et al., 2007]. They are often the best performers for real-life solutions although they perform poorly for high-dimensional problems in classification [John Lu,

2010]. It is based on the principle that instances inside of a database with similar properties will be near each other. Instances are considered as points inside of a n-dimensional space in which each n-dimension correlates to a n-feature used to characterize an instance. The relative distance between these points matter more than their absolute position on the space [Maglogiannis, 2007]. k-nearest neighbour uses the prototype method in order to depict the training data into points in the feature space. This prototype has an associated class label. The algorithm must find out how many protoypes to use and where to apply them. There are many ways to calculate the distance between instances, such as the Manhattan, Minkowsky or the Chebychev distance, but the algorithm usually uses the Eucledian distance in the feature space in order to determine which is the specific prototype that is closest to a point x. Given the query point x, the algorithm finds the k training points that are closest to x and afterwards it classifies with regard to the majority of votes given by the k neighbours [John Lu, 2010]. In other words, it finds the number of k cases that are most similar to the new case. It's a form of care-based reasoning [Cooper et al., 1997]. Each feature is converted in order to have mean zero and variance 1, just in case each feature takes different units. The performance of the algorithm varies depending on the number of k values assigned. There is no principled way to choose the k value. A common method of choosing the k value is by cross-validation or by trying multiple values of k and calculating the classification error and choosing the k that provides the least error [Maglogiannis, 2007]. Odd numbers of k are commonly used to break ties (1,3,5,7 or 9). The bigger the k values, the more they help to lower the consequences of noise within the training dataset, although caution must be taken with choosing large values of k because they tend to misclassify if the individual classes are not very separated [Islam et al., 2007].

2.2 Support Vector Machines

Support vector machines (SVM) became popular for solving problems in classification, regression and novelty detection [Bishop, 2006]. The way support vector machines work is by processing the data to get a higher dimension than the original set had in order to be able to separate into two different categories with a hyperplane [Larranaga et al., 2006]. It is in this high dimensional space where a linear decision surface is made due to its special properties which allows it to create a good generalizing model [Cortes and Vapnik, 1995].

To make this model more optimal and get a better classifier, the algorithm must find the hyperplane with the greatest margin. In order to make this optimal hyperplane, only a

small quantity of training data, which are called support vectors, are needed to decide this margin [Cortes and Vapnik, 1995]. An advantage of the SVM is that the objective function is convex, therefore the solution of the optimization is straightforward [Bishop, 2006]. The SVM is basically a two class classifier, although in practice we usually have problems with more than 2 classes. It is possible to combine multiple two-class SVMs to create a multiclass SVM classifier.

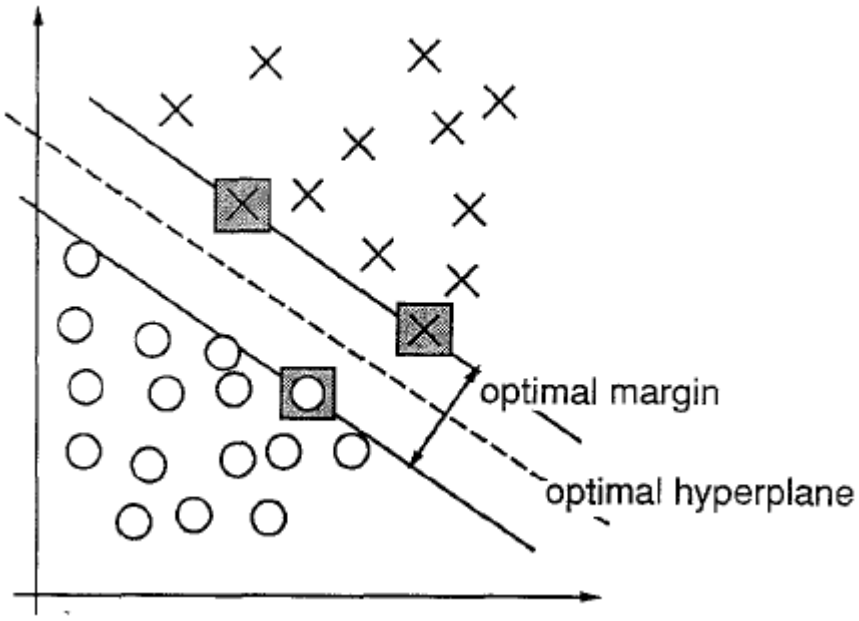

Figure 2.1: Classification with SVM in a 2 dimensional space. The support vectors, marked with grey squares, define the margin of largest separation between the two classes. (source: [Cortes and Vapnik, 1995])

2.3 Classification and regression trees (CART)

Tree-based models are a combinatory method where only one model makes predictions in any point in input space. It divides the space into rectangles, and places a model into each one [John Lu, 2010]. In order to create the model, given an input, a sequential making process in where the tree splits into two branches in each node. It is a sequence of binary decisions to single input variables [Bishop, 2006].
In illustration 2, we can see how the first input is divided in two different nodes depending on whether the parameter of the model (θ) is greater or not than X1. Furthermore, it can be seen that $X_1 < \theta_1$ is further divided depending on whether X2 is greater than

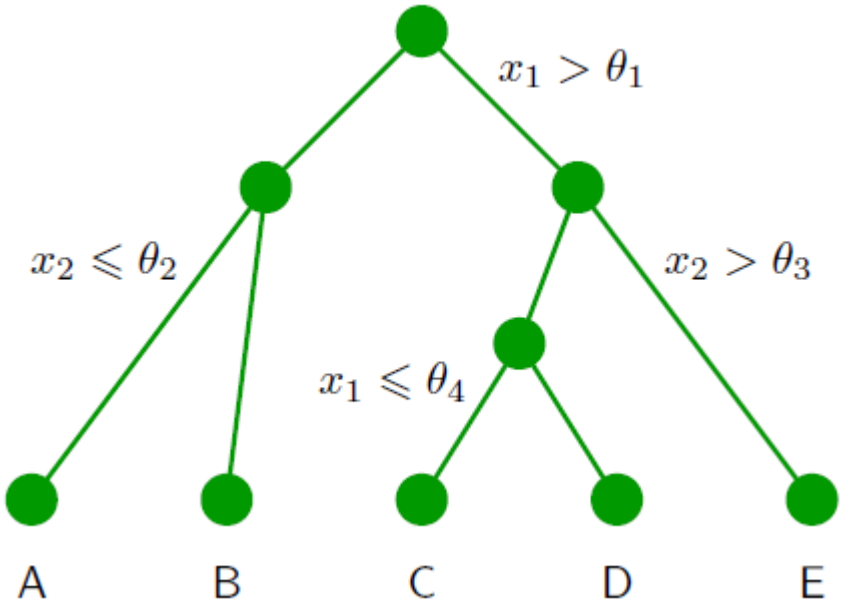

Figure 2.2: Binary tree with its corresponding partitioning of input space. (source: [Bishop, 2006])

X2, which forms two new regions: A and B. So for any new input, the tree will decide in which region it falls into, thanks to the decision criteria in each node. The reason why it is important in medical research is due to its simplicity and because it is easily interpretable. The feature space is fully explained with one single tree [John Lu, 2010]. The key factors to create a regression tree model is to determine the variables which will be used to make the binary decision, as well as the structure of the tree and the threshold for each node. Supposing that the model will partition into M regions R1, R2, . . . ,RM, and model the response as a constant c_m in each region:

$$f(x) = \sum_{m=1}^{M} c_m I(x \epsilon R_m). \tag{2.1}$$

Due to a high amount of combinatory solutions, the structure of the tree is created by forming one individual node at a time, and the choice of which variables to split, and the value of each threshold is determined by an exhaustive search. This search will find an optimal variable and threshold regarding the local average of the data, which will give the smallest sum-of-squares error [Bishop, 2006].

If we apply the criterion of minimizing the sum-of-squares error, $\sum(y_i - f(x_i))^2$, the optimal $\hat{c}_m$ is the average of y_i in region Rm:

$$\hat{c}_m = ave(y_i \mid x_i \epsilon R_m) \tag{2.2}$$

Afterwards, a greedy algorithm is used. Beginning with all of the data, the pair of half-planes is defined by:

$$R_1(j,s) = \{X|X_j \leq s\} \; and \; R_2(j,s) = \{X|X_j > s\} \tag{2.3}$$

And a splitting variable j and split point s is found that can solve:

$$min_{j,s}\left[min_{c_1}\sum_{x_1 \in R_1(j,s)}(y_i - c_1)^2 + min_{c_2}\sum_{x_1 \in R_2(j,s)}(y_i - c_2)^2\right] \tag{2.4}$$

For the chosen j and s, the calculation of the inner minimization is with:

$$\hat{c}_1 = ave(y_i \mid x_i \epsilon R_1(j,s)) \; and \; \hat{c}_2 = ave(y_i \mid x_i \epsilon R_2(j,s)) \tag{2.5}$$

For each splitting variable a split point s is determined and by searching all of the inputs the optimal pair (j, s) is found. Once the best split is found, the data is partitioned into two regions and the splitting process is done again on each of the subsequent two regions. This process is repeated on all of the regions [John Lu, 2010]. Afterwards, the algorithm must decide how big it should make the tree. A big tree will overfit the data, while a tree that is too small will not define the important structure. Tree size will determine the model's complexity, and it should be chosen depending on the data. An approach to determine the tree's size is to divide the tree nodes if the decrease in sum-of-squares exceeds some threshold. The problem with this approach is that a split that seems useless might lead to an even better split below it. Another approach is to find a larger tree "To" and stopping the splitting when a certain node size is reached. Afterwards, the tree can be pruned by a method called cost-complexity pruning. This method consists of defining a tree T obtained by pruning To. Terminal nodes are indexed by m, and Rm represents node m, and "T" defines the number of terminal nodes in T:

$$N_m = \#\{x_i \epsilon R_m big\} \tag{2.6}$$

$$\hat{c}_m = \frac{1}{N_m}\sum_{x_1 \epsilon R_m} y_i, \tag{2.7}$$

$$Q_m(T) = \frac{1}{N_m} \sum_{x_1 \epsilon R_m} (y_i - \hat{c}_m)^2, \tag{2.8}$$

And the cost complexity criterion is defined by:

$$C_\alpha(T) = \sum_{m=1}^{T} N_m Q_m(T) + \alpha|T|. \tag{2.9}$$

Then, for each α, the subtree T_α is found to minimize $C_\alpha(T)$.
The tuning parameter $\alpha \geq 0$ determines the tradeoff between tree size and how well it fits the data. Large values of α will give smaller trees T_α. If $\alpha = 0$, the full tree To will be given. Each α will give a unique T_α that minimizes C_α. To calculate α a five- or tenfold cross-validation is used, and a α is chosen which minimizes the cross-validated sum of squares. The final tree will be $T_{\hat{\alpha}}$ [John Lu, 2010].
One of the disadvantages of the tree regression model is that it is very sensitive to small changes in the training data [Bishop, 2006]. When a small change is made, it leads into a very different set of splits [John Lu, 2010] This is due to the hierarchical process of the model, an error in the top split will lead to errors on lower splits.

2.4 Neural networks

The creation of neural networks was historically motivated by the interconnected neurons in the brain [Reggia, 1993]. It can be defined as a network of processing elements that lead to global model behavior. Neural networks create linear combinations of the inputs and create a model as a nonlinear function of these features [John Lu, 2010]. It can be both used as a regression or a classification model. The elementary processing units are called neurons or nodes and they are divided into layers in a way that only units within two consecutive layers are connected [Larranaga et al., 2006]. The simplest neural network is called a perceptron. It consists of a one neutron classifier that uses a threshold activation function in order to separate two classes by a linear discrimination function.

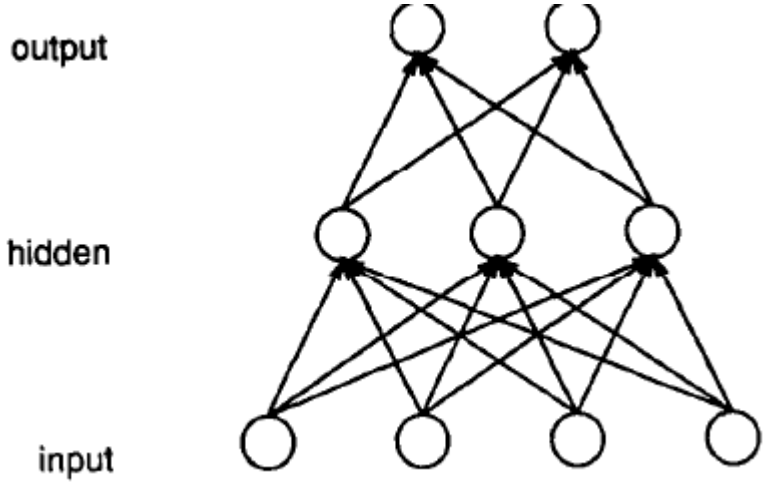

Figure 2.3: In this image a neural network with four input nodes, three hidden nodes, and two output nodes are shown. The connections run in the direction from input to hidden nodes and from hidden nodes to output nodes. (source: [Reggia, 1993])

Several perceptrons combined are called a multilayer perceptron. The output of the intermediate layers is sent only to the highest layer. A multilayer perceptron with only two hidden layers can approximate any classification problem [Larranaga et al., 2006]. In neural networks, each node ni is associated with a numeric activation level $a_i(t)$ and a time t [Reggia, 1993]. A node communicates its activation level to its surroundings at any point in time. The nodes receive the total input activation $in_j(t)$ which they use to update its own level of activation. Furthermore, each connection is associated with a weight w_{ji} and are also used to update node activations. If an activation of a node n_i tends to increase the activation of the neighbor node n_j, then the connection is called a excitatory link and is labeled by $w_{ji} > 0$. On the other hand, if an activation of a node n_i tends to decrease the activation of the neighbor node n_j, then the connection is an inhibitory link and is labeled by $w_{ji} < 0$. Another component of neural networks is the learning rule, which refers to modifications in the network due to its experiences over time. This learning rule describes how the weights on connections change over the function of time.

2.5 Literary review on modern application of predictive models

Basic prediction has been used in a variety of fields. For example, Nate Silver, founder and editor in chief of FiveThirtyEight, is a statistician that uses predictive models in order to predict the US elections with a high degree of success thanks to the compilation and analysis of political data to give the possible outcome of the upcoming elections [N., 2008]. Another example is Google finding out trends and fluctuations of the stock market [D., 2013] by finding the correlation between internet searches for a company's name and its trade volume, although it could not predict its price on the stock market. In the field of medicine, in the 1960s, diagnostic decision support systems were starting to be used with Bayesian applications. Several methods where proposed in order to remove the effect of redundant results [Miller, 1977]. In the 1990s, several papers discussed supervised and unsupervised [Hadzikadic, 1992] machine learning techniques applied in medicine. An example of a supervised technique is the neural network, formed by "active processing elements whose local interactions over time lead to global model behavior" [Reggia, 1993]. In [Baxt, 1992], Baxt uses neural networks to identify complex patterns and relationships that are difficult to find by a human. The application of neural networks to identify acute myocardial infarction in patients that go to the emergency room when they have anterior chest pain, has been shown to be more accurate than physicians. Heuristics have also been used as predictive analysis, by helping doctors make better decisions. With heuristics, the strategy is to ignore part of the information and focus on a couple key elements to take part in the prediction. Marewksi and Gigerenzer [Marewski and Gigerenzer, 2012] explain when heuristics can outperform other methods that use a lot more information. The method consists of asking a few yes-or-no questions, like simply looking for an anomaly in the patient's electrocardiogram, or for example if the patient is complaining of chest pain. Within the same line of investigation, a highly sensitive clinical decision rule was created in order to decide whether or not to perform a computer tomography (CT) scan on patients with minor head trauma, due to the fact that only a small percentage of patients get worse and require medical intervention. If that is the case, then an early diagnosis of intracranial hematoma by CT and the following surgery is necessary. In [Kline et al., 2008] decision rules were used in order to decide if a pulmonary embolism (PE) test should be performed on patients with low risk factors due to the fact that experts suggest that PE is still often missed at a high rate. Another popular use of predictive analysis is the development of machine learning methods in order to predict mortality in patients initially diagnosed with pneumonia. This is useful in order to give

the doctor an idea of whether the patient is suitable to go home or should stay in the hospital to receive further intensive care [Cooper et al., 1997].

Chapter 3

Methodology

3.1 Dataset description

The German electronic health records contained several inputted data of different attributes which were studied in order to find predictors. Table 3.1 shows the most important attributes from the electronic health records.

Event type	Attribute
Information about the patient	Case ID, Patient ID, Age, Gender, Diagnosed code, Hospital cost
Admission to the hospital	Case ID, Admission date, Discharge date, Department code
Procedure	Case ID, Date of the issued procedure, Department in which the procedure is made, Procedure code
Discharge	Case ID, Discharge date, Department code

*Case ID is defined by a unique ID to identify patients.

Table 3.1: Event types and attributes from the electronic health records

The data extracted from the German electronic health records in order to make predictions with Machine Learning consists of five attributes in order to predict the length of stay of the patients:

- The first two attributes consist of the age and the gender of the patient.
- The next attribute is the ICD code. The ICD is the International Classification of Diseases and Related Health Problems. It was created by the World Health

Organization (WHO) [World-Health-Organization, 2018]. Its purpose is to allow different countries to share health information using a common language. The ICD lists all the possible diseases, disorders, injuries and other related health conditions. This list is organized by categories and by group of diseases. The ICD codes included in the database have the German version 10 (ICD-10). In order to simplify the ICD-10 codes that were entered in the electronic health records, all the codes were classified into its main chapters. Table 3.3 defines all the main categories that were used in this thesis to classify the pacient ICD code. The ICD-10 code uses its first three characters to determine the category of the diagnostic, and the next three characters to give further details such as anatomical site or severity and the last character is used for expansion.

- The next attribute is the number of procedures that the patient has undergone during the hospital stay. A higher number of procedures could indicate a higher length of stay.
- The final predictor is the day of the week that the patient is admitted to the hospital. According to [Carter and Potts, 2014] discharges on a Saturday or Sunday are improbable due to medical personal with less experience that work during the weekend, so it could lead to an increased length of stay.

Patient Identification Number	**Age**	**Gender**	**ICD category**	**Number of procedures**	**Day of the week admitted**	**LOS**	**LOS group**
Patient 1	11	Male	18	5	Wednesday	6	Short-term
Patient 2	36	Female	11	1	Monday	1	Short-term
Patient 3	61	Male	2	10	Wednesday	7	Short-term
Patient 4	4	Female	19	1	Monday	1	Short-term
Patient 5	73	Female	2	7	Monday	70	Long-term

Table 3.2: Example datapoints extracted from the electronic health records for the prediction of the lenght of stay

For the classification models, the patients were divided into two groups, according to the real LOS. Group 1 stayed less than 30 days and group 2 stayed 30 days or more.
The term long-term patients is defined by patients staying in a hospital for 30 days or more. The number of long-term patients in a hospital is an indicator used in hospitals

because shorter lengths of stay are related with increased income to the hospital because it increases the hospital turnover rate [Baek et al., 2018].

Chapters	Blocks	Title
I	A00–B99	Certain infectious and parasitic diseases
II	C00–D48	Neoplasms
III	D50–D89	Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism
IV	E00–E90	Endocrine, nutritional and metabolic diseases
V	F00–F99	Mental and behavioural disorders
VI	G00–G99	Diseases of the nervous system
VII	H00–H59	Diseases of the eye and adnexa
VIII	H60–H95	Diseases of the ear and mastoid process
IX	I00–I99	Diseases of the circulatory system
X	J00–J99	Diseases of the respiratory system
XI	K00–K93	Diseases of the digestive system
XII	L00–L99	Diseases of the skin and subcutaneous tissue
XIII	M00–M99	Diseases of the musculoskeletal system and connective tissue
XIV	N00–N99	Diseases of the genitourinary system
XV	O00–O99	Pregnancy, childbirth and the puerperium
XVI	P00–P96	Certain conditions originating in the perinatal period
XVII	Q00–Q99	Congenital malformations, deformations and chromosomal abnormalities
XVIII	R00–R99	Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified
XIX	S00–T98	Injury, poisoning and certain other consequences of external causes
XX	V01–Y98	External causes of morbidity and mortality
XXI	Z00–Z99	Factors influencing health status and contact with health services
XXII	U00–U99	Codes for special purposes

Table 3.3: Classification by chapters of the diseases in ICD-10 [WHO, 2010].

3.2 Data processing

Log data recorded between April 2012 and December 2015 were extracted from the electronic health records (EHR) of a hospital in Germany to find factors that could help predict the length of stay of a patient. This accounted for a total of 127265 patients that

were recorded in this period. The patients were admitted and discharged between 2012 and 2015. A total of 122 patients that were in the hospital at the end of the year of 2015 were excluded from the study. Furthermore, only patients that were assigned with a single code of the International Classification of Disease (ICD-10) in the primary data set were included for analysis, in order to obtain the LOS of patients that is assumed to be an uncomplicated admission. Additionally, a total of 8979 day cases, which are defined as patients formally admitted for a medical procedure or surgery in the morning and discharged within the same day, were also excluded from the analysis. Furthermore, only cases assigned to the pay area of DRG were taken into account in this study. All cases within specialized departments such as psychiatric institutions and psychosomatic institutions, such as PSY or PIA where excluded from this study. The reason of the encounter was also taken into account, and all patients that were newborns or were included as an accompanying person or caregiver were not included. In a nutshell, a total of 18804 patient encounters were used for the analysis. To create the machine learning models, the data was partitioned using 80% for training data in order to learn the model with the remaining 20% used to test the model to make sure the model does not overfit the data. Table 3.4 summarizes the number of cases used for each step of the process of creating the models.

Number of cases used for the training of the model (80%)	15043
Number of cases used for the validation of the model (20%)	3761
Total number of cases used for the analysis	18804

Table 3.4: Number of cases used to create the machine learning models

3.3 Descriptive and univariate analyses

In this section an exploratory and a statistical analysis of the predictors and the length of stay will be made in order to understand from multiple perspectives what could affect the LOS in the hospital. The gender of the patient was studied because there are specific diseases that a specific gender is more prone to suffer from than the other one and viceversa. The patients in this study were evenly distributed with gender. 49,24% of subjects were male and 50,76% were female. Male patients stayed an average of 6,18 days in the hospital, while female patients stayed 7,58 days on average.

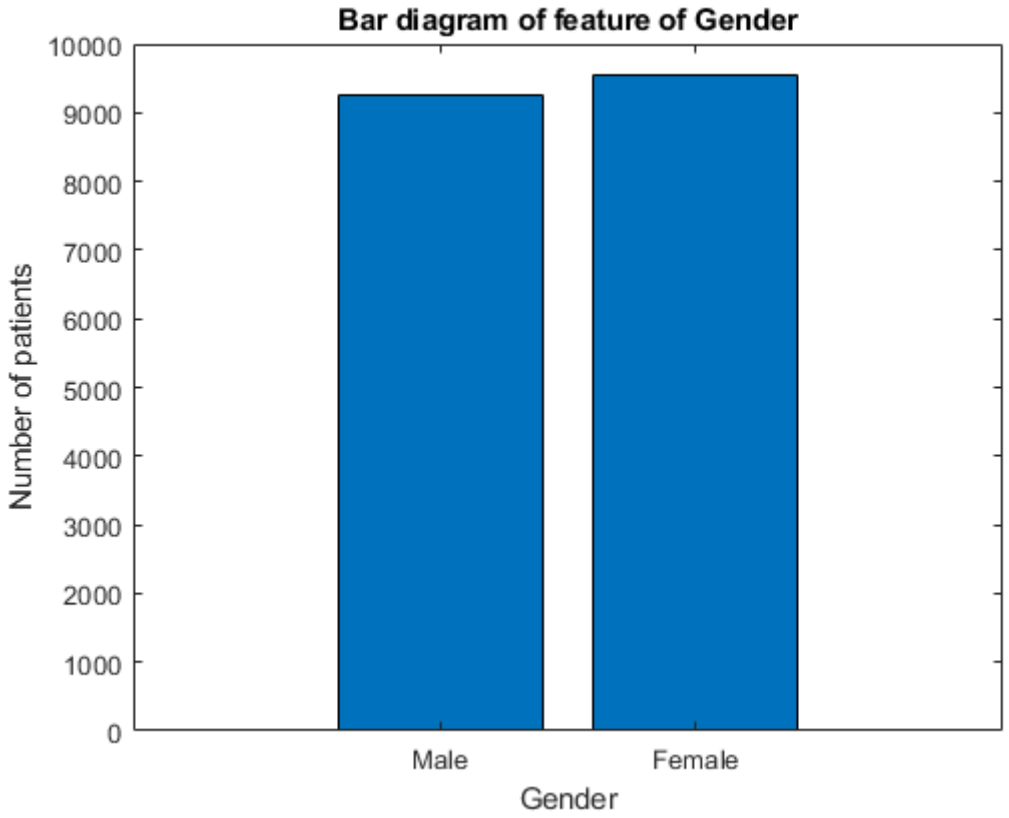

Figure 3.1: This image shows the distribution of gender of the inpatients included in the study. It accounts for a total of 9259 (49,24%) male patients and a total of 9545 (50,76%) of female patients)

Figure 3.2: This boxplot presents the median age of all the patients of about 26. The age range was from 0 to 97 years old.

The average age of the hospitalized patients was 30 years old. The average number of procedures while the patients were in the hospital was 2,38 days, with a range from 0 to 58 procedures.

Variable	**Mean**	**Median**	**Std**	**IQR**	**Min**	**Max**
Age	30,08	26	24,44	8-57	0	97
Number of procedures	2,38	1	4,92	0-2	0	58

Table 3.5: Statistics of the age and the number of procedures received by the inpatients

Age was considered a predictor of the LOS, as explained by several research papers, such as [Smith et al., 2008] and [Carter and Potts, 2014]. In Table 3.6 it can be seen that the average number of LOS increases as the age of the patients also increases.

Age	# patients	Age		Length of stay in days					
		Mean	Median	Mean	Med.	Std	IQR	Min	Max
0-10	5493	3,23	2	2,42	2	2,59	1-3	1	76
11-20	2889	15,05	15	3,22	2	3,61	1-4	1	71
21-30	1963	25,46	25	4,6	3	6,73	2-6	1	84
31-40	1786	35,21	35	6	3	11,64	2-6	1	91
41-50	2112	45,83	46	9,16	3	17,02	2-6	1	91
51-60	1856	55,15	55	11,96	4	75	2-8	1	91
61-70	1360	65,19	65	14,14	4	23,72	2-10	1	91
71-80	1052	74,87	75	18,75	4	28,91	2-15,25	1	91
81-90	266	84,34	84	23,66	5	34,32	2-32,5	1	91
91-100	27	93,03	93	14,33	4	25,24	2-8	1	88
All patients	18804	30,08	26	6,89	2	14,17	1-5	1	91

Table 3.6: Statistics of the length of stay categorized by age

The mean length of stay for all patients was of approximately 7 days. The mean length of stay increased on each age group. This can be interpreted as that the older the patient, the higher their chance to be a long term patient. It can be seen that between the age of 0 and 40 the length of stay stays relatively constant, but after the age of 40 the length of stay increases at a much higher rate. The age group with the highest mean of length of stay was of the people aged 81-90, having a mean LOS of 23,66. Furthermore, each age group has a lot of outliers as it can be seen in Figure 3.3.

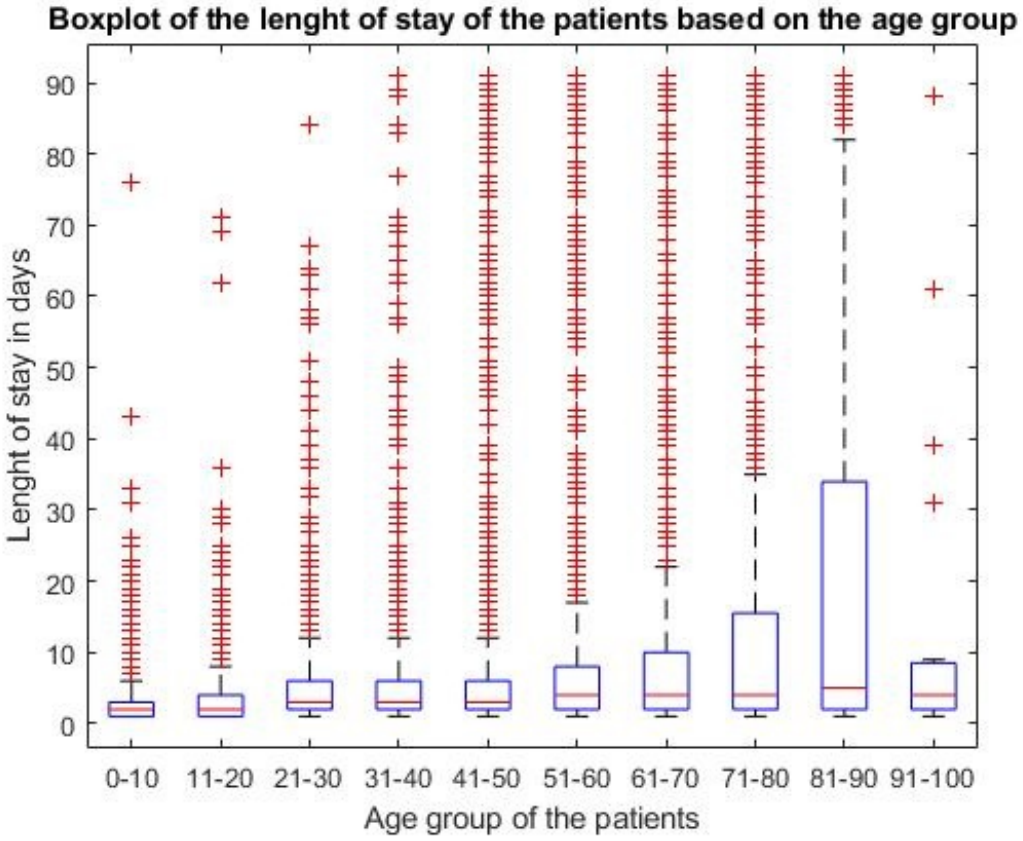

Figure 3.3: This boxplot presents the length of stay of the patients based on the age group categorized by the decade.

The length of stay was analyzed based on ICD-10 codes. The diagnosis given to the patient played a big role relating to how long the patient would stay in the hospital. Table 3.7 gives the statistics of the length of stay based on the category of the diagnosis given to the inpatients. Diagnoses within categories XXI (Factors influencing health status and contact with health services), II (Neoplasms), III (Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism) and XII (Diseases of the skin and subcutaneous tissue) give a much higher average LOS than the other categories. The top three categories with the greatest frequency of diagnosis was category XIX (injury, poisoning and certain other consequences of external causes) with 2707 patients diagnosed, followed by category II (neoplasms) which had 2553 patients diagnosed and category XI (diseases of the digestive system) with 2152 patients diagnosed.

Chapters	Blocks	# of Patients	Length of stay in days				
			Mean	Median	IQR	Min	Max
I	A00–B99	403	3,67	3	2-5	1	33
II	C00–D48	2553	19,04	5	3-27	1	89
III	D50–D89	105	11,06	3	2-5	1	84
IV	E00–E90	275	5,14	3	2-8	1	31
V	F00–F99	289	1,6	1	1-2	1	12
VI	G00–G99	641	1,96	1	1-2	1	12
VII	H00–H59	99	2,3	2	1-3	1	7
VIII	H60–H95	484	4,95	6	3-6	1	10
IX	I00–I99	749	1,83	1	1-2	1	15
X	J00–J99	1661	3,82	3	2-6	1	12
XI	K00–K93	2152	2,68	2	1-3	1	15
XII	L00–L99	1186	9,54	6	3-14	1	76
XIII	M00–M99	450	5,41	3	2-5,75	1	66
XIV	N00–N99	1633	2,72	2	1-4	1	15
XV	O00–O99	908	4,92	3	2-6	1	63
XVI	P00–P96	178	3,43	3	2-4,75	1	16
XVII	Q00–Q99	554	3,43	2	2-4,75	1	43
XVIII	R00–R99	1105	1,84	1	1-2	1	9
XIX	S00–T98	2707	2,477	1	1-3	1	36
XX	V01–Y98	0	0	0	0	0	0
XXI	Z00–Z99	672	37,31	5	1-88	1	91
XXII	U00–U99	0	0	0	0	0	0

Table 3.7: Length of hospital stay by chapters of the ICD-10

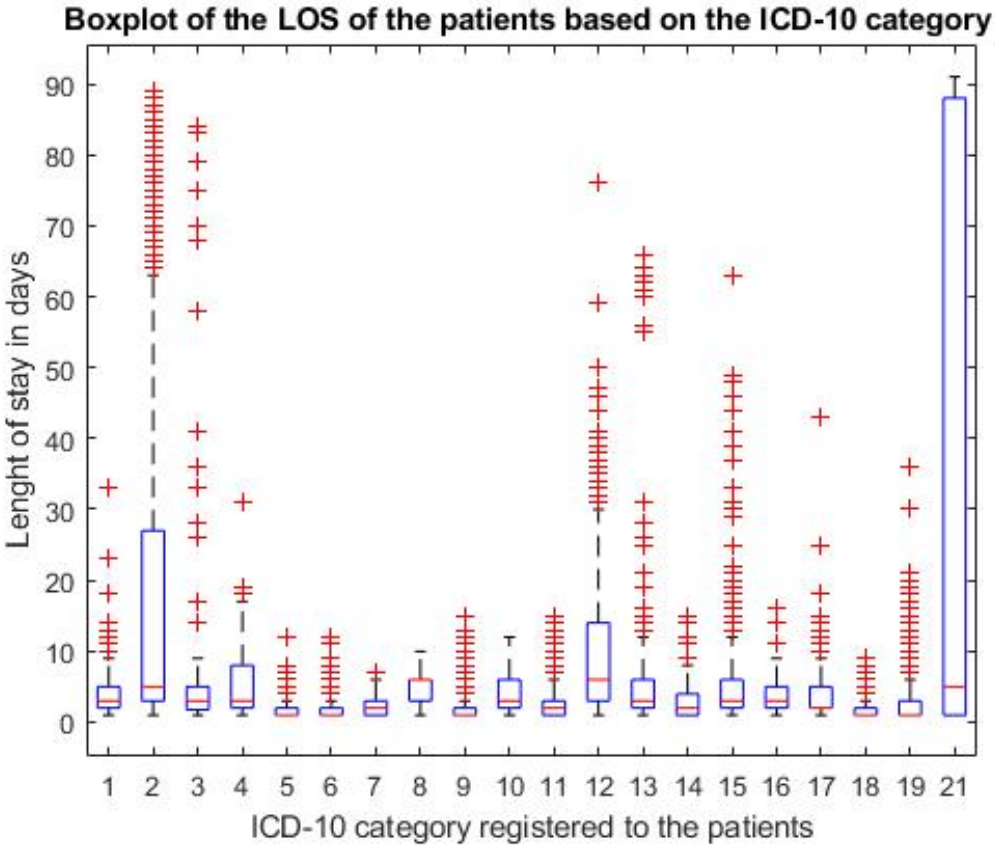

Figure 3.4: This boxplot presents the length of stay in days based on the chapters of the ICD-10 which were assigned to the patients.

Hospital stay based on the number of procedures showed that the patients that had undergone a higher number of procedures (6 or more) have a much higher length of stay (34,93 days on average and IQR 5-71) than the patients that had no procedures (3,75 days on average and IQR 1-3), one procedure (3,67 days on average and IQR 1-5), 2 procedures (4,04 days on average and IQR 1-5), 3 procedures (5,84 days on average and IQR 2-6), 4 procedures (8,93 days on average and IQR 2-7) or 5 procedures (9,42 days on average and IQR 2-7).

Number of procedures	# of patients	Length of Stay in days					
		Mean	Median	Std	IQR	Min	Max
0	5608	3,75	2	9,07	1-3	1	88
1	5571	3,67	2	5,54	1-5	1	86
2	2968	4,04	3	4,79	1-5	1	64
3	1571	5,84	3	9,88	2-6	1	71
4	1006	8,93	4	15,87	2-7	1	88
5	577	9,42	4	17,02	2-7	1	87
6 or more	1503	34,93	19	33,89	5-71	1	91
Total	18804	6,89	2	14,17	1-5	1	91

Table 3.8: Length of hospital stay by number of procedures performed

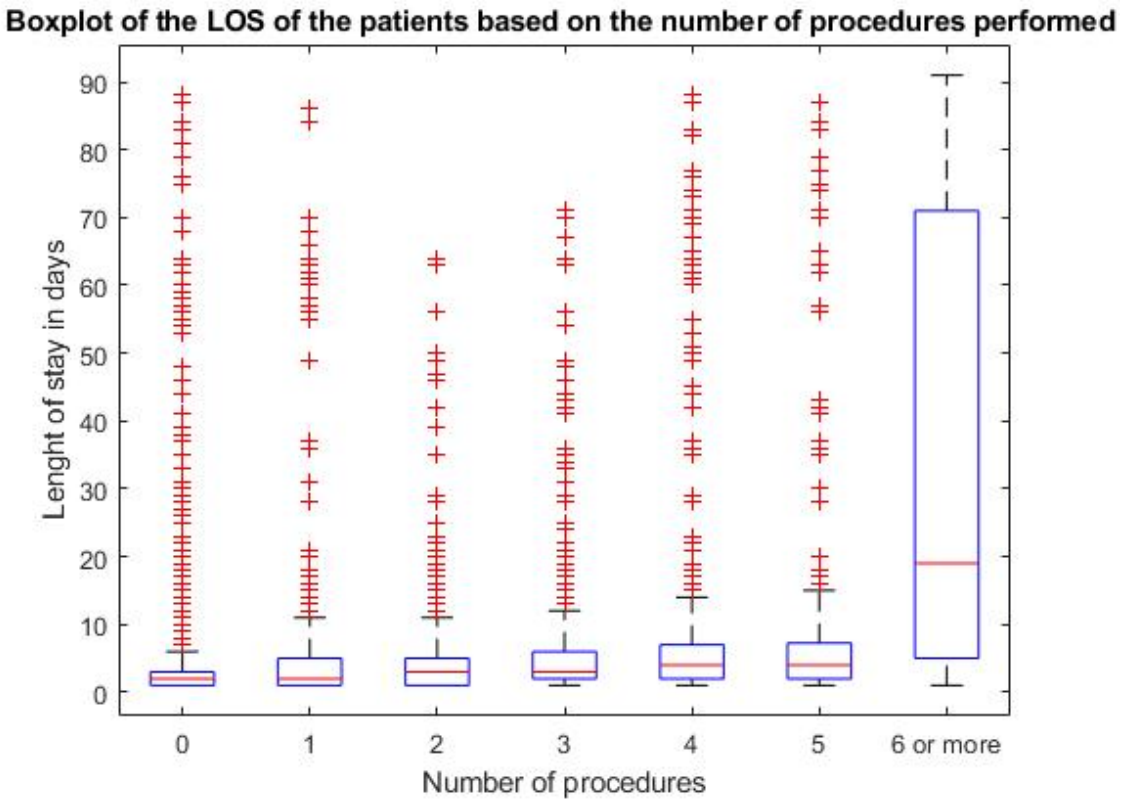

Figure 3.5: This boxplot presents the length of stay in days based on the number of procedures performed on the patients.

The day of being admitted has a great impact on how many days the patient will stay in the hospital. A patient who is admitted on a Monday, Tuesday or Thursday will have a

similar LOS in the hospital. However, if the patient is admitted on a Wednesday, the LOS increases by about 2 days, when compared to a patient admitted on a Monday. Patients admitted on the weekend have an average LOS of about 3 days, 4 days less than if they would be admitted on a Monday. It is also worth mentioning that about 3 times more patients are admitted in the hospital on a Monday (3848 patients) than on a Saturday (1214 patients). This has to do with how the hospital operates, due to the fact that more qualified personnel work during the week than during the weekends.

Day of the week admitted to the hospital	**# of patients**	**Length of Stay in days**					
		Mean	**Median**	**Std**	**IQR**	**Min**	**Max**
Monday	3848	7,05	3	14,91	1-5	1	91
Tuesday	3197	6,65	2	14,08	1-6	1	91
Wednesday	3526	8,98	3	19,26	2-6	1	91
Thursday	3046	7,12	2	15,28	1-5	1	90
Friday	2420	8,08	3	17,23	1-5	1	89
Saturday	1214	3,1	2	4,97	1-4	1	87
Sunday	1553	2,88	2	2,33	1-4	1	22
Total	18804	6,89	2	14,17	1-5	1	91

Table 3.9: Length of hospital stay depending on the day of the week admitted to the hospital

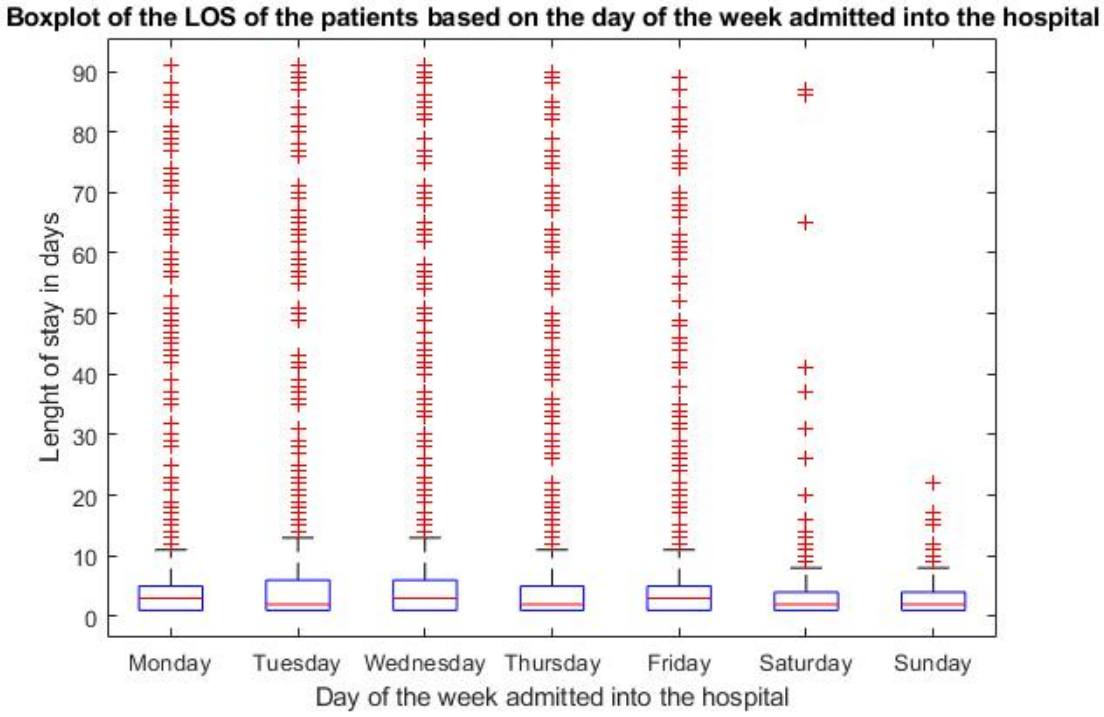

Figure 3.6: This boxplot presents the length of stay in days based on the day of the week the patient was admitted into the hospital.

As seen in Table 3.10, the majority of patients stayed less than 30 days in the hospital, while 980 patients stayed 30 days or longer. Long-term patients had a significantly higher number of procedures received (14,48 procedures on average) than short-term patients that stayed under 7 days (1,42 procedures on average).

Variables	**A (under 7 days)**	**B (7 - 30 days)**	**C (30 days or more)**
Number of patients	15550	2274	980
Average LOS in days	2,46	11,32	66,88
Procedures per patient	1,42	3,69	14,48

Table 3.10: Comparison of the three different categories of length of stay in the hospital

In order to test the independence of the variables with the outcomes, the t-test was performed. It can be said that the variables have a relationship with the prediction when the probability of the test statistic is less or equal to the probability of the alpha error rate. In that case the null hypothesis is rejected. All variables were found to be significant ($p<0,005$): age ($p<0.001$), gender ($p<0.001$), ICD category ($p<0.001$), entrance day ($p<0.001$), and number of procedures ($p<0,001$).

Chapter 4

Thesis results

4.1 Hardware and software used

All calculations for this thesis have been made from a Windows operating system. All models were worked on with Matlab [MathWorks, 2019]. Matlab is a multi-paradigm numerical computing environment, and it uses matrix calculations. The version of Matlab used in this thesis is the R2018b.

4.2 Performance metrics

The purpose of using performance metrics is to find out how effective a model is. This section will discuss the different performance metrics applied both to regression and for classification. The performance metrics that will be used in this thesis for regression are the mean absolute error, the root mean square error and the coefficient of determination. In addition, in order to evaluate the classification models, a confusion matrix will be used along with the metrics that are derived from it, such as: True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). From these metrics, four evaluation measures will be calculated: Accuracy, Precision, Recall and F-1 score.

4.2.1 Performance metrics for the regression models

The evaluation metrics used for the regression models are the root mean square error (RMSE), the mean absolute error (MAE) and the coefficient of determination R^2 . Both RMSE and MAE have been used to evaluate performance in previous models. The difference is that MAE gives the same weight to all errors and the RMSE penalizes variance because it gives errors with larger absolute values more weight than errors with smaller absolute values [Chai and Draxler, 2014]. RMSE can never be smaller than MAE. MAE

averages the absolute values of the residuals while RMSE is the square root of the MSE. Both RMSE and MAE have the same magnitude as the variable being predicted [Lin et al., 2018].
The MAE and RMSE are calculated as

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|e_i| \tag{4.1}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}|e_i^2|} \tag{4.2}$$

where we assume we have n samples of the models errors (e_i,i=1,2,3...,n).
R^2 is defined as the proportion of variance explained by the regression model and its used to check how well the model can predict the dependent variable with the dependent models. It is dimensionless and does not depend on the units used [Nagelkerke et al., 1991]. A R^2 of 0 is considered to signify that the independent variables cannot predict the dependent variable, and a R^2 of 1 means that the dependent variable can be predicted without error. Any variable of R^2 between 0 and 1 gives the extent of how much a dependent variable is predictable. A R^2 of 0,8 means that 80% is predictable. R^2 is expressed as:

$$R^2 = 1 - MSE/Var(y) \tag{4.3}$$

With Var(y) being the variance of the value.

4.2.2 Performance metrics for the classification model

In order to find the success of prediction of a model of binary classification, the confusion matrix will be used, in which it contains information of true positives, false positives, true negatives and false negatives. The confusion matrix presents the ways in which the model gets confused when predicting. These four metrics are then used to predict different ratios to show the performance level of the model [Visa et al., 2011].

- True positives (TP) give the number of correctly predicted positive values.
- False negatives (FN) give the number of data points where the actual class predicted was true and it was predicted as false.
- False positives (FP) give the number of data points where the actual class predicted was false and it was predicted as true.
- True negatives (TN) give the number of correctly predicted false values.

In this master's thesis the confusion matrix will have a two by two layout like the one showed in Table 4.1 which will consist of two classes: positive and negative. TP and TN are considered to be instances where the model correctly predicted if a patient will stay long-term in the hospital or not, while FP and FN are instances in which the model incorrectly predicted long-term or short-term patients respectively. In order to have a high classification accuracy the rate of TP and TN should be high, while the rates of FP and FN should be relatively low. It is important that both FP and FN are low because it can lead to false decision making in clinical care.

		Predicted	
		Negative	Positive
True	Negative	True Negatives (TN)	False Positives (FP)
	Positive	False Negatives (FN)	True Positives (TP)

Table 4.1: Confusion matrix table.

The first ratio that will be calculated with the confusion matrix will be the accuracy, which gives a percentage of how often the classifier is correct:

$$Accuracy = \frac{TruePositives + TrueNegatives}{Total} \tag{4.4}$$

The next ratio is the precision, which gives the percentage of the data points that the model says are relevant compared to those that are actually relevant.

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{4.5}$$

The third ratio calculated will be the Recall, which is the ability of the model to detect the relevant cases within the data points.

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{4.6}$$

The final ratio is called the F-1 score and it is defined as the weighted average of precision and recall. The F-1 score gives equal weight to both precision and recall, and a model with optimal balance of both these metrics will have a high F-1 score [Goutte and Gaussier, 2005].

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4.7}$$

Another metric used will be the receiver operating characteristic (ROC) and the area under the ROC curve (AUC). The ROC curve plots sensitivity as a function of commission

error (1 – specificity) and it encapsulates model performance over all conditions a model could operate in by using all the data the model provides [Lobo et al., 2008]. AUC is a discrimination index that gives the probability that a presence will have a higher predicted value than an absence.

4.3 Definition of the parameters of the CART models

The model was partitioned using 80% for training data in order to learn the model with the remaining 20% used to test the model to make sure the model does not overfit the data.

When creating the tree, its complexity must be defined because it has a great effect on the accuracy of the model. The tree complexity is controlled by the stopping criteria used and the pruning method employed. Furthermore, the tree complexity is defined by the total number of nodes, total number of leaves, the tree depth and how many attributes were used. CART consists of two phases: growing and pruning. The tree is produced in a top-down recursive approach and each node divides the set into smaller subsets until a stopping criteria is specified or there is no sufficient gain in information based on a splitting measure [Rokach and Maimon, 2005]. A complex tree with many leaves is usually very accurate with the training data but it will not be as accurate on a different test set. This is due to the fact that it is overtrained (or overfit). On the other hand, if the tree is not that complex, it will not attain a very high accuracy, even though the training accuracy will be similar to the independent test set [Math-Works, 2018]. Both the classification and regression tree models have three parameters that can control the complexity of the tree in Matlab. The first parameter is the "maximal number of branch node splits", the second parameter is the "minimum leaf size" (a small value of this parameter will give deep trees), and the third parameter is the "minimum parent size". Another method to control the tree depth is by pruning [Rokach and Maimon, 2005]. It was created to solve the dilemma of whether it was better to overfit or underfit a tree. Pruning works by letting the tree overtrain the model by not being very strict with the stopping criterion and then "cutting" the tree into a smaller tree, removing branches that do not contribute much information and therefore not adding much accuracy to the model. The optimal pruned tree is therefore the tree that achieves minimum cost on test data. It is recommended that the chosen pruned tree will be the "1 SE" tree which is the smallest tree with an estimated cost within 1 standard error of the minimum cost (or "0 SE" tree) [Steinberg and Colla, 2009].

4.4 Results of the classification tree

The performance of the classification tree will be presented in this section. The purpose of the classification tree is to predict whether a patient will be a long-term patient (30 days or longer in the hospital) or a short-term patient (less than 30 days in the hospital). A classification tree is performed with the variables age, gender, ICD code, number of procedures and day of the week that the patient was accepted. In the classification tree, the "yes" prediction will be of the patients that are long-term patients, while the "no" prediction will be of patients staying in the hospital for less than 30 days. A confusion matrix, its associated ratios, the ROC and AUC will be calculated for both the training dataset and for the test dataset in order to know if the model overtrains the data or not.

4.4.1 Results of the classification tree without pruning

In this section a classification tree is performed with the default parameters of Matlab. In order to check the performance of the algorithm, the classification tree model was first tested on the training dataset and later on the test dataset. This is done in order to see if the model overfits the data or not. The tree consists of 24 levels.

Classification tree without pruning		
Total number of nodes	**Number of levels**	**Total computational time**
335	24	10 seconds

Table 4.2: Total number of nodes, levels and computational time of the classification tree without pruning.

Results of the classification tree on the training dataset

In Table 4.3 the confusion matrix for the classification tree on the training dataset is shown.

		Predicted class(from classifier)	
	Total=15044	**Not a long-term patient**	**Long-term patient**
True class	**Not a long-term patient**	14192(TN)	75(FP)
	Long-term patient	166(FN)	611(TP)

Table 4.3: Confusion matrix of the classification tree of the training dataset.

The model correctly predicted 14803 instances, while it incorrectly predicted a total of 241 instances.

ROC AUC	Accuracy	Precision	Recall	F-1 Score
0.9898	0,98398032	0,89067055	0,78635779	0,83137725

Table 4.4: Decision tree classification performance of the training dataset.

Table 4.4 shows the classification tree performance on the training data and it can be seen that the accuracy has a score of 0,98 while precision got a score of 0,89. Precision measures the percentage of the positive values predicted by the model that are actually positive. Therefore, the model obtaining a precision of 0,89 can be interpreted as 11% of patients being classified as short-term patients, when in reality they were going to be long-term patients. In a similar way, a recall of 0,78 can be interpreted as the model predicting 22% of patients as short-term stay when they should have been predicted as long-term patients. The F-1 score is of 0,83 and the AUC is 0,98.

Results of the classification tree on the test dataset

In table 4.5 the confusion matrix for the classification tree on the test dataset is shown.

		Predicted class(from classifier)	
	Total=3760	**Not a long-term patient**	**Long-term patient**
True class	**Not a long-term patient**	3495(TN)	62(FP)
	Long-term patient	71(FN)	132(TP)

Table 4.5: Confusion matrix of the classification tree of the test dataset.

The model correctly predicted 3627 instances, while it incorrectly predicted a total of 133 instances.

ROC AUC	Accuracy	Precision	Recall	F-1 Score
0.9519	0,96462766	0,68041237	0,65024631	0,66466165

Table 4.6: Decision tree classification performance of the test dataset.

Table 4.6 represents the classification tree performance on the test data where it shows that the overall accuracy is 0,96 while precision got a score of 0,68. The classification tree obtaining a precision of 0,68 can be interpreted as 32% of patients being classified as

short-term patients, when in reality they were going to be long-term patients. In a similar way, a recall of 0,65 can be interpreted as the model predicting 45% of patients as short-term stay when they should have been predicted as long-term patients. The F-1 score is of 0,66 and the AUC is 0,95. As it can be seen, the performance of the classification tree is worse on the test set than with the training set. This is due to the fact that the model is overtrained so it cannot classify as effectively new data that the model has not "seen" beforehand.

4.4.2 Results of the classification tree with pruning

In this section, the previous classification tree will be "pruned" in order to reduce the overall size of the tree by removing parts of it that have little prediction power. The technique of pruning will also make the classifier less complex and will help increase the accuracy of the final classifier by reducing overfitting. As stated beforehand, the chosen pruned tree is the smallest tree with an estimated cost within 1 standard error of the original tree. It was calculated that the optimal pruning of the tree is from 24 levels that the original tree has to only 8 levels. Therefore, the pruned tree was formed by pruning a total 16 levels from the 24 levels of the original regression tree.

Classification tree with pruning		
Total number of nodes	**Number of levels**	**Total computational time**
47	8	8 seconds

Table 4.7: Total number of nodes, levels and computational time of the classification tree with pruning.

Results of the classification tree on the training dataset

In Table 4.8 the confusion matrix for the classification tree on the training dataset is shown.

		Predicted class (from classifier)	
	Total=15044	**Not a long-term patient**	**Long-term patient**
True class	**Not a long-term patient**	14130(TN)	125(FP)
	Long-term patient	265(FN)	524(TP)

Table 4.8: Confusion matrix of the classification tree from the training dataset.

The model correctly predicted 14654 instances, while it incorrectly predicted a total of 390 instances.

ROC AUC	Accuracy	Precision	Recall	F-1 Score
0.964	0,97407604	0,80739599	0,66413181	0,72328767

Table 4.9: Decision tree with pruning classification performance of the training dataset.

Table 4.9 shows the classification tree performance on the training data. It can be seen that the most of the classification performances are better than the unpruned classification tree shown in Table 4.4. This is because the previous model overlearned the training dataset.
The overall accuracy is 0,97 and the precision got a score of 0,80. The classification tree obtaining a precision of 0,80 can be interpreted as 20% of patients being classified as short-term patients, when in reality they were going to be long-term patients. In a similar way, a recall of 0,66 can be interpreted as the model predicting 44% of patients as short-term stay when they should have been predicted as long-term patients. The F-1 score is of 0,72 and the AUC is 0,96.

Results of the classification tree on the test dataset

In Table 4.10 the confusion matrix for the classification tree on the test dataset is shown.

		Predicted class (from classifier)	
	Total=3760	**Not a long-term patient**	**Long-term patient**
True class	**Not a long-term patient**	3527(TN)	42(FP)
	Long-term patient	67(FN)	124(TP)

Table 4.10: Confusion matrix of the classification tree from the test dataset.

The model correctly predicted 3651 instances, while it incorrectly predicted a total of 109 instances.

ROC AUC	Accuracy	Precision	Recall	F-1 Score
0.9698	0,97101064	0,74698795	0,64921466	0,69208633

Table 4.11: Decision tree with pruning classification performance of the test dataset.

Table 4.11 shows the classification tree performance on the test data. This model performs better on the test data than the unpruned classification tree due to the fact of overtraining

of the previous model. It is clear that the pruned tree is a better classifier because its not as complex and because it performs better on new dataset from which the classifier has not learned.
The overall accuracy is 0,97 and the precision got a score of 0,74. The classification tree obtaining a precision of 0,74 can be interpreted as 26% of patients being classified as short-term patients, when in reality they were going to be long-term patients. In a similar way, a recall of 0,64 can be interpreted as the model predicting 46% of patients as short-term stay when they should have been predicted as long-term patients. The F-1 score is of 0,64 and the AUC is 0,96.

4.5 Results of the regression tree

The performance of the regression tree will be presented in this section. The regression model is performed to predict the actual number of days the patient will stay in the hospital. The model will use the variables age, gender, ICD code, number of procedures and day of the week that the patient was accepted. The performance metrics used for the regression tree will be the mean absolute error, the root mean square error and the coefficient of determination. The model will be tested for both the training dataset and for the test dataset in order to know if the model overtrains the data or not.

4.5.1 Results of the regression tree without pruning

In this section a regression tree is created with the default parameters of Matlab. In order to check the performance of the regression model, it was first tested on the training dataset and later on the test dataset. This is done in order to see if the model overfits the data or not. The tree consists of a total of 95 levels.

Regression tree without pruning		
Total number of nodes	**Number of levels**	**Total computational time**
4209	95	9 seconds

Table 4.12: Total number of nodes, levels and computational time of the regression tree without pruning.

Results of the regression tree on the training dataset

In Table 4.13 the results of the regression tree on the training dataset is shown. The model received a mean absolute error of 2,49, a root mean square error of 9,06 and a coefficient of determination of 0,66.

MAE	RMSE	R-squared
2,493	9,0655	0,6665

Table 4.13: Performance metrics of the regression tree without pruning on the training dataset.

Results of the regression tree on the test dataset

In Table 4.14 the results of the regression tree on the test dataset is shown. The model received a mean absolute error of 3,82, a root mean square error of 11,16 and a coefficient of determination of 0,53. It is clear that the regression tree model is overtrained due to the increased errors in the test dataset compared to the training dataset.

MAE	RMSE	R-squared
3,821	11,162	0,5311

Table 4.14: Performance metrics of the regression tree without pruning on the test dataset.

4.5.2 Results of regression tree with pruning

Pruning was performed on the original regression model in order to prevent it from overfitting the data. Regression trees use hold-out pruning [Kitts, 1997]. The tree first finds all nodes whose leaves only have terminals. However, if these nodes have a leaf that is a terminal and another which are subtrees, it cannot be removed. The algorithm then proceeds to remove the leaves with only terminals, and it checks the overall error. If the error improves or stays the same with the reduced tree, then these leaves will be removed. Then the algorithm continues the same process until no parent removal improve the error. The acceptable error chosen for the pruned tree is of 1 standard deviation of the original tree. The tree was pruned and reduced 60 levels. Therefore, it was reduced from 95 levels that the original tree had to only 35 levels. The algorithm took 1 minute and 6 seconds to find the best pruning level.

Pruned regression tree		
Total number of nodes	**Number of levels**	**Total computational time**
227	35	1:06 minutes

Table 4.15: Total number of nodes, levels and computational time of the regression tree with pruning.

Results of the regression tree on the training dataset

In Table 4.16 the results of the pruned regression tree on the training dataset is shown. The model received a mean absolute error of 3,42, a root mean square error of 10,57 and a coefficient of determination of 0,54.

MAE	**RMSE**	**R-squared**
3,426	10,574	0,5489

Table 4.16: Performance metrics of the regression tree with pruning on the training dataset.

Results of the regression tree on the test dataset

In Table 4.17 the results of the regression tree on the test dataset is shown. The model received a mean absolute error of 3,57, a root mean square error of 10,47 and a coefficient of determination of 0,56. It can be seen that the pruned tree does not overfit the data because the errors of the test dataset remain similar to the errors of the model on the training dataset. This makes the pruned regression model a better model for classification since the tree is less complicated and it has received better performance metrics than the unpruned tree on the test dataset. The pruned tree has shown an improved MSE compared to the unpruned tree when used in the test dataset (3,57 compared to 3,82), a reduction in RMSE (10,47 compared to 11,16) and an increase on R-squared (0,56 compared to 0,53).

MAE	**RMSE**	**R-squared**
3,575	10,475	0,5679

Table 4.17: Performance metrics of the regression tree with pruning on the test dataset.

In the figure 4.1 the pruned regression tree is shown with its four first decision levels. It can be seen that the first decision made to calculate the length of stay is to divide the patients in two groups, those whose ICD category is higher or equal than 17,5 and those whose ICD category is less than 17,5. If the patients was assigned a ICD category of less than 17,5 then it goes to the next level (to the left side) in which its further divided depending again on the ICD category. On the other hand, if the patient was assigned a ICD category of more than 17,5, then it goes to the next level (on the right side) in which it will further divide depending on the number of procedures performed on the patient during his stay in the hospital. The regression tree will continue to make binary decisions until all the leafs are reached.

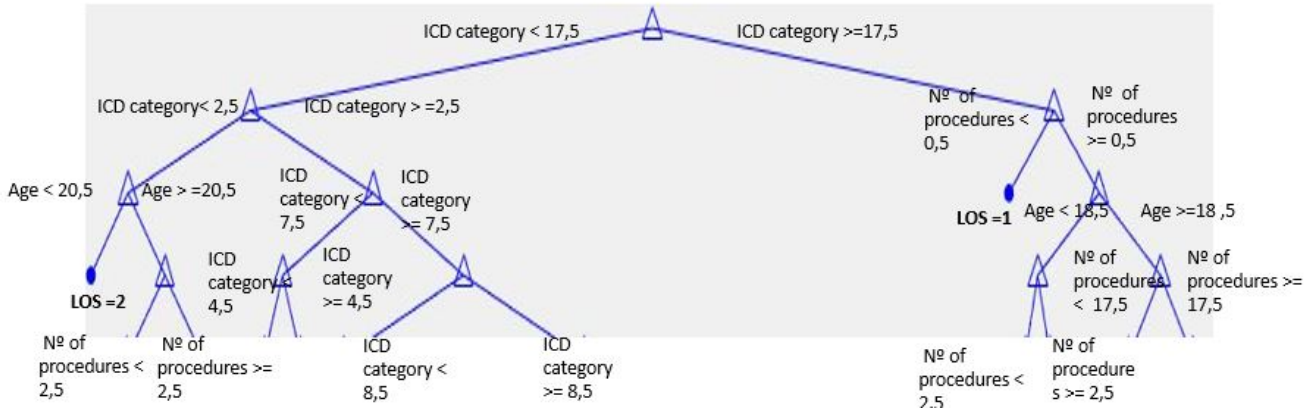

Figure 4.1: This figure illustrates the first four levels of the pruned regression tree.

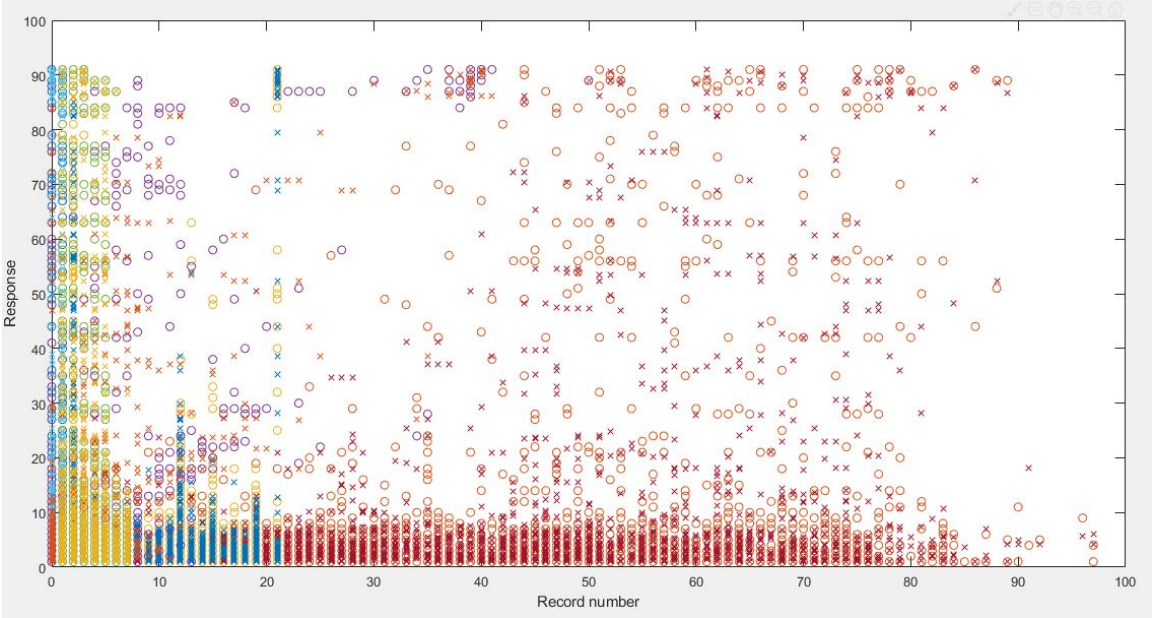

Figure 4.2: Response plot in which the real length of stay (the circles) and the predicted length of stay (the crosses) of the regression model on the test data is projected.

In figure 4.2, a response plot is shown. The 'y' axis represents the length of stay and the 'x' axis represents the number of datapoints. The 'x' axis is normalized from 0 to 100. It can be seen that the regression tree is capable of correctly predicting the length of stay, when the length of stay is below 10 days. Furthermore, as the length of stay of a patient increases, the error of prediction also increases, that is, there is a larger distance between the real length of stay and the predicted length of stay, as it can be seen in the graph. This can be explained due to the large variance of the length of stay (from 0 to 91 days) and the great number of outliers, which makes the algorithm have bigger errors as the length of stay increases.

4.6 k-nearest neighbour(KNN) classification results

In this section, a k-nearest neighbour (KNN) model will be trained in order to classify and predict whether a patient will be a long-term patient (30 days or longer in the hospital) or a short-term patient, staying less than 30 days in the hospital. The KNN model is performed with the variables age, gender, ICD code, number of procedures and day of the week that the patient was accepted. For this model it will be considered that the "yes" prediction will be of the patients that are long-term patients, while the "no" prediction

will be of patients staying in the hospital for less than 30 days. A confusion matrix, its associated ratios, the ROC and AUC will be calculated for both the training dataset and for the test dataset in order to know if the model overtrains the data or not. The model was partitioned using a 80% for training data in order to learn the model with the remaining 20% used to test the model to make sure the model does not overfit the data. The performance of the k-nearest neighbour classifier depends on the value of k [Islam et al., 2007]. Several values of k will be tested on the classifier to see which one is the most optimal and produces the least error. The values of k that will be tested on both the training and test data set are k=1,3,5,7,9. Due to the fact that all values of k are odd numbers, there is no chance of a tie. The method used to calculate distances between the instances will be the Eucledian distance.

4.6.1 Results of the KNN classification with the training dataset

K=1		**Predicted**		**Evaluation measure**	**Score**
	Total=1544	Not long-term	Long-term	**ROC AUC**	0,97
True	Not long-term	14232(TN)	47(FP)	**Accuracy**	0,99408402
	Long-term	42(FN)	723(TP)	**Precision**	0,93896104
				Recall	0,94509804
				F-1 Score	0,93497326

Table 4.18: Confusion matrix and performance metrics of the KNN classification model with K=1 on the training dataset.

K=3		**Predicted**		**Evaluation measure**	**Score**
	Total=1544	Not long-term	Long-term	**ROC AUC**	0,99
True	Not long-term	14197 (TN)	82(FP)	**Accuracy**	0,98165382
	Long-term	194(FN)	571(TP)	**Precision**	0,87442573
				Recall	0,74640523
				F-1 Score	0,79975155

Table 4.19: Confusion matrix and performance metrics of the KNN classification model with K=3 on the training dataset.

K=5		Predicted		Evaluation measure	Score
	Total=1544	Not long-term	Long-term	**ROC AUC**	0,99
True	Not long-term	14190(TN)	89(FP)	**Accuracy**	0,97706727
	Long-term	256(FN)	509(TP)	**Precision**	0,85117057
				Recall	0,66535948
				F-1 Score	0,74304636

Table 4.20: Confusion matrix and performance metrics of the KNN classification model with K=5 on the training dataset.

K=7		Predicted		Evaluation measure	Score
	Total=1544	Not long-term	Long-term	**ROC AUC**	0,98
True	Not long-term	14186(TN)	93(FP)	**Accuracy**	0,97467429
	Long-term	288(FN)	477(TP)	**Precision**	0,83684211
				Recall	0,62352941
				F-1 Score	0,7097931

Table 4.21: Confusion matrix and performance metrics of the KNN classification model with K=7 on the training dataset.

K=9		Predicted		Evaluation measure	Score
	Total=1544	Not long-term	Long-term	**ROC AUC**	0,98
True	Not long-term	14199(TN)	80(FP)	**Accuracy**	0,97268014
	Long-term	331(FN)	434(TP)	**Precision**	0,84435798
				Recall	0,56732026
				F-1 Score	0,672

Table 4.22: Confusion matrix and performance metrics of the KNN classification model with K=9 on the training dataset.

On the training dataset, it can be seen that the optimal number of k is of 3. Increasing the number of k increases accuracy until it reached a saturation point after k=3 where accuracy does not increase when the number of k goes higher. The AUC of the ROC achieved for k=3 is of 0,99, with an accuracy of 0,98. The precision got a score of 0,87 which can be interpreted as 13% of patients being classified as short-term patients, when in reality they were going to be long-term patients. Furthermore, a recall of 0,74 can be interpreted as the model predicting 36% of patients as short-term stay when they should have been predicted as long-term patients. The F-1 score is of 0,79.

4.6.2 Results of the KNN classification with the test dataset

K=1		Predicted		Evaluation measure	Score
	Total=3760	Not long-term	Long-term	**ROC AUC**	0,81
True	Not long-term	3479(TN)	66(FP)	**Accuracy**	0,96223404
	Long-term	76(FN)	139(TP)	**Precision**	0,67804878
				Recall	0,64651163
				F-1 Score	0,65465649

Table 4.23: Confusion matrix and performance metrics of the KNN classification model with K=1 on the test dataset.

K=3		Predicted		Evaluation measure	Score
	Total=3760	Not long-term	Long-term	**ROC AUC**	0,9
True	Not long-term	3499(TN)	46(FP)	**Accuracy**	0,96542553
	Long-term	84(FN)	131(TP)	**Precision**	0,74011299
				Recall	0,60930233
				F-1 Score	0,66268657

Table 4.24: Confusion matrix and performance metrics of the KNN classification model with K=3 on the test dataset.

K=5		Predicted		Evaluation measure	Score
	Total=3760	Not long-term	Long-term	**ROC AUC**	0,92
True	Not long-term	3509(TN)	36(FP)	**Accuracy**	0,96648936
	Long-term	90(FN)	125(TP)	**Precision**	0,77639752
				Recall	0,58139535
				F-1 Score	0,66162963

Table 4.25: Confusion matrix and performance metrics of the KNN classification model with K=5 on the test dataset.

K=7		Predicted		Evaluation measure	Score
	Total=3760	Not long-term	Long-term	**ROC AUC**	0,95
True	Not long-term	3507(TN)	38(FP)	**Accuracy**	0,96489362
	Long-term	94(FN)	121(TP)	**Precision**	0,76100629
				Recall	0,5627907
				F-1 Score	0,64484848

Table 4.26: Confusion matrix and performance metrics of the KNN classification model with K=7 on the test dataset.

K=9		Predicted		Evaluation measure	Score
	Total=3760	Not long-term	Long-term	**ROC AUC**	0,95
True	Not long-term	3512(TN)	33(FP)	**Accuracy**	0,96329787
	Long-term	105(FN)	110(TP)	**Precision**	0,76923077
				Recall	0,51162791
				F-1 Score	0,6103937

Table 4.27: Confusion matrix and performance metrics of the KNN classification model with K=9 on the test dataset.

When performing the KNN model on the test dataset, it can be seen that the optimal number of k is of 7. Increasing the number of k increases the AUC of the ROC and the other performance metrics until it reached a saturation point after k=7 where the performance of the KNN model does not increase when the number of k goes higher. The AUC of the ROC achieved for k=7 is 0,95, with an accuracy of 0,96. The precision got a score of 0,76 which can be interpreted as 24% of patients being classified as short-term patients, when in reality they were going to be long-term patients. Furthermore, a recall of 0,56 can be interpreted as the model predicting 44% of patients as short-term stay when they should have been predicted as long-term patients. The F-1 score is of 0,64. The performance metrics that matter are the ones measured from the use of the model on the test dataset since it is new data that the KNN did not learn. As seen on the previous section, a k=3 used on the training dataset gave the highest performance, but when KNN was tested with k=3 on the test model, the AUC of the ROC and the other performance metrics where much lower when compared to the performance training dataset, which indicates some overtraining. KNN with k=7 is found to be the most robust model since the performance metrics of the training dataset are relatively similar to the ones of the test dataset.

4.7 Comparison of results between the classification tree and k-nearest neighbour(KNN)

In the present section the classification tree and the k-nearest neighbour(KNN) will be compared to see which model predicts and classifies better between a long-term patient or a short-term patient. It was found that the most optimal classification tree was a pruned tree from 24 levels to only 16 levels. On the other hand, the most optimal k-nearest neighbour model was the one in which k=7.

KNN					
	Predicted			**Evaluation measure**	**Score**
K=9	Total=3760	Not long-term	Long-term	**ROC AUC**	0,95
True	Not long-term	3507(TN)	38(FP)	**Accuracy**	0,96
	Long-term	94(FN)	121(TP)	**Precision**	0,76
				Recall	0,56
				F-1 Score	0,64

Classification tree					
	Predicted			**Evaluation measure**	**Score**
	Total=3760	Not long-term	Long-term	**ROC AUC**	0,96
True	Not long-term	3527(TN)	42(FP)	**Accuracy**	0,97
	Long-term	67(FN)	124(TP)	**Precision**	0,74
				Recall	0,64
				F-1 Score	0,69

Table 4.28: Confusion matrix and performance metrics of the KNN and the classification tree models on the test dataset.

It can be seen that there is a similarity in classification performance with KNN and classification trees. The classification tree has an AUC of the ROC of 0,96 while the KNN has an AUC of 0,95. The accuracy obtained in both models also resemble each other(0,97 in the tree compared to 0,96 of the KNN). The precision of the KNN is slightly higher than the classification tree (0,76 compared to 0,74). The classification tree has a much higher Recall than the KNN model (0,64 compared to 0,56). This can be interpreted as the KNN model predicting 44% of patients as staying short-term in the hospital when in reality they were long-term patients. On the other hand, the classification tree predicted 36% of patients as staying short-term in the hospital when in reality they were long-term

patients. The classification tree obtained a higher F-1 score than the KNN model (0,69 compared to 0,64). Finally, the KNN model had a computational time of 1 second while the classification tree had a computational time of 8 seconds. As a conclusion, the tree may be regarded as a better classifier between long and short length of stay with these specific variables due to the better performance metrics received.

Chapter 5

Conclusions and future work

This chapter will provide a summary of the thesis and discuss some possibilities on future work.

5.1 Introduction

The present thesis evaluated the usage of machine learning algorithms in the interest of predicting the length of stay of an individual staying in a hospital based on real data from electronic health records. The prediction of the length of stay has been a big issue for hospitals because it is an important factor that indicates the efficiency of a hospital management and the use of the hospital's resources. Furthermore, an increase of length of stay has been linked to greater acquired infections in the hospital. Having knowledge of the factors that cause an increased length of stay will help hospitals plan better and use their resources more efficiently to decrease the length of stay of patients. It was found that the variables age, gender, ICD code, number of procedures and day of the week that the patient was accepted could predict the length of stay. Certain common machine learning algorithms used for prediction were analyzed and compared theoretically. Machine learning models such as k-Nearest Neighbour, Support Vector Machines, Regression Trees and Neural Networks were found to be widely used for prediction in the field of medicine. In this master's thesis, the following research questions were addressed:

- Using routine data extracted from hospital electronic health records, can effective predictors of an individual patient be used in order to predict whether a patient will be a long-term patient (staying in the hospital for 30 days or longer) or a short-term patient (staying less than 30 days) with the method of classification trees or with the algorithm of k-nearest neighbour?

- Is it possible to use a regression tree in order to calculate the exact length of stay in days using routine data extracted from hospital electronic health records?
- Which techniques are the most effective for choosing an appropriate training dataset for the prediction of length of stay and how is it possible to solve overfitting problems related to machine learning algorithms?

5.2 Conclusions

For the binary outcome of a long-term patient or a short-term patient, models were generated with the machine learning techniques of classification trees and k-nearest neighbour. Afterwards, performance metrics were used to see which model is superior for predictions. Both models had a good classification performance. However, the model that has a better capacity for prediction was found to be the classification tree. It had the best accuracy of 97%, and a higher AUC of the ROC (0,96). Although it had a slightly lower precision than the KNN (74% compared to 76%), the Recall was much higher (64% compared to 56%). This can be interpreted as the KNN model predicting 44% of patients as staying short-term in the hospital when in reality they were long-term patients. On the other hand, the classification tree predicted 36% of patients as staying short-term in the hospital when in reality they were long-term patients. The classification tree was regarded as a better classifier between long and short length of stay due to the better performance metrics received. Similarly, a study by Hyunyoung Baek et. al [Baek et al., 2018], found similar results when classifying between long-term and short-term patients with the model Random forest, with an accuracy of the classification model of 0.9732.
Regarding the regression tree, the model was able to predict the exact length of stay with good accuracy. The length of stay of the patients was highly variable, so a mean absolute error of 3,42, a root mean square error of 10,57 and a R-squared of 0,54 were considered to be good performance metrics.

5.3 Discussion

The findings in this study contribute to the scientific literature because a new combination of variables has been investigated to predict the length of stay in the hospital. Furthermore, most research papers that used machine learning to estimate the length of stay, focused on a specific department, such as for elderly people in institution long-term, in psychiatric wards or in the intensive care unit. Other studies focused on calculating

the length of stay with patients with a specific condition or disease. This thesis took into account all departments of the hospital and procedures performed.
The hospital costs were calculated based on the variables age, gender, ICD code, number of procedures, day of the week that the patient was accepted and total length of stay with a regression tree, but the results were poor and therefore were not presented in this thesis. The hospital cost is a complex and highly variable value that could not be predicted with great accuracy with these specific variables.
A few limitations of this study are as follows: the dataset used for the calculations originated from only one hospital in Germany and thus these results are not representative for all hospitals in Germany. Therefore, the results of this thesis serve only to evaluate the methods capable of predicting the LOS. Another limitation found was the over simplification of the ICD-10 code into only 22 categories and omitting subcategories.
Further research could investigate if data found in electronic health records can effectively predict hospital cost. Future studies could also examine whether the performance metrics could be improved by increasing or decreasing the training dataset, varying the variables implemented, by changing the parameters of the used models in this thesis, or experimenting with other machine learning models to see if they offer a better prediction. In addition, the variable of ICD-10 category, which was assigned to each patient, was found to be very correlated with the duration of the stay in the hospital. Certain diseases and injuries tend to be more serious than others. Further research could investigate the subcategories of each chapter to find out which diseases, causes of injury and signs and symptoms are the cause for a greater length of hospital stay.

Bibliography

Amarasingham, R., Patzer, R. E., Huesch, M., Nguyen, N. Q., and Xie, B. (2014). Implementing electronic health care predictive analytics: considerations and challenges. *Health Affairs*, 33(7):1148–1154.

Baek, H., Cho, M., Kim, S., Hwang, H., Song, M., and Yoo, S. (2018). Analysis of length of hospital stay using electronic health records: A statistical and data mining approach. *PloS one*, 13(4):e0195901.

Baxt, W. G. (1992). Analysis of the clinical variables driving decision in an artificial neural network trained to identify the presence of myocardial infarction. *Annals of emergency medicine*, 21(12):1439–1444.

Bellazzi, R. and Zupan, B. (2008). Predictive data mining in clinical medicine: current issues and guidelines. *International journal of medical informatics*, 77(2):81–97.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. springer.

Bishop, C. M. et al. (1995). *Neural networks for pattern recognition*. Oxford university press.

Bueno, H., Ross, J. S., Wang, Y., Chen, J., Vidán, M. T., Normand, S.-L. T., Curtis, J. P., Drye, E. E., Lichtman, J. H., Keenan, P. S., et al. (2010). Trends in length of stay and short-term outcomes among medicare patients hospitalized for heart failure, 1993-2006. *Jama*, 303(21):2141–2147.

Carter, E. M. and Potts, H. W. (2014). Predicting length of stay from an electronic patient record system: a primary total knee replacement example. *BMC medical informatics and decision making*, 14(1):26.

Chai, T. and Draxler, R. R. (2014). Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature. *Geoscientific model development*, 7(3):1247–1250.

Chen, H., Fuller, S. S., Friedman, C., and Hersh, W. (2006). *Medical informatics: knowledge management and data mining in biomedicine*, volume 8. Springer Science & Business Media.

Cooper, G. F., Aliferis, C. F., Ambrosino, R., Aronis, J., Buchanan, B. G., Caruana, R., Fine, M. J., Glymour, C., Gordon, G., Hanusa, B. H., et al. (1997). An evaluation of machine-learning methods for predicting pneumonia mortality. *Artificial intelligence in medicine*, 9(2):107–138.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

Foraker, R. E., Rose, K. M., Chang, P. P., Suchindran, C. M., McNeill, A. M., and Rosamond, W. D. (2014). Hospital length of stay for incident heart failure: A therosclerosis r isk in c ommunities (aric) cohort: 1987–2005. *Journal for Healthcare Quality*, 36(1):45–51.

Geissler, A., Scheller-Kreinsen, D., Quentin, W., and Busse, R. (2011). Germany: Understanding g-drgs. *Diagnosis-Related Groups in Europe*, pages 243–271.

Goutte, C. and Gaussier, E. (2005). A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. pages 345–359.

Hadzikadic, M. (1992). Automated design of diagnostic systems. *Artificial Intelligence in Medicine*, 4(5):329–342.

Islam, M. J., Wu, Q. J., Ahmadi, M., and Sid-Ahmed, M. A. (2007). Investigating the performance of naive-bayes classifiers and k-nearest neighbor classifiers. In *2007 International Conference on Convergence Information Technology (ICCIT 2007)*, pages 1541–1546. IEEE.

Janke, A. T., Overbeek, D. L., Kocher, K. E., and Levy, P. D. (2016). Exploring the potential of predictive analytics and big data in emergency care. *Annals of emergency medicine*, 67(2):227–236.

John Lu, Z. (2010). The elements of statistical learning: data mining, inference, and prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 173(3).

Kapadohos, T., Angelopoulos, E., Vasileiadis, I., Nanas, S., Kotanidou, A., Karabinis, A., Marathias, K., and Routsi, C. (2017). Determinants of prolonged intensive care unit stay in patients after cardiac surgery: a prospective observational study. *Journal of thoracic disease*, 9(1):70.

Kitts, B. (1997). Regression trees. Technical report, Technical Report, http://www. appliedaisystems. com/papers/RegressionTrees. d oc.

Kline, J. A., Courtney, D. M., Kabrhel, C., Moore, C., Smithline, H., Plewa, M., Richman, P., O'neil, B., and Nordenholz, K. (2008). Prospective multicenter evaluation of the pulmonary embolism rule-out criteria. *Journal of Thrombosis and Haemostasis*, 6(5):772–780.

Larranaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., Lozano, J. A., Armananzas, R., Santafé, G., Pérez, A., et al. (2006). Machine learning in bioinformatics. *Briefings in bioinformatics*, 7(1):86–112.

Lin, H., Long, E., Ding, X., Diao, H., Chen, Z., Liu, R., Huang, J., Cai, J., Xu, S., Zhang, X., et al. (2018). Prediction of myopia development among chinese school-aged children using refraction data from electronic medical records: A retrospective, multicentre machine learning study. *PLoS medicine*, 15(11):e1002674.

Lobo, J. M., Jiménez-Valverde, A., and Real, R. (2008). Auc: a misleading measure of the performance of predictive distribution models. *Global ecology and Biogeography*, 17(2):145–151.

Maglogiannis, I. G. (2007). *Emerging artificial intelligence applications in computer engineering: real word ai systems with applications in ehealth, hci, information retrieval and pervasive technologies*, volume 160. Ios Press.

Marewski, J. N. and Gigerenzer, G. (2012). Heuristic decision making in medicine. *Dialogues in clinical neuroscience*, 14(1):77.

Miller, M. C. (1977). *Medical Diagnostic Models: A Bibliography by M. Clinton Miller, III...[et Al.].*, volume 1. University Microfilms.

Nagelkerke, N. J. et al. (1991). A note on a general definition of the coefficient of determination. *Biometrika*, 78(3):691–692.

Neuman, M. D., Rosenbaum, P. R., Ludwig, J. M., Zubizarreta, J. R., and Silber, J. H. (2014). Anesthesia technique, mortality, and length of stay after hip fracture surgery. *Jama*, 311(24):2508–2517.

Reggia, J. A. (1993). Neural computation in medicine. *Artificial intelligence in medicine*, 5(2):143–157.

Rokach, L. and Maimon, O. (2005). Decision trees. In *Data mining and knowledge discovery handbook*, pages 165–192. Springer.

Sharma, A., Dunn, W., O'Toole, C., and Kennedy, H. G. (2015). The virtual institution: cross-sectional length of stay in general adult and forensic psychiatry beds. *International journal of mental health systems*, 9(1):25.

Smith, I., Elton, R., Ballantyne, J., and Brenkel, I. (2008). Pre-operative predictors of the length of hospital stay in total knee replacement. *The Journal of bone and joint surgery. British volume*, 90(11):1435–1440.

Steinberg, D. and Colla, P. (2009). Cart: classification and regression trees. *The top ten algorithms in data mining*, 9:179.

Visa, S., Ramsay, B., Ralescu, A. L., and Van Der Knaap, E. (2011). Confusion matrix-based feature selection. *MAICS*, 710:120–127.

WHO (2010). International Classification of Diseases 10th Revision. Matlab, URL: `https://es.mathworks.com/products/matlab.html`. Accessed June 30, 2019.

Xie, H., Chaussalet, T. J., and Millard, P. H. (2005). A continuous time markov model for the length of stay of elderly people in institutional long-term care. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 168(1):51–61.

Internet sources

D., L. (2013). Big data gets bigger: now Google trends can predict the market. Forbes, URL: http://www.forbes.com/sites/davidleinweber/2013/04/26/big-data-gets-bigger-now-google-trendscan-predict-the-market/. Accessed February 2019.

Math-Works (2018). Improving classification trees and regression trees. https://es.mathworks.com/help/stats/improving-classification-trees-and-regression-trees.html.

MathWorks (2019). General description. Matlab, URL: https://es.mathworks.com/products/matlab.html. Accessed May 2019.

N., S. (2008). Frequently asked questions. CBC Radio, URL: http://fivethirtyeight.com/features/frequently-asked-questions-lastrevised/. Accessed February 2019.

World-Health-Organization (2018). International classification of diseases (icd) information sheet. https://www.who.int/classifications/icd/factsheet/en/.

Appendix A

Text description of the pruned classification tree for the binary classification of long-term or short-term stay patient

1 if Number of procedures<19.5 then node 2 elseif Number of procedures>=19.5 then node 3 else 1

2 if ICD category<3.5 then node 4 elseif ICD category>=3.5 then node 5 else 1

3 class = 2

4 if Number of procedures<5.5 then node 6 elseif Number of procedures>=5.5 then node 7 else 1

5 if Number of procedures<14.5 then node 8 elseif Number of procedures>=14.5 then node 9 else 1

6 if Age<23.5 then node 10 elseif Age>=23.5 then node 11 else 1

7 if Number of procedures<9.5 then node 12 elseif Number of procedures>=9.5 then node 13 else 2

8 class = 1

9 if ICD category<16.5 then node 14 elseif ICD category>=16.5 then node 15 else 1

10 class = 1

11 if Number of procedures<0.5 then node 16 elseif Number of procedures>=0.5 then node 17 else 1

12 if Gender<0.5 then node 18 elseif Gender>=0.5 then node 19 else 1

13 if Entrance day<6 then node 20 elseif Entrance day>=6 then node 21 else 2

14 class = 1

15 class = 2

16 if ICD category<1.5 then node 22 elseif ICD category>=1.5 then node 23 else 1

17 if Number of procedures<2.5 then node 24 elseif Number of procedures>=2.5 then node 25 else 1

18 if Age<50.5 then node 26 elseif Age>=50.5 then node 27 else 1

19 if Entrance day<4.5 then node 28 elseif Entrance day>=4.5 then node 29 else 2

20 if Age<74.5 then node 30 elseif Age>=74.5 then node 31 else 2

21 class = 1

22 class = 1

23 if Age<37 then node 32 elseif Age>=37 then node 33 else 2

24 class = 1

25 if Gender<0.5 then node 34 elseif Gender>=0.5 then node 35 else 1

26 class = 1

27 if Age<59.5 then node 36 elseif Age>=59.5 then node 37 else 1

28 if Age<65.5 then node 38 elseif Age>=65.5 then node 39 else 2

29 class = 2

30 class = 2

31 class = 1

32 class = 1

33 class = 2

34 class = 1

35 if Number of procedures<4.5 then node 40 elseif Number of procedures>=4.5 then node 41 else 1

36 class = 2

37 class = 1

38 if Number of procedures<6.5 then node 42 elseif Number of procedures>=6.5 then node 43 else 1

39 class = 2

40 if Entrance day<4.5 then node 44 elseif Entrance day>=4.5 then node 45 else 1

41 class = 1

42 if Age<36.5 then node 46 elseif Age>=36.5 then node 47 else 1

43 if Age<50.5 then node 48 elseif Age>=50.5 then node 49 else 2

44 class = 1

45 if Entrance day<6 then node 50 elseif Entrance day>=6 then node 51 else 2

46 class = 2

47 class = 1

48 class = 2

49 class = 1

50 class = 2

51 class = 1

Appendix B

Text description of the pruned regression tree for the calculation of the LOS

1 if ICD category<17.5 then node 2 elseif ICD category>=17.5 then node 3 else 1

2 if ICD category<2.5 then node 4 elseif ICD category>=2.5 then node 5 else 1

3 if Number of procedures<0.5 then node 6 elseif Number of procedures>=0.5 then node 7 else 1

4 if Age<20.5 then node 8 elseif Age>=20.5 then node 9 else 2

5 if ICD category<9.5 then node 10 elseif ICD category>=9.5 then node 11 else 1

6 class = 1

7 if Age<18.5 then node 12 elseif Age>=18.5 then node 13 else 1

8 if Number of procedures<0.5 then node 14 elseif Number of procedures>=0.5 then node 15 else 2

9 if Number of procedures<2.5 then node 16 elseif Number of procedures>=2.5 then node 17 else 4

10 if Number of procedures<1.5 then node 18 elseif Number of procedures>=1.5 then

node 19 else 1

11 if Number of procedures<7.5 then node 20 elseif Number of procedures>=7.5 then node 21 else 2

12 if Number of procedures<2.5 then node 22 elseif Number of procedures>=2.5 then node 23 else 1

13 if Number of procedures<17.5 then node 24 elseif Number of procedures>=17.5 then node 25 else 1

14 if ICD category<1.5 then node 26 elseif ICD category>=1.5 then node 27 else 2

15 if Number of procedures<3.5 then node 28 elseif Number of procedures>=3.5 then node 29 else 1

16 if Number of procedures<0.5 then node 30 elseif Number of procedures>=0.5 then node 31 else 1

17 if Gender<0.5 then node 32 elseif Gender>=0.5 then node 33 else 2

18 if ICD category<4.5 then node 34 elseif ICD category>=4.5 then node 35 else 1

19 if ICD category<8.5 then node 36 elseif ICD category>=8.5 then node 37 else 1

20 if ICD category<10.5 then node 38 elseif ICD category>=10.5 then node 39 else 2

21 if Number of procedures<11.5 then node 40 elseif Number of procedures>=11.5 then node 41 else 18

22 class = 1

23 if ICD category<18.5 then node 42 elseif ICD category>=18.5 then node 43 else 2

24 if Number of procedures<1.5 then node 44 elseif Number of procedures>=1.5 then node 45 else 1

25 if Entrance day<2.5 then node 46 elseif Entrance day>=2.5 then node 47 else 89

26 if Age<14.5 then node 48 elseif Age>=14.5 then node 49 else 2

27 class = 2

28 if ICD category<1.5 then node 50 elseif ICD category>=1.5 then node 51 else 1

29 class = 2

30 if ICD category<1.5 then node 52 elseif ICD category>=1.5 then node 53 else 1

31 if Gender<0.5 then node 54 elseif Gender>=0.5 then node 55 else 1

32 if Age<57.5 then node 56 elseif Age>=57.5 then node 57 else 8

33 if Number of procedures<3.5 then node 58 elseif Number of procedures>=3.5 then node 59 else 4

34 if Number of procedures<0.5 then node 60 elseif Number of procedures>=0.5 then node 61 else 2

35 if ICD category<6.5 then node 62 elseif ICD category>=6.5 then node 63 else 1

36 if ICD category<7.5 then node 64 elseif ICD category>=7.5 then node 65 else 1

37 if Age<44.5 then node 66 elseif Age>=44.5 then node 67 else 1

38 if Age<14.5 then node 68 elseif Age>=14.5 then node 69 else 2

39 if ICD category<11.5 then node 70 elseif ICD category>=11.5 then node 71 else 2

40 if Age<77.5 then node 72 elseif Age>=77.5 then node 73 else 14

41 if Entrance day<1.5 then node 74 elseif Entrance day>=1.5 then node 75 else 18

42 class = 2

43 class = 1

44 if ICD category<18.5 then node 76 elseif ICD category>=18.5 then node 77 else 1

45 if Age<57.5 then node 78 elseif Age>=57.5 then node 79 else 5

46 if Number of procedures<39.5 then node 80 elseif Number of procedures>=39.5 then node 81 else 88

47 if Number of procedures<39.5 then node 82 elseif Number of procedures>=39.5 then node 83 else 89

48 if Age<1.5 then node 84 elseif Age>=1.5 then node 85 else 2

49 class = 3

50 class = 2

51 class = 1

52 if Entrance day<5.5 then node 86 elseif Entrance day>=5.5 then node 87 else 1

53 class = 56

54 if Number of procedures<1.5 then node 88 elseif Number of procedures>=1.5 then node 89 else 4

55 if Number of procedures<1.5 then node 90 elseif Number of procedures>=1.5 then node 91 else 1

56 if Entrance day<4.5 then node 92 elseif Entrance day>=4.5 then node 93 else 2

57 if Number of procedures<5.5 then node 94 elseif Number of procedures>=5.5 then node 95 else 8

58 class = 42

59 if Number of procedures<5.5 then node 96 elseif Number of procedures>=5.5 then node 97 else 4

60 if Entrance day<5.5 then node 98 elseif Entrance day>=5.5 then node 99 else 2

61 if ICD category<3.5 then node 100 elseif ICD category>=3.5 then node 101 else 1

62 class = 1

63 if ICD category<8.5 then node 102 elseif ICD category>=8.5 then node 103 else 1

64 if Age<13.5 then node 104 elseif Age>=13.5 then node 105 else 1

65 if Age<12.5 then node 106 elseif Age>=12.5 then node 107 else 6

66 class = 2

67 if Number of procedures<4.5 then node 108 elseif Number of procedures>=4.5 then node 109 else 1

68 if Number of procedures<2.5 then node 110 elseif Number of procedures>=2.5 then node 111 else 2

69 if Number of procedures<0.5 then node 112 elseif Number of procedures>=0.5 then node 113 else 6

70 if Age<3.5 then node 114 elseif Age>=3.5 then node 115 else 2

71 if ICD category<12.5 then node 116 elseif ICD category>=12.5 then node 117 else 1

72 if ICD category<13.5 then node 118 elseif ICD category>=13.5 then node 119 else 14

73 class = 14

74 if Number of procedures<15.5 then node 120 elseif Number of procedures>=15.5 then node 121 else 18

75 if Number of procedures<17.5 then node 122 elseif Number of procedures>=17.5 then node 123 else 22

76 class = 1

77 if ICD category<20 then node 124 elseif ICD category>=20 then node 125 else 1

78 if Number of procedures<3.5 then node 126 elseif Number of procedures>=3.5 then node 127 else 4

79 class = 1

80 class = 88

81 class = 91

82 if Number of procedures<38.5 then node 128 elseif Number of procedures>=38.5 then node 129 else 89

83 class = 91

84 if Entrance day<5.5 then node 130 elseif Entrance day>=5.5 then node 131 else 1

85 class = 2

86 if Age<28.5 then node 132 elseif Age>=28.5 then node 133 else 5

87 class = 1

88 class = 1

89 class = 4

90 class = 2

91 if Age<74.5 then node 134 elseif Age>=74.5 then node 135 else 21

92 class = 2

93 class = 3

94 if Number of procedures<3.5 then node 136 elseif Number of procedures>=3.5 then node 137 else 8

95 if Number of procedures<9.5 then node 138 elseif Number of procedures>=9.5 then node 139 else 2

96 if Entrance day<6 then node 140 elseif Entrance day>=6 then node 141 else 4

97 if Number of procedures<9.5 then node 142 elseif Number of procedures>=9.5 then node 143 else 84

98 if Age<18 then node 144 elseif Age>=18 then node 145 else 2

99 class = 3

100 class = 1

101 if Age<28.5 then node 146 elseif Age>=28.5 then node 147 else 10

102 if Number of procedures<0.5 then node 148 elseif Number of procedures>=0.5 then node 149 else 1

103 class = 1

104 if Number of procedures<3.5 then node 150 elseif Number of procedures>=3.5 then node 151 else 1

105 if Entrance day<2.5 then node 152 elseif Entrance day>=2.5 then node 153 else 2

106 class = 7

107 class = 6

108 if Entrance day<3.5 then node 154 elseif Entrance day>=3.5 then node 155 else 2

109 if Number of procedures<7.5 then node 156 elseif Number of procedures>=7.5 then node 157 else 1

110 if Number of procedures<0.5 then node 158 elseif Number of procedures>=0.5 then node 159 else 2

111 class = 2

112 if Age<55.5 then node 160 elseif Age>=55.5 then node 161 else 3

113 class = 6

114 class = 2

115 if Number of procedures<0.5 then node 162 elseif Number of procedures>=0.5 then node 163 else 2

116 if Number of procedures<1.5 then node 164 elseif Number of procedures>=1.5 then node 165 else 1

117 if ICD category<14.5 then node 166 elseif ICD category>=14.5 then node 167 else 1

118 if Entrance day<2.5 then node 168 elseif Entrance day>=2.5 then node 169 else 14

119 class = 1

120 class = 18

121 class = 25

122 if Entrance day<2.5 then node 170 elseif Entrance day>=2.5 then node 171 else 22

123 if Number of procedures<20.5 then node 172 elseif Number of procedures>=20.5 then node 173 else 28

124 if Age<52.5 then node 174 elseif Age>=52.5 then node 175 else 3

125 class = 1

126 class = 4

127 class = 5

128 if Entrance day<4.5 then node 176 elseif Entrance day>=4.5 then node 177 else 89

129 class = 89

130 class = 1

131 class = 3

132 class = 3

133 class = 5

134 class = 21

135 class = 1

136 class = 1

137 class = 8

138 class = 2

139 class = 84

140 if Number of procedures<4.5 then node 178 elseif Number of procedures>=4.5 then node 179 else 4

141 class = 4

142 if Entrance day<3.5 then node 180 elseif Entrance day>=3.5 then node 181 else 4

143 class = 84

144 class = 2

145 class = 1

146 if Entrance day<2.5 then node 182 elseif Entrance day>=2.5 then node 183 else 10

147 class = 3

148 if Age<10.5 then node 184 elseif Age>=10.5 then node 185 else 2

149 if Age<17.5 then node 186 elseif Age>=17.5 then node 187 else 6

150 if Entrance day<1.5 then node 188 elseif Entrance day>=1.5 then node 189 else 1

151 if Entrance day<1.5 then node 190 elseif Entrance day>=1.5 then node 191 else 2

152 class = 3

153 if ICD category<5.5 then node 192 elseif ICD category>=5.5 then node 193 else 2

154 if Number of procedures<2.5 then node 194 elseif Number of procedures>=2.5 then node 195 else 2

155 class = 1

156 class = 1

157 if Age<68.5 then node 196 elseif Age>=68.5 then node 197 else 3

158 if Age<0.5 then node 198 elseif Age>=0.5 then node 199 else 2

159 if Age<1.5 then node 200 elseif Age>=1.5 then node 201 else 5

160 class = 3

161 class = 1

162 class = 1

163 if Age<21.5 then node 202 elseif Age>=21.5 then node 203 else 2

164 if Age<44.5 then node 204 elseif Age>=44.5 then node 205 else 1

165 if Number of procedures<5.5 then node 206 elseif Number of procedures>=5.5 then node 207 else 1

166 if Age<0.5 then node 208 elseif Age>=0.5 then node 209 else 1

167 if Age<56.5 then node 210 elseif Age>=56.5 then node 211 else 2

168 if Number of procedures<9.5 then node 212 elseif Number of procedures>=9.5 then node 213 else 10

169 if Entrance day<4.5 then node 214 elseif Entrance day>=4.5 then node 215 else 14

170 class = 17

171 if Entrance day<3.5 then node 216 elseif Entrance day>=3.5 then node 217 else 22

172 class = 29

173 class = 28

174 class = 1

175 if Entrance day<1.5 then node 218 elseif Entrance day>=1.5 then node 219 else 2

176 class = 89

177 class = 87

178 class = 63

179 if Entrance day<2.5 then node 220 elseif Entrance day>=2.5 then node 221 else

4

180 if Age<66.5 then node 222 elseif Age>=66.5 then node 223 else 4

181 class = 70

182 class = 10

183 class = 12

184 class = 3

185 class = 2

186 class = 1

187 class = 6

188 class = 2

189 class = 1

190 class = 3

191 class = 2

192 class = 4

193 class = 2

194 if Age<74.5 then node 224 elseif Age>=74.5 then node 225 else 1

195 class = 2

196 class = 3

197 class = 1

198 class = 2

199 if Entrance day<5.5 then node 226 elseif Entrance day>=5.5 then node 227 else 2

200 class = 3

201 if Age<6.5 then node 228 elseif Age>=6.5 then node 229 else 5

202 if Number of procedures<2.5 then node 230 elseif Number of procedures>=2.5 then node 231 else 2

203 if Number of procedures<2.5 then node 232 elseif Number of procedures>=2.5 then node 233 else 2

204 if Age<15.5 then node 234 elseif Age>=15.5 then node 235 else 1

205 class = 2

206 if Entrance day<5.5 then node 236 elseif Entrance day>=5.5 then node 237 else 1

207 class = 10

208 if Number of procedures<0.5 then node 238 elseif Number of procedures>=0.5 then node 239 else 5

209 if Number of procedures<3.5 then node 240 elseif Number of procedures>=3.5 then node 241 else 1

210 if Number of procedures<0.5 then node 242 elseif Number of procedures>=0.5 then node 243 else 2

211 class = 2

212 class = 10

213 class = 15

214 class = 13

215 class = 14

216 class = 23

217 class = 22

218 class = 2

219 class = 3

220 class = 4

221 if Age<54.5 then node 244 elseif Age>=54.5 then node 245 else 5

222 class = 5

223 class = 4

224 class = 1

225 class = 2

226 class = 1

227 class = 2

228 if Entrance day<3.5 then node 246 elseif Entrance day>=3.5 then node 247 else 5

229 class = 7

230 if Age<18.5 then node 248 elseif Age>=18.5 then node 249 else 3

231 class = 2

232 if Entrance day<5.5 then node 250 elseif Entrance day>=5.5 then node 251 else 2

233 if Number of procedures<4.5 then node 252 elseif Number of procedures>=4.5 then node 253 else 2

234 if Entrance day<5.5 then node 254 elseif Entrance day>=5.5 then node 255 else 3

235 class = 1

236 if Entrance day<2.5 then node 256 elseif Entrance day>=2.5 then node 257 else 1

237 class = 3

238 class = 4

239 class = 2

240 if Number of procedures<2.5 then node 258 elseif Number of procedures>=2.5 then node 259 else 1

241 if Entrance day<3.5 then node 260 elseif Entrance day>=3.5 then node 261 else 2

242 if Age<23.5 then node 262 elseif Age>=23.5 then node 263 else 2

243 if Age<0.5 then node 264 elseif Age>=0.5 then node 265 else 2

244 class = 63

245 class = 5

246 class = 5

247 class = 1

248 if Age<7.5 then node 266 elseif Age>=7.5 then node 267 else 3

249 class = 2

250 if Number of procedures<1.5 then node 268 elseif Number of procedures>=1.5 then node 269 else 2

251 class = 2

252 if Age<41.5 then node 270 elseif Age>=41.5 then node 271 else 3

253 class = 2

254 if Age<11.5 then node 272 elseif Age>=11.5 then node 273 else 3

255 class = 2

256 if Number of procedures<3.5 then node 274 elseif Number of procedures>=3.5 then node 275 else 1

257 if Entrance day<3.5 then node 276 elseif Entrance day>=3.5 then node 277 else 7

258 if Age<55.5 then node 278 elseif Age>=55.5 then node 279 else 1

259 class = 1

260 class = 3

261 class = 4

262 class = 1

263 if Age<24.5 then node 280 elseif Age>=24.5 then node 281 else 2

264 if Entrance day<4.5 then node 282 elseif Entrance day>=4.5 then node 283 else 2

265 if Age<9.5 then node 284 elseif Age>=9.5 then node 285 else 2

266 if Entrance day<1.5 then node 286 elseif Entrance day>=1.5 then node 287 else 1

267 class = 3

268 if Age<45.5 then node 288 elseif Age>=45.5 then node 289 else 2

269 class = 1

270 class = 1

271 class = 3

272 if Entrance day<3.5 then node 290 elseif Entrance day>=3.5 then node 291 else 3

273 class = 5

274 class = 1

275 class = 3

276 if Age<30.5 then node 292 elseif Age>=30.5 then node 293 else 6

277 if Age<33.5 then node 294 elseif Age>=33.5 then node 295 else 1

278 if Gender<0.5 then node 296 elseif Gender>=0.5 then node 297 else 1

279 if Number of procedures<0.5 then node 298 elseif Number of procedures>=0.5 then node 299 else 1

280 class = 4

281 if Entrance day<2.5 then node 300 elseif Entrance day>=2.5 then node 301 else 2

282 class = 2

283 class = 3

284 if Number of procedures<1.5 then node 302 elseif Number of procedures>=1.5 then node 303 else 1

285 if Number of procedures<2.5 then node 304 elseif Number of procedures>=2.5 then node 305 else 2

286 class = 1

287 if Number of procedures<1.5 then node 306 elseif Number of procedures>=1.5 then node 307 else 2

288 class = 2

289 if Entrance day<4.5 then node 308 elseif Entrance day>=4.5 then node 309 else 1

290 class = 3

291 class = 1

292 class = 6

293 class = 7

294 class = 1

295 if Age<48.5 then node 310 elseif Age>=48.5 then node 311 else 7

296 class = 1

297 if Entrance day<1.5 then node 312 elseif Entrance day>=1.5 then node 313 else 2

298 class = 1

299 if Entrance day<6.5 then node 314 elseif Entrance day>=6.5 then node 315 else 1

300 if Age<34.5 then node 316 elseif Age>=34.5 then node 317 else 3

301 if Entrance day<3.5 then node 318 elseif Entrance day>=3.5 then node 319 else 2

302 class = 2

303 if Number of procedures<3.5 then node 320 elseif Number of procedures>=3.5 then node 321 else 1

304 if Age<16.5 then node 322 elseif Age>=16.5 then node 323 else 2

305 class = 2

306 class = 1

307 class = 2

308 class = 2

309 class = 1

310 class = 12

311 class = 7

312 if Age<50.5 then node 324 elseif Age>=50.5 then node 325 else 1

313 class = 2

314 if Gender<0.5 then node 326 elseif Gender>=0.5 then node 327 else 1

315 class = 2

316 if Age<29.5 then node 328 elseif Age>=29.5 then node 329 else 2

317 class = 3

318 class = 2

319 if Age<36.5 then node 330 elseif Age>=36.5 then node 331 else 1

320 class = 1

321 class = 2

322 class = 1

323 class = 2

324 if Number of procedures<0.5 then node 332 elseif Number of procedures>=0.5 then node 333 else 1

325 class = 2

326 if Number of procedures<1.5 then node 334 elseif Number of procedures>=1.5 then node 335 else 1

327 if Age<58.5 then node 336 elseif Age>=58.5 then node 337 else 1

328 class = 1

329 class = 2

330 class = 1

331 class = 2

332 class = 1

333 class = 4

334 class = 1

335 class = 3

336 class = 3

337 class = 1

Obligatory Signed Declaration

concerning this thesis titled

CALCULUS OF LENGTH OF STAY ON THE BASIS OF THE GERMAN DRG DATA USING CART AND K-NEAREST NEIGHBOUR

"I hereby declare that the present thesis was composed by myself and that the work contained herein is my own. I also confirm that I have only used the specified resources. All formulations and concepts taken verbatim or in substance from printed or unprinted material or from the Internet have been cited according to the rules of good scientific practice and indicated by footnotes or other exact references to the original source.

The present thesis has not been submitted to another university for the award of an academic degree in this form. This thesis has been submitted in printed and electronic form. I hereby confirm that the content of the digital version is the same as in the printed version.

I understand that the provision of incorrect information may have legal consequences."

Graz, July 5, 2019

(VICENT JOSEPH ANDREU TRAWICK)