Leiden Institute of Advanced Computer Science
Bioinformatics Group

# Predicting Graft Survival in Kidney Transplantation using Neural Networks and Semantic Modelling.

*Master Thesis*

Petros Dimitriadis

Supervisors:

Katherine Wolstencroft
Jan Lindeman
Alexandre Gultyaev

Leiden, September 2018

# Contents

**Abstract**

Kidney transplantation is considered the foremost treatment for patients suffering from chronic end-stage kidney disease, extending life expectancy and quality of life. Successful transplantation and organ survival are determined by many physical and biochemical factors, which in combination with a shortage of suitable donor organs, makes the process of kidney graft allocation complex.

The aim of this project was to develop new approaches to this problem. We have developed Artificial Neural Networks (ANNs) for predicting graft survival based on donor-recipients characteristics and biochemical measurements. In addition, we proposed a semantic model that can be used as the basis for a Personalized Decision Support System (PDSS) that will allow clinicians to describe the characteristics of new donor-recipient pairings and obtain predictions from the model.

The data used in this study were gathered by the Netherlands Organ Transplantation Registry (NOTR) between 2000 and 2017 and capture the information of 10.410 operations with deceased donors.

We produced two predictive models. The first used the complete dataset of pre- and post-operative variables, the second used only pre-operative variables. The predictive performance from the post-operative model shows an accuracy of 97% in forecasting graft survival, whilst the pre-operative model is 76.8% accurate. The results of our proposed models are comparable with other prognostic and diagnostic models proposed in the literature for other health problems, establishing ANNs as a promising tool to support kidney graft allocation.

**Acknowledgement**

The success of this project required a lot of guidance and assistance from many people and I feel indebted to express my gratitude to all these people who contributed and supported me in the completion of this project.

First of all, I would like to express my great appreciation to Dr Jan Lindeman for trusting to me this interesting project, but also for giving me the opportunity to work with him and share his knowledge in transplantations with me.

I owe my deep gratitude to Dr Katy Wolstencroft for her weekly supervision and patient guidance throughout this project. Her useful critiques, advice and assistance were invaluable in order to accomplish this project. In addition, her pleasant attribute made this collaboration joyful and productive the same time.

I would also like to acknowledge Dr Alexandre Gultyaev as the second reader of my thesis and I am gratefully indebted to him for his valuable comments.

Furthermore, I gratefully acknowledge the contributions of my mentors, Michele Kok and Lise Stork in the clinical and the technical part of the project respectively. Without your support and your inspiration girls, this project would have not been successfully completed. A special thanks to Joost Martens for the brainstorm session at the beginning of our projects

I would like to thank my family, who without their love and support I would have not been able to accomplish this Masters. For this reason I would like to dedicate this work to them.

Last but not least, I would like to express my special thanks to my friends that supported me, encouraged me but also motivate me sometimes throughout these years.

*Petros Dimitriadis*

# Chapter 1

# Introduction

## 1.1 Problem Description

Medical advances developed in the last century have extended life expectancy and improved the quality of life for patients suffering from severe organ diseases. Renal transplantation is the surgical operation where patients suffering from end-stage renal disease have their malfunctioning kidney(s) replaced with a healthy functioning kidney from a donor[1]. A successful organ transplantation may prevent early mortality and extend life expectancy three fold, whilst also improving the quality of life for people suffering from end-stage renal disease [2][3][4].

Chronic renal failure is diagnosed when kidney functionality falls below 15% to properly filter the blood and produce hormones and urine [5][6]. This can be caused by a metabolic disorder, genetic diseases, hypertension, malignancies or diabetes. Conventional treatments in acute kidney disease are mainly based on hemodialysis and peritoneal dialysis [7]. These treatments aim to maintain a patients life and extend life expectancy for 5-10 years depending on other medical conditions, without improving organs functionality [8][7]. Patients who undergo long term dialysis are at higher mortality risk due to the increased risk of cardiovascular disease while the quality of life of these people is affected because of the long hospitalizations and the fact that they are prone on infections [8][7][2].

The alternative is organ transplantation which despite the beneficial impact in life quality and reduced risks compered to conventional treatments, is a multiparameter process involving a number of limitations that have raised the scientific interest in the field. Organ shortage from donors is the main factor restricting renal transplantations due to the elevated number of organs in demand in comparison with those supplied [2][3][9]. In addition to the extended number of patients on the waiting list to get transplanted, there are also some clinical limitations that affect the success of the transplantation. The barriers that concern clinicians on organ allocation could be donor - recipient compatibility, immunologic rejection, delayed graft function (DGF) and patients elegibility to become a potential recipient [10][11]. Delayed Graft Function (DGF) is the condition immediately after the transplantation where the kidney does not start functioning straight away due to the heal-

ing time it needs after the surgery. DGF may last from a few days up to a few weeks and patients during this period may need to undergo dialysis. Grafts from living donors are less prone on DGF, while organs procured from cadaveric donors show a higher probability of experiencing DGF (around 30%). Heart disease, severe respiratory conditions, active malignancies, drug and alcohol addictions, prolonged duration of dialysis are only some of the contraindications that prohibit patients to be potential recipients [12]. If the clinicians decide that a patient does not fulfill the requirements for being an eligible recipient, then the best treatment for those is to continue being on dialysis [13].

Potential donors can be classified into two categories: Living donors and Cadaveric donors. Living donors can be genetically related, which is the most often case, but also they can be non-related [12]. Cadaveric donors are divided into brain dead donors or heart beating donors (DBDs), who are patients that are considered dead but their heart is still pumping and cardiac death donors (DBDs). Deceased transplants are those originated from donors who have usually passed away due to an accident, heart attack or stroke. However, not all the deceased transplants are eligible for transplantation. Transplantation organizations have developed strict guidelines for organ selection and allocation to avoid graft rejection and at the same time to enhance the survival rate of the organ by implanting it to the most suitable recipient [12].

Once patient's assessment is completed and the transplantation team approves them as candidate recipients, they are put on the waiting list. Clinicians are requested to make the optimal decision for graft allocation based on specific characteristics of the donor's and the patient's profiles according to an explicit protocol that defines all the steps that need to be taken and proceed with the transplantation procedure [14]. However, due to the nature of the problem, this process is not that straightforward. There are multiple interacting factors that need to be considered correctly, which cannot be easily handled by a human being. This is where personalized medical decision support systems are applied to enhance the precision on the decision making process. In chapter 4 we will discuss how semantic modelling application can assist physicians to make more accurate decisions for their patients.

The Dutch Transplant Foundation is one of eight cooperating organizations around Europe that form the Eurotransplant organization. The scope of Eurotransplant is to exchange deceased donor grafts across the country members and allocate them to the most suitable recipient with transparency and equal opportunities for all the patients. Eurotransplant plays the role of mediator among the transplantation centers involved in the program. To properly allocate the grafts, a waiting list is formed for all the eligible recipients which records the patient's characteristics in order to be prioritized. The allocation criteria used for graft distribution are blood group, tissue characteristics, clinical urgency and waiting time [14]. Once a donor is available Eurotransplant accesses the prioritized recipient waiting list to allocate the organ to the most suitable recipient based on the allocation criteria [14].

When a matching recipient - donor pair is found, the operation is arranged. The transplantation lasts approximately three hours and requires general anesthesia [5][12]. The new organ is implanted in recipients lower abdomen, while in most of the cases the

malfunctioning kidney remains in its position [5][12]. The anastomosis is completed by connecting the arteries and vein from the new organ to recipient's blood supply and finally the ureter is attached to the recipient's bladder. Short cold ischemia times and short re-warm enhance initial graft function, which takes 3-5 days for a living kidney and 5-15 days for a cadaveric kidney. Figure 1.1 depicts where the implanted graft is located on the recipient's body[1].

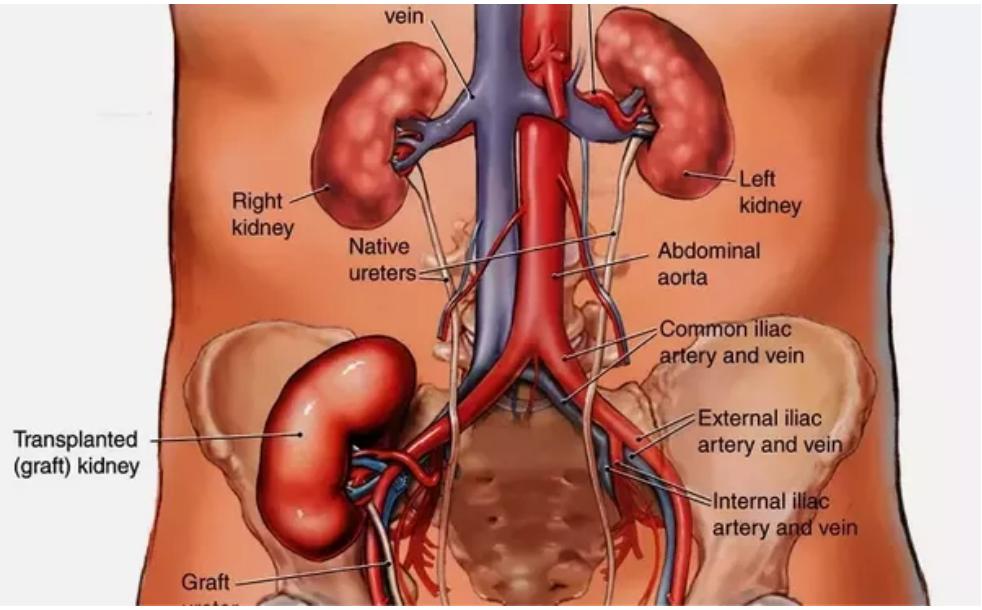



Figure 1.1: A transplanted kidney.

In the early years of transplantation the transplants originated only from living donors [15]. However, due to the graft shortage and the introduction of immunosuppresion therapies, today there are three types of potential donors to enrich the donors pool. Those are living donors, cardiac death donors (DBD) and brain death donors (DCD) [5]. But again many people are still on the waiting lists and die waiting for a graft. This shortage of grafts sparks the discussion for relaxing the restrictions on the criteria for potential donors in order to use organs procured from 'extended' donors who would otherwise be discarded from the selection process [16].

Many studies have been conducted regarding proper organ allocation conditions and the protocol that clinicians need to follow during the operation. In the light of kidney graft shortage, Schaapherder et al studied the results of using transplants from Cardiac death donors (DCD) and Brain Death donors (DBD) [3]. The study conducted in 3.611 DBD and 2.711 DCD cases collected from January $1^{st}$ 1990 until December $31^{st}$ 2017 from the Dutch Transplant Foundation. The results revealed that long term graft survival was equivalent in the two types of donors despite the higher number of incidences of primary non-function and delayed graft function for DCDs. Primary non-function (PNF) describes

[1] Source : https://www.quora.com/What-is-the-success-rate-of-kidney-transplantation

the immediate and irreversible dysfunction of the kidney [17]. This condition is diagnosed with biopsy and is found that causes 0.6% - 8% of the graft losses. Recipients who are diagnosed with PNF need dialysis immediately and are of high mortality risk.

Ittersum et al conducted a statistical analysis for transplantations in The Netherlands for the period from 1995 to 2005 [18]. The main objective of this study was to identify the effect of using lower quality organs from deceased donors on two categories of recipients: those below the age of 40 and those who suffer from diabetes. The authors concluded that grafts from marginal donors have a high impact in the two categories under study in comparison with the general number of transplantations.

The consequences of ischemia reperfusion injury in long term perspective were studied by Tennankore et al [19]. Ischemia reperfusion injury is defined as the harm caused in the tissue during the time it is not supplied with blood. The authors focused on the influence of prolonged warm ischemia time and the long term outcome in these cases. The high association between DGF and delayed organ reperfusion was identified in this study. The authors claim that prolonged warm ischemia can cause irreversible tissue injury that enhances DGF and has a long term effect in graft survival and patients mortality. Another interesting study conducted by Smith et al regarding the outcome for recipients that receive transplants form elder donors [20]. The study examined 329 transplantation cases where both recipients and donor were older than 65 years old. The study revealed that PNF shown to occur more often when the cold ischemia time was extended, while 86% of the recipients died with functioning graft, which might be due to the reduced cold ischemia time.

All in all, successful kidney transplantation is a problem that requires numerous of factors to be taken into account to make the appropriate decision on how the grafts must be allocated and give life to thousands of people suffering of end stage renal disease. Furthermore, the limitations caused by the Standard Criteria Donor (SCD) in combination with the people dying on the waiting lists due to graft shortage add to the problem which needs to be solved soon. In literature, many studies have been conducted regarding kidney transplantations. However, the main debate of these studies is to statistically identify the significance of different parameters in graft survival but not to propose a state of the art approach that will facilitate the decision making for graft allocation. For example Fritsche el at investigated the effect of using graft from old donor (above 60 years old) for old recipients (above 60 years as well)[21]. The conclusion of old to old program revealed that 85.3% of the recipients had one year survival and graft survival was 83.6% in comparison with 86.9% and 89.5% respectively for transplantation performed based on SCD. In another study, Massie et al discuss the impact of the new allocation criteria introduced in the states[22]. According to this study healthier grafts are allocated to younger and healthier patients whilst graft from cadaveric donors where allocated to sensitized patients and racial/ethnic minorities. However, the results after nine months of the time these allocation criteria started to be applied shown that DGF was increased, which may affect the long term survival of the grafts. In this study we propose a model based on Artificial Neural Networks that will take into account all these important factors that previous studies discussed and predict the post-operative outcome for each donor recipient pair. In this way

our model can aid clinical decision making in individual basis and support the decisions based on the clinical outcome of previous transplantations.

### Problem definition

Given a set of characteristics $S_i = \{x_0, x_1, ...x_n\}$ from the donor and the recipient for each operation performed and a set of labels $A = \{y_0, y_1, ...y_n\}$ which represent the clinical outcome of this operation, we develop a classifier based on Artificial Neural Networks. The number of features that compose the trasplantation data set is 112 and the number of the examples cases that involves is 10.410. The proposed model is capable of learning the characteristics of the patients for each distinct label and based on this knowledge it can generalize it in new cases by classifying them into the most similar class of the training data. Then, the outcome of the classifier can be used to construct a semantic model to support medical decision making for patient prioritization and graft allocation.

Making predictions of medical outcomes is an extremely challenging task owing to the diversity of the characteristics that describe each patient and the final reaction of their body. In addition, when we are talking about people's lives, the gap of the medical error should be minimized as much as possible and try to make the most accurate prediction to save people's lives. For this reason, robust techniques, such as Machine learning (ML) and data mining, are required to make the accurate prediction and support the decision made by clinicians for each individual case. However, although some may argue that ML techniques are very abstract to trust for medical decisions, we believe that their ability to assimilate knowledge from previous example cases in comparison with the limited number of factors that clinicians take into account [23], can establish them as a powerful tool to support clinical decision.

## 1.2 Methodology

In this section we discuss the methodology for solving the problem in graft survival prediction which arises from the graft allocation. The workflow graph in figure 1.2 demonstrates the pipeline of the steps taken to solve the problem. The process begins with the hypothetical motivation on how the problem can be solved based on the previous studies in the literature. Data collection and data pre-processing follow. Next, we prepare the data for modeling, by splitting them into training and validation sets. The training process includes feature extraction, classification and training the model. In the last phase of modelling, the performance is assessed, to validate the accuracy of the model. In the final step of this project we propose a semantic model that would be the basis for future development of a decision support system that can exploit the knowledge obtained from the model, to support clinical decisions. In the next paragraphs we are going to discuss each of the aforementioned steps in more detail.

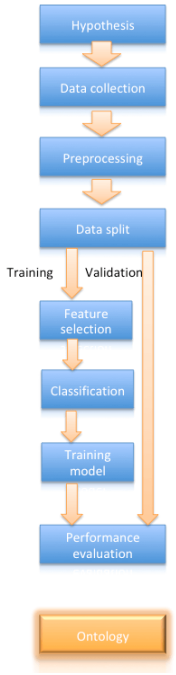



Figure 1.2: Workload pipeline of the project

### 1.2.1 Hypothesis

Many novel technological advances in machine learning have found applications in health care prognosis and diagnosis. These technologies are capable of dealing with enormous amounts of heterogeneous data to make accurate predictions [24][23]. As we discussed in the previous section (1.1), transplantations are a rather complex topic, with many parameters playing a crucial role in graft survival, while the nature of the data that represent the donors and the recipients status are very diverse. Since the early years of transplantations, many publications appeared to discuss the optimal preoperative monitoring of the donor-recipient pair [25]. However, in order to perfectly understand all the clinical trials and observational studies we need to have specialized knowledge in pharmacology, immunology, endocrinology, nephrology and physiology [25]. On the other hand, clinicians can deal with a limited amount of parameters during the decision-making process, which may not exceed the number of five factors per decision [23]. In addition, every patient is a unique case [26]. In other words, every patient shows a different response to a specific treatment, which implies that every patient needs to be treated as an individual [26].

Taking all these parameters into account, we need to develop a prediction model that can take care of all the individual parameters in each operation and forecast the expected outcome and validate it based on prior knowledge. This would help us develop a personalized clinical decision support system in order to make the most accurate decision. The decision support system then is going to detect the most matching recipient - donor pair regarding long term transplant survival. In this way, graft allocation would happen more

precisely and the grafts would be distributed to people whose profile expresses the highest probability to match and function in the maximum.

### 1.2.2 Data collection

The data used in this project is a collective effort from the 8 kidney transplantation center existing in the Netherlands which are cooperators in the Dutch Transplant Foundation. The data acquired in the period of $1^{st}$ of January 1990 until the $31^{st}$ of December 2017 and the records of each transplantation happened with deceased donors in this period were kept at the Netherlands Organ Transplant Registry (NOTR) [3]. The information included in the data contain donor and recipient characteristics but also a variety of preoperative and post-operative clinical measurements that characterize the medical condition of the patients. The amount of transplantation examples, which is 10.402 records, is sufficient to include the required variance to perform accurate predictions even for exeptional examples [27]. This is because every patient has a different reaction in a given treatment which leads to unique patterns in the data-set.

### 1.2.3 Preprocessing

Data preprocessing consists one of the most important operations that need to be considered before analyzing the data. Despite the wealth of information provided in Electronic health records (EHRs), the nature of data acquisition followed imposes significant limitations. This is because data are gathered by various people and each of them may record the same information in a different way in the absence of strict guidelines or any curation process. Another common phenomenon that affects the quality of EHRs in that information is omitted to be recorded. In additions, when EHRs from various medical centers are merged, are found to be incompatible with each other due to the use of different protocols in each center. Consequently, the resulting data-sets are incomplete and contain spacious entries limiting the efficiency of the significant information captured on them. For this reason, as we will see later in section 2.2.1 various techniques are devised to deal with the missing data, data normalization and omitting useless data in a such a way that the computational procedure is sped up while the attributes of the data remain easily recognizable to improve the efficiency of the results.

### 1.2.4 Data Scaling

In order to improve the performance of a ML system and reduce the computational cost the data need to be normally distributed. For this reason we need to standardize the range of the values of each individual variable. As we will discuss later in section 3.4, we applied standart scaler which changes the distribution of the values of each variable in a such a way that the standard deviation is equal to one and the mean of the values is equal to zero.

### 1.2.5 Data spliting

Data splitting is a necessary step in all supervised learning methods used. In order to build an accurate model we first need to train the model in a given set of data and then evaluate it in data that the algorithm has not seen before. In this way, we can evaluate the knowledge obtained during the learning process on new example cases. This is the reason why we divide the original data set into a training and validation set. Next, for both training and testing sets, we need to define the attribute that the model has to predict according to the other features in the data. This attribute is called the label of the examples. As it is shown in matrix 1.2.5, every training example in the data-set $(x_1, x_2, x_3, ..., x_k)$ is transformed into a vector $(x_1, x_2, x_3, ..., x_n)$ of $n$ values, equal to the number of the features and the label of the example $y$, which is the variable that we use as a label and consists the predictive outcome.

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2n} \\ x_{31} & x_{32} & x_{33} & \cdots & x_{3n} \\ \vdots & \vdots & \ddots & \cdots & \vdots \\ x_{k1} & x_{k2} & x_{k3} & \cdots & x_{kn} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_k \end{bmatrix}$$

### 1.2.6 Feature selection - construction

Often, information obtained during the acquisition process is not useful due to the noise or the number of missing values. For example, the variable representing donors' creatinine levels after five years has around 60% of its values missed. Imputing such large amounts of missing information implies the introduction of bias, which reduces the reliability of the predictive outcome. Taking this into account together with the low importance of the captured information for short term organ survival , we can omit this feature. On the other hand, feature construction is the process which aims to yield more descriptive information from the raw data. As we will discuss later in chapter 2.2.2, a representative example is the information provided by the date-related variables. These variables cannot stand any useful information alone. However, when we compare them pairwise, we can extract powerful information for the procedure.

### 1.2.7 Model training and Classification

During the classification process, predictive algorithms are employed to generate associations between the selected input data and their labels. This is the training process of the system, where the model learns which characteristics describe each of the label classes. Then, based on a predefined decision threshold, the algorithm assigns the input data into different classes according to the characteristics of each class. In this project we employed Artificial Neural Networks to build a predictive model capable of forecasting the post-operative result of graft failure, discussed in more detail in chapter 3. An increasingly

amount of studies use Artificial Neural Network to construct prognostic and diagnostic models to support clinical decisions with an accuracy higher than the one achieved by physicians [28][29][30][31][32][33].

### 1.2.8 Performance evaluation

Model evaluation is the end stage of the learning procedure. The efficiency of the model to predict the outcome on unseen data shows how accurate the model is. In this project we will use two different measures to evaluate the performance of the model; the accuracy and the loss. In addition, in sections 3.6.1 and 3.6.2 we will perform 10-fold cross validation to assess the ability of our proposed models to generalize the obtained knowledge in unobserved data.

### 1.2.9 Personalized decision support system (PDSS)

Personalized Decision Support System is the health information system that provides knowledge and assists decision making for clinicians [34]. In this study we develop the semantic model that captures the conceptual knowledge for donor-recipient pair in kidney transplantations. This model can be utilized as the building block to construct the PDSS by integrating the results of our predictive models in graft survival and provide clinicians an easily accessible and interpret-able way of these results.

## 1.3 Challenges

Dealing with medical data is always challenging, especially when the goal of the project has an effect on peoples lives. EHRs are usually characterized of their data incompleteness ans inconsistency, which arise the challenge to find the optimal way to deal with the missing information. In addition, we have noticed that one important variable, according to the organ allocation protocol provided by Eurotransplant is missing. This is the donors blood group. However, since we only want to predict graft survival in the current study, this missing information may not impact the accuracy of our predictor. But if the research question of the project was to predict the most compatible donor-recipient pair this missing variable would significantly impact the predictive outcome. Another challenge that we need to tackle in this project is to fine-tune the parameters and the architecture of the ANN to achieve the most precise models for the task.

## 1.4 Research scope and objectives

There are many publications in the literature that are concerned with the various factors that affect graft survival. However, the majority of these studies focuses only on one factor that either has negative effect in the final outcome or positive. In addition, most of the

studies perform statistical analysis on the data that they have collected and they do not really focus on predictive models for individual outcomes.

Based on that fact, the goal of this project is to develop two predictive models that would learn the relationships of the transplantation database and it would efficiently predict the post-operative outcome for the recipient. The first model or the postoperative model would be applied in the whole data-set provided for this project and it will be trained to predict graft survival based on the preoperative and the postoperative characteristics of the donor-recipient pairs. While the second model, or pre-operative model will be trained only on the pre-operative information for the donors and the recipients and it will forecast postoperative graft survival. The predicted outcomes of our approach could then be used to develop a clinical decision support system with high fidelity so that grafts could be allocated to the recipients that show the best survival rate. More precisely in this study we will :

- apply data mining techniques to deal with the incomplete and noisy data provided by EHRs and discuss if medical data are appropriate to apply machine learning technologies,
- develop a predictive model to forecast graft survival from all the information provided for donor and recipients,
- develop a model that is able to predict graft survival based only in the information we have for the recipients and donors before the operation,
- evaluate the accuracy of the two proposed models,
- develop a semantic model that captures the conceptual knowledge from our data-set

### 1.4.1 Research questions

Based on the goals we previously stated, we can formulate the following research questions :

- What data mining techniques can be applied to deal with incomplete and class imbalanced medical data ?
- Are Artificial Neural Networks capable of predicting graft survival, given recipient-donor information ?
- How can we convey the results of the predictive models back to clinicians using semantic modeling ?

## 1.5 Outline

This report is divided into four chapters. In the second chapter we are discussing stepwise the data analysis pipeline. First the data will be explored, providing information for the data acquisition and the correlations among the variables that consist our data-set. In the end of this section we will demonstrate the network that represent the knowledge clinicians have about the transplantation dataset. In addition, we will discuss the preprocessing step we applied and also feature engineering and feature selection. In the following chapter we give ground knowledge about the Artificial Neural Networks (ANNs). Next we will discuss previous application of ANNs in the medical sector. After that, the experimental setup is discuss follow by the results of the pre-operative model and the post-operative model. Chapter 4 introduces the significance of semantic modeling in decision support systems and immediately after the ontology of transplantations is constructed. Finally, in the last chapter we will discuss the conclusions of this study and answer the research questions, but also, we will suggest some potential future work.

# Chapter 2

# Data analysis pipeline

The performance of machine learning applications depends on the quality of the data they are applied to. Data need to be accurate, reliable, complete and interpretable, so that the models can perform more accurate [35]. When a matter of life is involved in the information that the data describe and decisions need to be made based on the efficiency of the predictive model, data quality must be the best possible [36][37][38]. Nevertheless, despite the advantages that the scientific community experiences from the replacement of the traditional charts with Electronic Health Records (EHRs), new challenges are introduced [38]. Well et al claim in their study that the cause of these challenges is due to the fact that data collection initially introduced not for scientific research, but just to make patients profiles easily accessible by other clinicians [38]. In fact, real world raw data despite the tremendous volume appear to be highly susceptible to noise, heterogeneous and incomplete, impacting the quality of the performed models [35][37].

Noisy data could be the result of "fat-finger error" during the gathering procedure, sensor failure or by entering values out of range for a specific measurement [39]. Different departments in the same hospital for example, could record patients history following different protocols, but when these data-sets are integrated, they would result in a messy data-set with duplicated information. While data incompleteness refers to the problem of missed observations in the data-set [36]. Working with health care data is challenging. Data curation and data mining technologies are required to improve data quality and therefore the predicted efficiency of the developed models. In the following paragraphs we first describe our data-set and the data-set constructed out of it for the prognostic model we built in chapter 3. Next we introduce the graphical representation of the variables in our data-set and discuss the correlations among them. In section 2.2 we explain the procedure of data pre-processing, more specifically, the techniques used to impute the missing information and the way we dealt with the outliers. After that, the methods used to construct and select features are debated. Lastly, we conclude with some interesting observations related to this chapter.

## 2.1 Data exploration

In this section we introduce the data-set used in this project and try to understand our data. To better understand the challenges clinicians face when trying to to consider all the factor included in our study, we developed a network, where the nodes represent the variables of the data and the edges the influences among them according to the clinical knowledge. The network revealed the high connected nature of all features showing that clinicians require assistance to deal with all this information involved. At the end of this section we will discuss the correlations among the variables and we will draw some conclusions based on the observations.

### 2.1.1 Data acquisition

Each record in the trasnplantation data-set represents a patients medical operation or experiment contacted [35]. The rows of the data-set correspond to the medical records and the columns to the attributes of that record, or in other words its characteristics. The variables can be divided into three main categories; personal information of the patient, such as age, sex, etc; biochemical measurements which are quantifiable numeric values, for example, systolic blood pressure [38]. The last category contains diagnostic results classified with ontological term, which are sometimes expressed with free text [38].

This study was based on information obtained from the Netherlands Organ Transplantation Registry (NOTR) concerning kidney transplant recipients collected from the eight kidney transplantation centers in the Netherlands that composed the Dutch Transplant foundation [3]. Data collection and distribution occurred according to Euratransplant data policy[40]. The data concerns all recipients who received a graft from a cadaveric donor (Cardiac death donor (DCD) or Brain Death Donor (DBD)) between January $1^{st}$ 1990 and December $31^{st}$ 2017. The data-set contains 10.410 records and 61 attributes, 18 for the donors and 43 for the recipients. The variables represent donor and recipient characteristics, biochemical measurements and the characteristics of each operation. They also include follow-up information of the recipients three months after the transplantation, then a year later and after five years [3]. An overview of the variables contained in the data-set is shown in table 2.1.

Attributes can be divided into four main categories as listed below :

- **Categorical variables:** are features which take values that represent a finite number of categories. For example, donors cause of death are recorded in four different categorical values, namely, stroke, trauma, cardiac arrest and other.
- **Binary variables:** are those categorical attributes that are classified in exactly two categories. Sex for instance has two states, male and female. Another example of boolean variable is recipients graft loss which has two states, yes and no.
- **Numeric variables :** express quantitative measures that take real or integer values and are distinguished into two subcategories.

| Donor | Recipient | | |
|---|---|---|---|
| d_sex | r_sex | r_primary_disease | r_first_dialysis_date_diff |
| d_age | r_age | r_initial_disease_recurrent | r_last_dialysis_technique_cat |
| d_height | r_height | r_pre_emptive_transplant | r_graft_fail_date |
| d_weight | r_weight | r_ischaemic_period_warm_1 | r_seen_max_date_diff |
| d_BMI | r_BMI | r_ischaemic_period_warm_2 | r_transplant_date |
| d_MDRD | r_blood_group | r_ischaemic_period_cold | r_death_date_diff |
| d_hypertension | r_PRA | r_delayed_graft_function | r_days_death |
| d_smoking | r_proteinuria_M3 | r_graftloss | |
| d_cadaveric_type | r_proteinuria_Y1 | r_early_graftloss2 | |
| d_death_cause_cat | r_proteinuria_Y5 | r_graft_fai_cause | |
| d_NHB_cat | r_MDRD_M3 | r_graft_fail_cause_cat | |
| d_diabetes | r_MDRD_Y1 | r_retransplant | |
| d_hypotensive_periods_duration | r_MDRD_Y5 | r_preservation_solution_type_cat | |
| d_admission_date_diff | r_creatinine_M3 | r_dead | |
| d_death_date_diff | r_creatinine_Y1 | r_death_cause | |
| d_creatinine_last | r_creatinine_Y5 | mismatch_DR | |
| d_creatinine_highest | r_combined_transplants | mismatch_A | |
| d_nephrectomy_date_diff | r_follow_up_date | mismatch_B | |

Table 2.1: Donor - recipient attributes

  - **Discrete:** features are those that can have a finite or countable finite number of values, which can be integer or not. A good representative discrete variable is the age, which can take values from 0 to a finite integer number.
  - **Continuous:** attributes are those that are represented by float-pointing numbers, such as the biochemical measurements.

- **Date related variables:** are the data elements that express the date of some event . For example the date when the donor had the nephrectomy, or when the recipients had their first dialysis. Date related variables, are not actually features themselves, however, if these dates are valid, they carry useful information that can be used to construct meaningful features out of them as we will discuss in section 2.2.2.

As we mentioned above, the initial data-set is composed of 10.410 records, however, we decided to exclude the cases where recipients had combined transplantations due to the inherit risk associated with a double transplantation. The same tactic was followed by Port et al in their statistical study about the characteristics of the recipients that reduce transplant survival in [16]. The bar plot in figure 2.1 depicts the distribution of the three types of transplantasions recorded in the data-set. In our experimental data-set we only kept individuals that had either a right or left kidney transplant and we excluded the 94 cases where recipients got both kidneys transplanted.

For the purpose of this project to answer the two research question, we made two separate data-sets. The first data-set is composed of all the preoperative and postoperative attributes provided by the NOTR, so that accurate predictions can be made regarding graft survival, as described above. The second data-set is made only with the characteristics of the donor and the preoperative information available for the recipient. With this data-set,

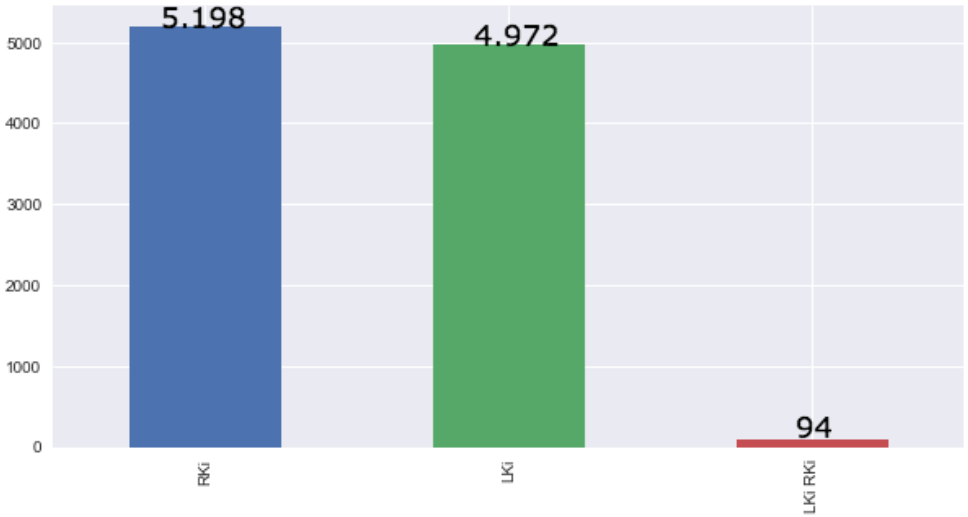



Figure 2.1: The distribution of the label variable.

our developed model would be able to predict graft survival even before the operation takes place. The variables of this data-set are presented in table 2.2

| **Donor** | | **Recipient** | |
|---|---|---|---|
| d_sex | d_cause_death_cat | r_sex | r_primary_disease |
| d_age | d_NHB_cat | r_age | r_initial_disease_recurrent |
| d_height | d_diabetes | r_height | r_pre_emptive_transplant |
| d_weight | d_hypotensive_periods_duration | r_weight | mismatch_DR |
| d_BMI | d_admission_date_diff | r_BMI | mismatch_DR |
| d_MDRD | d_death_date_diff | r_blood_group | mismatch_A |
| d_hypertension | d_creatinine_last | r_PRA | mismatch_B |
| d_smoking | d_creatinine_highest | r_combined_transplants | r_graftloss |
| d_cadaveric_type | d_nephrectomy_date_diff | r_first_dialysis_date_diff | r_last_dialysis_technique_cat |

Table 2.2: Donor - recipient attributes for the second model.

### 2.1.2 Network of variables

In our attempt to understand better the data, we identified the influences of each variable in our data-set on the other features based on clinical knowledge and we constructed a network out of these relations. The network illustrated in figure 2.2 is a graphical representation of the variables that compose our data-set, constructed by the software for network analysis Gephi [41]. Nodes demonstrate the variables included in our data-set, while the edges represent the associations among these variables based on the knowledge of the physicians in our team. The size of the nodes expresses the betweenness centrality of the graph, or in other words the importance of each node in the graph [42]. The greater the betweeness centrality the larger the size of the node in the graph. Another influencing

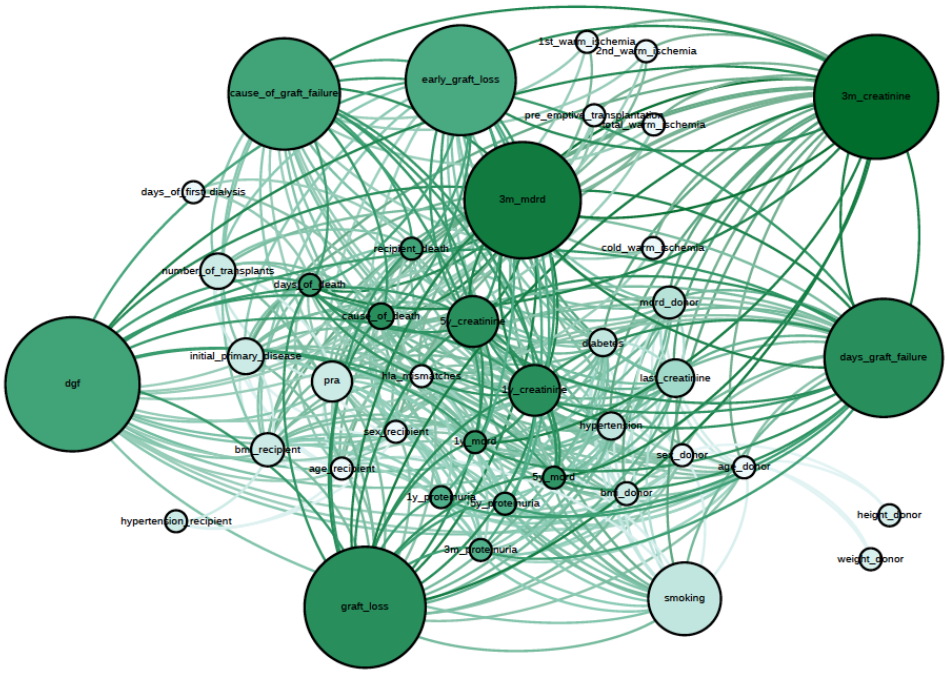



Figure 2.2: A simple network constructed by the variables of our data-set and the predicted assosiations among them.

attribute of a graph is its colors. In our graph, colors have been chosen based on the in-degree of the node, with a lighter color indicating lower in-degree and darker ones show high in-degree. The number of node in the network is equal to the number of the variables used in our prediction model and 361 edges. Looking at the network, we see that some parameters that are believed to have high importance in the success of transplantation have been identified in our network as well. For example, we have seen in the literature that DGF is one of the major parameters that affect graft survival. Likewise, creatinine levels after the operation are believed to be the most effective indicator of kidney functionality and it seems that it plays an import role also in our network. In contrast, other variables such as warm ischemia time and duration on dialysis although they are crucial in organ transplantation, their importance is not identified in our network. This is because the network is simply built based on the influential relations among the variables and not based on real data. Warm ischemia time is defined the time that the tissue(kidney) is not supplied with blood adequately while it remains in body temperature conditions [43]. Warm ischemia is used to describe two different conditions during the operation. Warm ischemia 1 expresses the time when the organ is removed from the ice and is placed in the recipients body until anastomosis takes place and is reperfused. Warm ischemia 2 describes the time from the moment the organ is cross clampimg until cold perfusion starts [43]. While, Cold ischemic time is the period between warm ischemia 1 and warm ischemia 2 and it describes the period that the organ is placed in the ice. However, despite the

tight inter-connectivity among the features, network properties are capable of identifying the important role played by some variables, as it is shown in the graph.

### 2.1.3 Correlations

Another way to better understand our data is to look deeper and identify the hidden dependencies amongst variables. Or in other words, look at the correlations of the data. Correlation coefficient is a measure to evaluate the strength and the direction of the linear association between any pair of variables in a given data-set [44]. The values of this measure range between -1,1. When two variables have correlation coefficient of 1 or -1, are said to be positively / negatively perfect linear related, while variables with 0 correlation coefficient are not linear correlated at all [44].

To estimate the correlations among the variables in our data-set we used Pearsons correlation coefficient. This measure calculates the centered and standardized product of two given variables as shown in equation 2.1.

$$r = \frac{\sum(x_1 - \overline{x})(y_i - \overline{y})}{[\sum(x_1 - \overline{x})^2 \sum(y_1 - \overline{y})^2]^{1/2}} \tag{2.1}$$

The correlations of our data-set are depicted in figure 2.3. Looking at the numbers, we observe that most positive and negative high correlated variables are those which represent opposite concepts. For example recipients that have no graft failure and recipients who are alive are highly positively correlated, with correlation coefficient of 0.79, while recipients that are alive and graft failure are negatively correlated with correlation coefficient -0.79.

| | **no graft failure** |
|---|---|
| **recipient alive** | 0.79 |
| **recipient dead** | -0.79 |

Dead recipients and days of death are also highly positive correlated in contrast with the alive recipients who are negatively correlated. In addition, negatively correlated relationship was captured for the variables that describe the recipients who had preemptive transplantation and they had transplanted their left kidney. Another association that was identified, is between the variable that represents the recipients who do not have a cause of death, so they are alive with the variable that represents the recipients who died with a functioning graft. Furthermore, the engineered variable that represents the days between transplantation and graft failure is negative correlated with the other engineered variable that expresses the period between transplantation and recipients death.

As it was expected, the variables that constructed from other raw variables, such as MDRD[1] and BMI[2] are highly positive correlated with the variables that originated from, namely creatinine levels and height and weight for the BMI.

[1]Glomerular fltration rate
[2]Body Mass Index

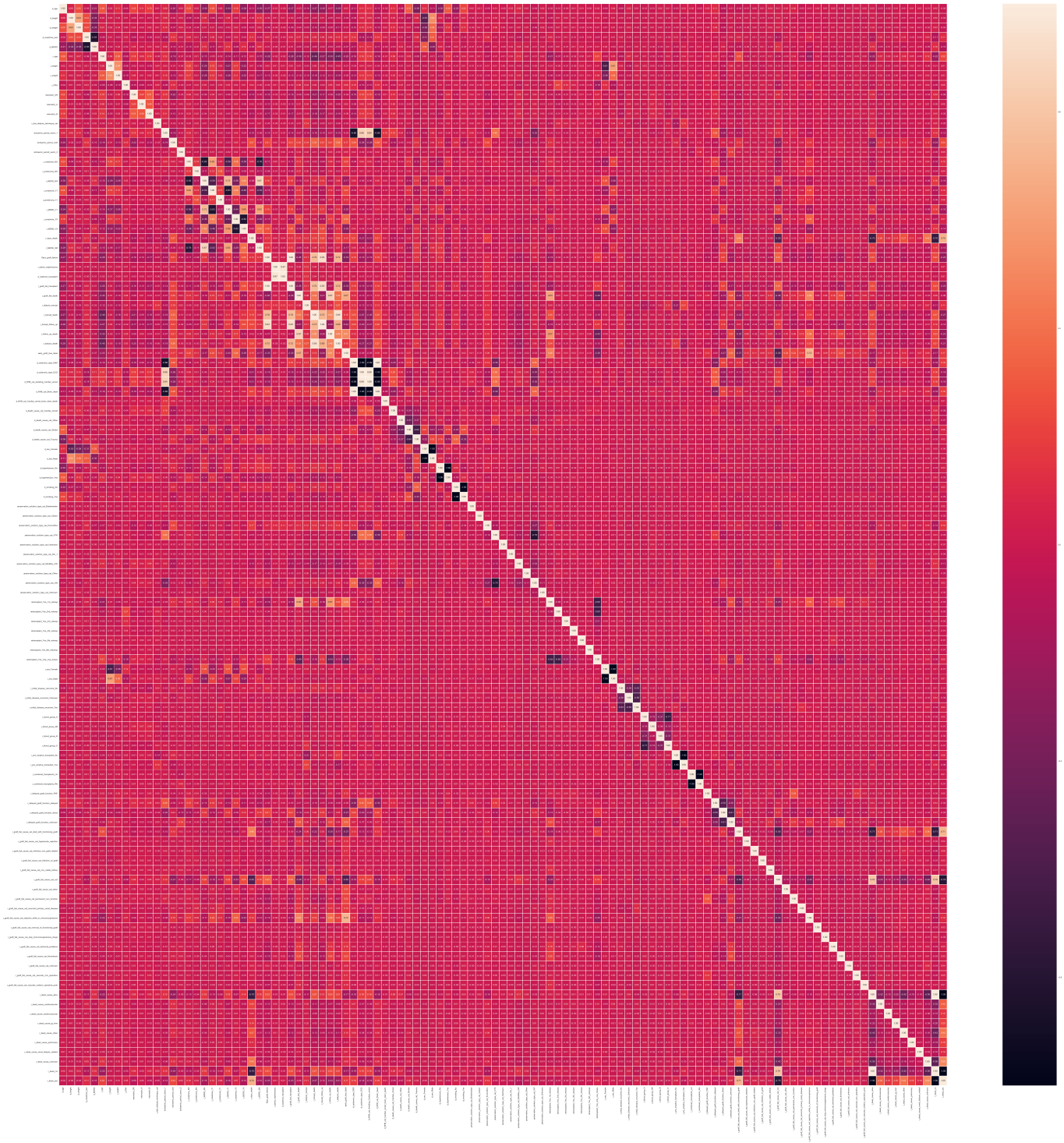


Figure 2.3: Correlation matrix of the data.

We also checked the correlations between the variable *graft loss*, which is used as the

predictive outcome in our model, with the rest of the variables in the data-set. We only found that it has 0.72 correlation coefficient with the variable that represents patients who died because of a rejection while they were under immunosupression treatment.

Another interesting observation in the correlation matrix was the relationship between warm ischemia period and the type of donors cadaveric, which is negatively correlated with DCDs and positively with DBDs. Similarly the negative correlation of the same variable with NHB Brain death and the positive with NHB cardiac arrest. An unexpected correlation was found between female recipients and their height with correlation coefficient of -0.61.

## 2.2 Data preparation

In this section we discuss the methods devised to prepare our data-set and overcome the restrictions that impacts its quality. We start with the data pre-processing techniques we used to detect outliers and fill the missing values. Next, we discuss two feature engineering techniques to construct features from the raw data and in the end of this chapter we focus on feature selection and extraction.

### 2.2.1 Data pre-processing

EHRs appear to be prone to data incompleteness and noise [45][38][36][18]. In addition, ML applications are highly susceptible to messy data that cause a number of problems on their performance [36][38]. First of all, the presence of missing values in the data and values that exceed the valid spectrum of a measurement import bias in the distribution of the data-set which misleads parameter estimation during the training processes of the system but also diminishes the representativity of the example cases [36][45][46]. Moreover, not accurate data may impact the statistical power of the model leading to irrational conclusions[36].

A number of studies focused on identifying the cause of such limitations in data structure and completeness and classified accordingly [38][45][36][47]. In diagnostic studies and clinical trial it is very common for clinicians to leave "blank cells" or in other words to miss some observations in their report when the results of the diagnostic tests are natural especially in cases when there are no signs for further investigation. For instance, when a patient potentially suffer from some disease according to their symptoms, clinicians inspect the cases further. On the other hand, if no symptoms exist, they do not ask for additional tests which result in missing data in the clinical record. Another reason that can cause missing values is the fact that some patients do not complete the trial study and participate in the scheduled follow-ups agreed [45].

To overcome the limitation of data incompleteness and data noise, different approaches have been developed. In the following paragraphs we are going to identify the presence of these problems and discuss the methodology used to overtake them.

### Outliers

Before starting to work with the missing values, it is wise to check the distribution of our data-set and identify potential outliers, to put it differently, values that exceed the expected limits for each variable, both in the minimum and the maximum value [48]. When outlier observation exist in the data-set, the distribution changes and consequently can mislead the predictive model. In order to identify the noise in our data-set we performed basic statistic analysis in our data-set. Table 1 in Appendix A depicts the results of the statistical analysis. For every variable we counted the number of missing values and we computed the mean of its distribution, the standard deviation, the min, the max and the $25^{th}$, $50^{th}$, $75^{th}$ percentiles.The standard deviation of a variable measures how to spread its values are and is calculated by the following formula [35].

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{2.2}$$

The min is the smallest value of the data-set, max is the largest, while the $25^{th}$,$50^{th}$, $75^{th}$ percentiles express the $25^{th}$, $50^{th}$ and the $75^{th}$ value respectively if we divide them by 100. The univariate approach proposed by Cousineau et al in [48] is then applied to detect noisy observations. In this method the criterion to indicate noisy observations is to find the minimum and the maximum values of the variable that are at least four times the standard deviation away from the mean, as described in equation 2.3 [48].

$$outlier_i \ \leq \ \mu_i - (4 \ \times \ \sigma_i) \ or \ outlier_i \ \geq \ \mu_i + (4 \ \times \ \sigma_i) \tag{2.3}$$

The results of the variables which seem to include noisy data are shown on the table 2.3.

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| **d_NHB_cat** | 3064.0 | 3.987 | 31.155 | 3.0 | 3.0 | 3.0 | 3.0 | 999.0 |
| **d_BMI** | 9712.0 | 24.810 | 4.304 | 9.693 | 22.491 | 24.489 | 26.573 | 66.597 |
| **d_creatinine_last** | 10363.0 | 105.005 | 429.878 | 0.09 | 56.0 | 72.0 | 92.0 | 13260.0 |
| **d_MDRD** | 10150.0 | 101.875 | 44.563 | 4.214 | 72.423 | 95.181 | 122.349 | 579.90 |
| **r_BMI** | 8038.0 | 25.168 | 4.386 | 10.810 | 22.052 | 24.690 | 27.770 | 48.442 |
| **r_PRA** | 10407.0 | 6.590 | 18.290 | 0.0 | 0.0 | 0.0 | 2.0 | 100.0 |
| **ischaemic_period_warm_1** | 9563.0 | 5.819 | 9.498 | 0.0 | 0.0 | 0.0 | 13.0 | 90.0 |
| **ischaemic_period_cold** | 9749.0 | 1243.375 | 479.997 | 0.0 | 890.0 | 1198.0 | 1521.0 | 3211.0 |
| **ischaemic_period_warm_2** | 9654.0 | 34.776 | 12.898 | 0.0 | 26.0 | 33.0 | 40.0 | 180.0 |
| **r_creatinine_M3** | 8313.0 | 160.092 | 87.980 | 18.0 | 113.0 | 141.0 | 181.0 | 2373.0 |
| **r_proteinuria_M3** | 6433.0 | 0.296 | 1.451 | -1.0 | 0.1 | 0.1 | 0.3 | 66.0 |
| **r_MDRD_M3** | 8305.0 | 46.206 | 19.413 | 4.70 | 33.369 | 44.153 | 56.502 | 212.073 |
| **r_creatinine_Y1** | 7551.0 | 148.609 | 69.500 | 41.0 | 108.0 | 134.0 | 170.5 | 1377.0 |
| **r_proteinuria_Y1** | 5742.0 | 0.325 | 2.170 | 0.0 | 0.0 | 0.1 | 0.2 | 99.0 |
| **r_MDRD_Y1** | 7547.0 | 48.672 | 19.028 | 5.199 | 35.776 | 46.594 | 59.383 | 193.296 |
| **r_proteinuria_Y5** | 3100.0 | 0.325 | 1.729 | 0.0 | 0.0 | 0.1 | 0.3 | 90.0 |
| **r_MDRD_Y5** | 4497.0 | 48.786 | 20.310 | 5.094 | 34.201 | 46.745 | 60.932 | 165.753 |

Table 2.3: Statistics for all the variables whose value goes beyond the expected range. For each variable the non missing values are counted, and the mean, standard deviation, min , max, $25^{th}$,$50^{th}$ and $75^{th}$ percentile are calculated.

The investigation to detect outliers in the categorical variables was made by checking the distribution of each variable to see if there are values out of the expected range. For example the *d NHB cat* variable has three values: 1, 2 and 999 indicating cardiac arrest brain stem death, awaiting cardiac arrest and unknown respectively. The 3 cases assigned the 999 value are replaced with *NaN* as missing values. Donor and recipient Body Mass Index (BMI) are two other variables that seem to be consisted of noise. The problem with the outliers in these variables will be solved by recalculating these variables from the weight and the height as will be discussed in the following section.

Panel Reactive Antibodies (PRA) is an immunological test to measure the presence of specific antigens on the recipient's blood. According to the rule for the outliers described in formula 2.3, values over 77 are replaced with *NaN*. Similarly, for the variables that describe donors last creatinine, recipients creatinine after three months, one year and five years, recipients proteinuria after three months, a year and five years. We calculated the max and the min values according to the rule and we replaced values that exceed it with *NaN*.

Ischemic times seems to involve noisy values as well. These times can vary significantly due to the various techniques used by the surgeons and the distances between the transplantation centers of the recipient the donor. According to Nabreska organ recovery cold ischemiac time cannot exceed 72 hours(in case the kidney is placed on a perfusion pump)[3]. Taking this into account and looking at the values that according to our definition of outliers should be omitted, we decided to not consider these ischemic periods as outliers.

### Missing values

Imputing the missing values of a data-set, consists an extremely challenging task in ML. In order to appropriately fill the missing observations, the nature of the information captured in the variable needs to be considered. According to that, decisions need to be made in the most accurate way for fulfill each individual variable. Filling the missing values is a task that attracted the scientific interest the last decades after the development of the sophisticated predictive models used today. Early studies on medical data were only focusing on statistical analysis of the data which was not affected from the absence of information in the data-set [49][50]. However, taking into consideration the bias and the distortion imported in the data-set from the missing values and the incorrect imputations and the impact in the predictive result in ML, diverse approaches have been developed on how to deal with these issues.

A general approach suggested in the literature is to omit all the examples where missing information is involved [38]. This approach can result in the loss of significant amounts of information if the proportion of the missing values is high. Another approach commonly used is that missing values of a variable are replaced with a single value [38][37][51][39]. This single imputation value could be the mean of the feature or its median or mode. This

[3]Source : http://www.nedonation.org/donation-guide/organ/acceptable-ischemic-times

technique is believed to reduce the variance of the data and lowers down the predictive error [37][51].

Other more sophisticated methods suggest replacing the missing data with multiple values in order to keep the variability rate of the data high, while the distribution of the data remains unchanged [51][37]. Such methods produce unbiased results and sufficient standard errors [47]. In this study we employed multiple conditional mean imputation. This method replaces the missing values with the mean of a selected population of the variable according to the criteria set [52][39][53]. In other words, we create subgroups of the variable and we calculate the mean of the subgroup and we impute it on the missing values. Aste et al [49] and Garcia et al [39] in their reviewing studies for sophisticated technologies to engineer missing data, suggest more heuristic ML methods such as the $k$- nearest neighbour algorithm, ANNs and the Expectation maximization algorithm (EM). These approaches show efficient performance in some data-sets and outperform other methods[39].

Since we already discussed the potential reasons of missing values in the data-set and the methods that can be used to impute them, in this section we are going to identify the missing values in the transplantations data-set and figure out how we can impute them. Our data-set is composed of 666.304 observations (10.411 examples x 64 features) where 111.533 values are missing (16,74%). Before starting to detect the missing values, we check the data-set for duplicated records. However, in our data-set there are no such examples. Next, we investigate if there are any records or variables that have all their observations missing. We only found one example where all observations were null and we excluded it from the data-set (example 10.411). Chakraborty et al proposed in their study that examples with more than 80% of observations missed can be dropped out without importing bias in the data [45]. We adopted this idea, but there were no records in our data-set that had at least 13 non-missing values.

The stacked plot in figure 2.4 illustrates the distribution of missing and non missing values per variable. As we can see some variables have many missing values, while some other have few or none.

The variables which show the highest rate in missing values are donors NHB category, donors hypertensive periods and recipients proteinuria after five years. Due to the high rate of missing values in the two latter variables, we decided to omit them in our data-set to avoid the potential imposed bias introduced. The high rate of missing values in NHB category for donors (70%) has a reasonable explanation. This variable concerns only donors that were cardiac dead (DCD), patients whose heart stopped beating and indicates how the patients died. This implies that the missing values refer to donors who were brain dead (DBD) or in other words, patients whose brain activity stopped.

As we already mentioned, the information captured by every variable has different nature which implies that we need to tackle them individually and apply different methods to replace the missing data. However, we try to group similar cases and present them collectively.

Before going into the filling of the missing values we will discuss the variables we constructed to aid us to impute the missing information more accurately and import as less bias as possible. The idea behind this practice is to group the values of some variables

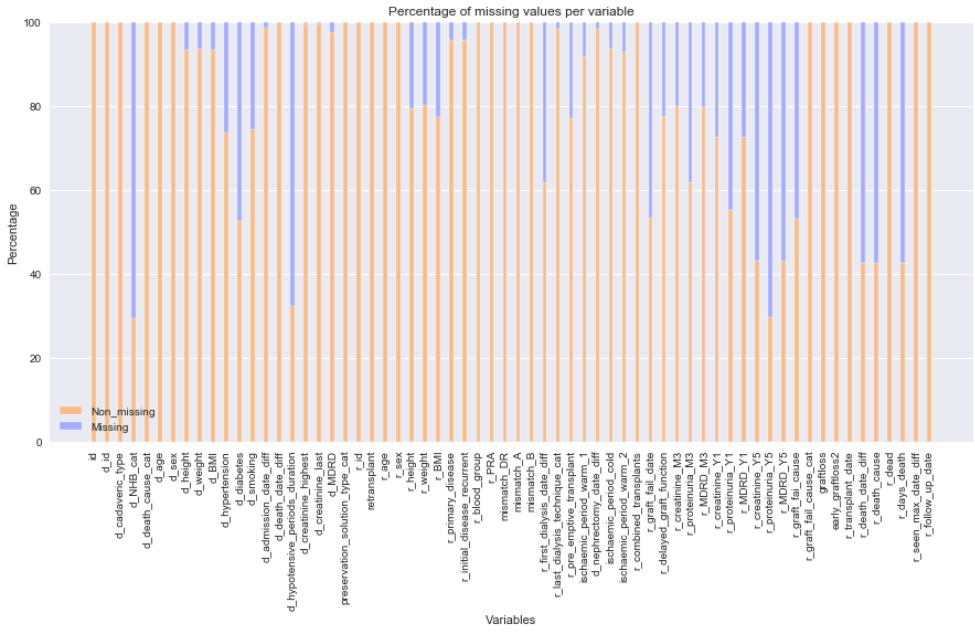


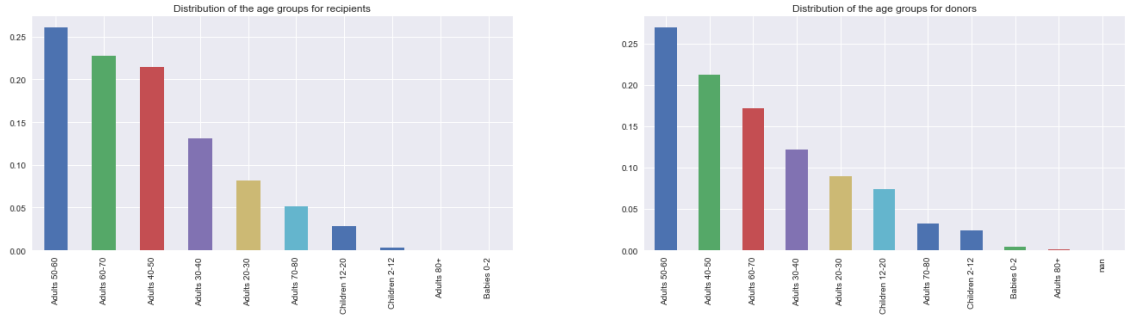

Figure 2.4: Percentage of missing and non missing values per variable.

and during the imputation process to use the mean of a group of similar cases to fill the missing values. The grouped variables we constructed are the following: the grouped age of the donors, the grouped height for the donor, the grouped weight for the donor and the respective variables for the recipients. In the grouped age variable, we grouped the ages of the patients into 10 distinct groups, namely from 0-2, 2-12, 12-20, 20-30, 30-40, 40-50, 50-60, 60-70, 70-80 and 80+ years old. Similarly, for the weight groups we created seven groups, one for people with 4-20 kgs, then for 20-40, 40-60, 60-80, 80-100, 100-130 and the last one for people weighting between 130-185 kgs. The height groups were made for ranges from 60-80, 80-100, 100-130, 130-150, 150-170, 170-180, 180-190, 190-210cm. The distribution of the population into these groups for both donors and recipients is shown in figures 2.5, 2.7, 2.6 . These groups are going to serve as a ground evidence during the imputation process by assigning the mean of a specific subgroup of the population into that groups missing information.

The replacing of the missing values with the variables that can be fixed with calculations from other variables. BMI is one of them. Once we know all the weights and the heights for the recipients and the donors, we recalculate the BMI and fill all the missing values existing in these variables. Formula 2.4 expresses how the BMI is computed.

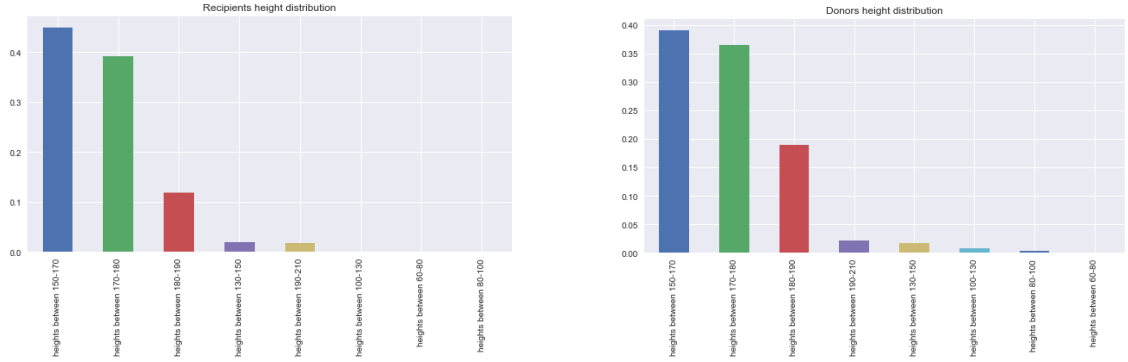
$$BMI = \frac{weight\ in\ kg}{(height\ in\ m)^2} \tag{2.4}$$

We applied the same technique to obtain the missing information for the MDRD variable. MDRD estimates the glomerular filtration rate for patients with chronic renal disease. There are two different formulas used to compute this variable, one for male patients shown




(a) Recipients age groups (b) Donors age groups

Figure 2.5: The distribution of the age groups for donors and recipients.




(a) Recipients height groups (b) Donors height groups

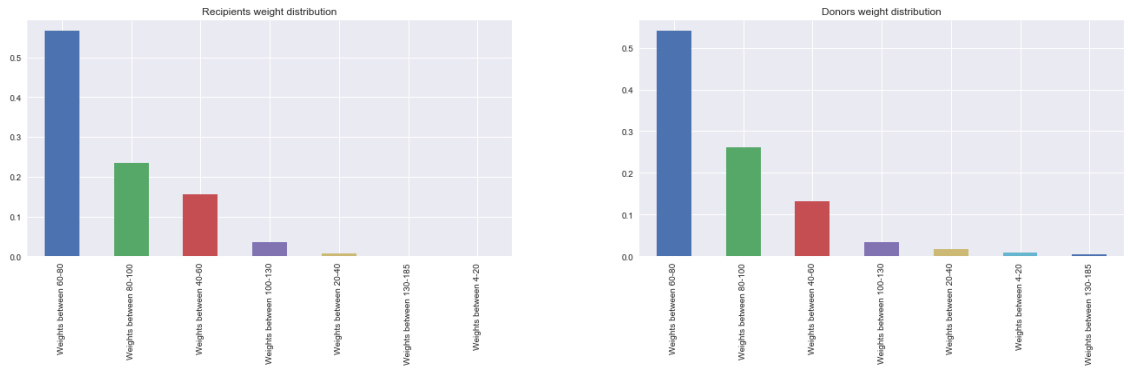
Figure 2.6: The distribution of the height groups for donors and recipients.

in equation 2.5 and the equivalent for female patients as described in formula 2.6. Applying these two formulas solved the problem with the outliers in these variables together with the problem of missing values.

$$MDRD_{male} = 186\ \frac{Last\ plasma\ creatinine_{\mu mol/l}}{88.4})^{-1.154}\ (Donor\ age\ yrs)^{-0.203} \tag{2.5}$$

$$MDRD_{female} = 186\ \frac{Last\ plasma\ creatinine_{\mu mol/l}}{88.4})^{-1.154}\ (Donor\ age\ yrs)^{-0.203}\ 0.742 \tag{2.6}$$

Next, we will give some examples of how we applied the multiple conditional mean to fill the missing values. The weight of donors and recipients had 796 and 2.066 missing values respectively. To fill them in, we constructed a rule which takes the mean of a subgroup of the total population based on the sex, the age groups and the height groups. For example, for a man, whose weight is missing, the algorithm will impute this value with the mean




(a) Recipients weight groups

(b) Donors weight groups

Figure 2.7: The distribution of the weight groups for donors and recipients.

weight of all the men that belong to the same age group and in the same height group. In a similar way, a woman whose weight is absent would be filled by the mean of the weight of all the women that belong to the same age group and the same height group. In the same way, we estimated the missing heights for both donors and recipients, we grouped the examples according to the sex, the grouped ages and the grouped weights and we used the mean of this subgroup.

For the 2.919 cases where the variable describing whether a donor was suffering from hypertension, we applied the same technique. The criteria in this case were the donor's sex, the age groups, the BMI and the variable expressing if the donor was suffering also from diabetes. However, four examples were not imputed with the method discussed above. For this reason, we made the assumption that it is more likely for clinicians to not record this variable when someone does not suffer from the disease and we assigned these four examples in the class indicating that the patient does not suffer from diabetes. The rules applied concerning the variable describing whether a donor was a smoker or not are the sex of the donor, the age groups and the variable that expresses the cause of donors death. In this way the 2.666 missing observations were filled.

The same technique was employed to fill the absent values for the creatinine levels for both recipients and donors. As we previously mentioned, to exclude the noisy examples in the data-set we removed all the values below zero and above 150. For donors creatinine the rule was formed by the sex, the age groups and the weight groups. However, the rule for the recipients is a bit more complex. According to the clinicians knowledge, the missing values are filled with the mean of the subgroup of the sex, age groups, weight groups, donors cadaveric type, delayed graft function, graft loss and creatinine levels after three months (when we were filling the missing values for creatinine after one year) and creatinine after one year (when we were imputing creatinine after three months). Graft loss is the medical condition when the implanted tissue is not functioning any more. Likewise, the 452 case where initial disease re-currency was not recorded for the recipients, we grouped the cases according to the sex, age groups, hypertension and diabetes.

Recipients last dialysis methods was a bit more complicated to estimate. Since we know that preemptive transplantation is referring to transplantations where recipients got transplanted before undergoing on dialysis [54] , we first checked which of the missing cases belong to this category and we found that 10 of the 182 missing examples were pre-emptive. To fill the rest of the missing information in this feature, we constructed an assisting variable that defines the year of the transplantation. This is because we noticed that across the years the methods used in dialysis vary. Then we grouped the missing cases according to the sex, age groups, BMI and the year of the transplantation and we calculated the mean of each subgroup.

The variable that defines which recipients had pre-emptive transplantation, had also a value 'unknown' which we replaced with NaN, like it is missing. Next we checked the cases where recipients did not undergo dialysis and this criterion filled the 2.399 missing values. The variable expressing DGF as we mentioned already is of high importance in graft survival. The missing rate of this variable is quite high, with 2.291 missing values (25%). To fill this missing information we first inspect the cases where the cause of graft failure is zero, or in other words no graft failure. There were 540 such cases in our data-set. To avoid importing bias in this variable, we assigned the rest of the missing values in the value 'unknown'.

The graft loss variable is the most important one in our data-set, since this variable is defined as our predictive outcomes. This means that there is no chance for mistakes in this feature. For this reason, the only example that was missing graft loss was discarded from our data-set.

Recipients death cause shows a 60% of information loss. However, the problem was solved by checking which patients where still alive and consequently did not have a death cause. Three examples remained with no death cause and we assigned them to the class of the alive people, since it is more likely for alive people to not be recorded for such a variable.

The process of filling the missing information for the date-related variables is a bit different. The missing information in a date related variable cannot easily predicted based on other features of the data-set. The only way to approach this is to use the dates of other events with which there is a reasonable relation. First, we checked which patients had a preemptive transplantation, which is a good indicator for the cases that did not have to undergo dialysis. 164 such cases where identified. However, for the remaining 3.373 examples we did not have any other useful information than the day of nephrectomy. Since we know that the average period of life for a patient with end-stage kidney disease is around five years, we replaced the missing values with the date that is 1.800 days before the nephrectomy. Similarly for the nephrectomy date variable, based on the assumption that nephrectomy happens more or less the day that the donor passes away. For the 148 cases where this information was missed, we replaced the day of nephrectomy with the day of donors death. We followed exactly the same tactic for the date of admission. Based on the date of transplantation, we calcutated the average period which recipients had to wait from the day of their admission until the day they got transplanted and we used it for the cases where this information was missed. For the cases where initial graft failure date was

missing, we checked if the recipient had graft loss at all and we replaced this dates with the day of today, since we do not know if these people are still live with working graft or if they are under dialysis.

### 2.2.2 Feature engineering

Feature engineering, also called feature construction consists one of the concepts applied during the data preparation process in developing artificial intelligent systems. The main idea behind feature construction is to develop new attributes from the raw data by using its domain knowledge [55][56]. In other words, ML derived features exploit the knowledge provided from features that are meaningless otherwise and transform the information in such a representation that can improve the performance of the system, as well as, to improve the computational complexity [55][57]. Zhao et al in their study about bankrupt prediction, analyze the importance of using engineered features instead of raw variables. The authors built four different predictive models which were applied into two data-sets: one with the original raw data and the second one which also included constructed variables. Concluding their study, Zhao et al highlighted the importance of using high-level attributes as equal as parameter optimization and model selection, in order to have a high accurate model. In the following paragraphs, we are going to discuss how feature engineering was applied in this project.

#### Variable construction from raw data

In our data-set we have 7 variables that define the dates of some events related to each operation. However, these variables alone, have no importance or knowledge to provide in the learning system. They are just numbers with no meaning. On the other hand, using pairs of these date related variables we can mine beneficial information that would support the learning process to make the correct decisions. For example the date of donors nephrectomy, has no other notion that just a date. Nonetheless, if we subtract that date from the date when the recipient got transplanted we build a feature that may be substantial for the learning process. Another significant feature we constructed is the period between the day the recipient got transplanted until the day the organ started mulfunctioning. We constructed eight more such features which are described in table 2.4.

#### Feature construction using formulas

In the previous section (2.2.1) when we were discussing the ways used to impute the missing values, we mentioned that some formulas were employed to achieve that. The BMI and the MDRD for both recipients and donors were the variables that used this feature engineering approach. As equation 2.4 shows, the height and the weight for the patients were used to construct the BMI feature. While we used the creatinine variable in the formulas 2.5 and 2.6 to calculate the MDRD for men and women respectively.

| New variable | Explanation |
|---|---|
| D admission - D nephrectomy | Day between donors admission until the nephrectomy |
| D nephrectomy - R transplantation | Period between donor nephrectomy until donors transplantation |
| R graft failure - R transplantation | Days from transplantation until graft failure |
| R graft failure - R death | Days from graft failure until death |
| R dialysis - R transplantation | Period the recipient underwent dialysis |
| R transplantation - R death | Period that recipient survived after transplantation |
| R transplantation - R follow up | Period from the transplantation until the last follow up |
| R dialysis - R death | Period from when the patient diagnosed with E.S.K.D till death |
| R follow up - R death | Period between last follow up and death |

Table 2.4: Feature engineering. Nine new features were constructed based on the dates describing various events during the transplantation procedure.

### Categorical variable binarization

The third and last technique we used to engineer new variables concerns categorical variables. Categorical variables usually carry a lot of useful information and they have the advantage of decoding the raw variables into categories. However, in machine learning projects, this information is difficult to decipher. Replacing the categories with numerical values is the solution to this issues. But the problem still exist and it may be even worse. For instance the variable sex can take two values, 0 for male and 1 for female. What the learning algorithm learns from these two values is that '1s' are more important. If the variable had more categories, lets say 16 for example, as many classes have in our data-set the variable recipients cause of death, is cause of death 16 more dangerous than cause of death 0 ? No, it is not, but this is how a ML argorithm perceives the different values that are assigned in the data-set. To conclude, numeric values that are assigned into categorical variables often mislead ML learning systems and negatively affect the learning process [58]. The solution on this problem was given by a feature engineering method which converts categorical variables into binary[58]. For the sex variable we mentioned earlier we will create two new variables,sex_male and sex_female which will have binary values, '1s' when it is true and '0s' when it is false. An example of a dummy variable as it is often called in ML is given in table 2.5. The algorithm creates $k$ new variables equal to the number of the classes of the variable and only one of the new constructed variables will be assinged with an one in each example[58].

In addition, the sum of the values of each variable should be equal to the distribution of the class that variable represents to. In the given example for instance, the sum of the values in dummy variable sex_male should be 2 and the sum of sex_female has to be 3, equal to the distribution of each class. A detailed list with all the dummy variables engineered and the "raw" variables that they came from are depicted at Appendix 5.2 in table 2. From the 20 categorical features in our data-set, we constructed 43 new binary variables, that is as many as the number of the classes of all these variables.

| sex | sex_male | sex_female |
|---|---|---|
| male | 1 | 0 |
| female | 0 | 1 |
| female | 0 | 1 |
| male | 1 | 0 |
| female | 0 | 1 |

Table 2.5: Example dummy variable. The derived variables have only 0s and 1s.

### 2.2.3 Feature selection and extraction

In the previous sections we discussed how our data were acquired and the processes employed to improve their quality. However, it is objective how adequate and relevant is a redundant amount of information to represent an object and how significant this information is for the research question [59]. Feature selection in ML applications is to select the variables that represent the data the best while the performance of the system is improved both from a computational cost and efficiency perspective[60][59].

For this reason, arises the problem that we need to use only a number of these features obtained from our data that are relevant in constructing a model which is easily interpreted, reduces the computational cost, avoids the curse of dimensionality and last but not least, increases the variance and consequently enhances the efficiency of the model[61]. Promising techniques to reduce redundancy and irrelevancy among the data without incurring much loss of information are feature selection and extraction, which creates new features from functions of the original features or returns a subset of the features respectively[62].

For this purpose we decided to omit the variables that had the highest rate of missing values, as we already mentioned previously. These variables are the recipient's creatinine measured after 5 years of the transplantation and the duration of hypertensive periods that donors experienced. We believe that in spite the fact that we imputed the missing values, a high bias could be imported and impact the classification accuracy. In addition, we excluded from the data-set the variable early graft loss which strongly indicates the outcome of the model. Early graft loss is the graft loss that happens within 30 days after the transplantation. To reduce the dimensionality of our data-set, we could also remove the variables used to calculate BMI and MDRD, namely the weight, heigh and creatinine for both donors and recipients. This is because the computed features are of a higher level of representation, while the individual variables do not offer any extra information in the presence of BMI and MDRD. However, since we applied the dimentionality reduction Principal Component Analysis (PRA) described in the following paragraph, we decided to not omit them.

#### Principal Component Analysis (PCA)

Principal component analysis (PCA) is a method in modern data analysis used to reduce multidimensional data sets into an optimal number of features that almost equally

represent the data[63][64]. The wide-ranged applicability of this method in different fields such as neuroscience and computer graphics rises from its ability to extract relevant information from high dimensional data-sets by calculating the eigenvalue decomposition of a data covariance matrix or singular value decomposition of a data matrix, usually after computing the mean centering from each attribute in the data[63]. As a consequence, a set of observations of possibly correlated features is converted into a set of values of linearly uncorrelated features, called the principal components, which helps to reveal the sometimes hidden, simplified structures that often underlie it and improve models accuracy and performance [64][63].

The plot in figure 2.8 shows the cumulative explained variance of the data-set in function with the number of components involved. From the graph we can see that at around 80 components, the variance reaches a peak. In other words, when we apply PCA the number of the initial features can be reduced from 114 to 80 components that comprise most of the descriptive information contained in the data-set.

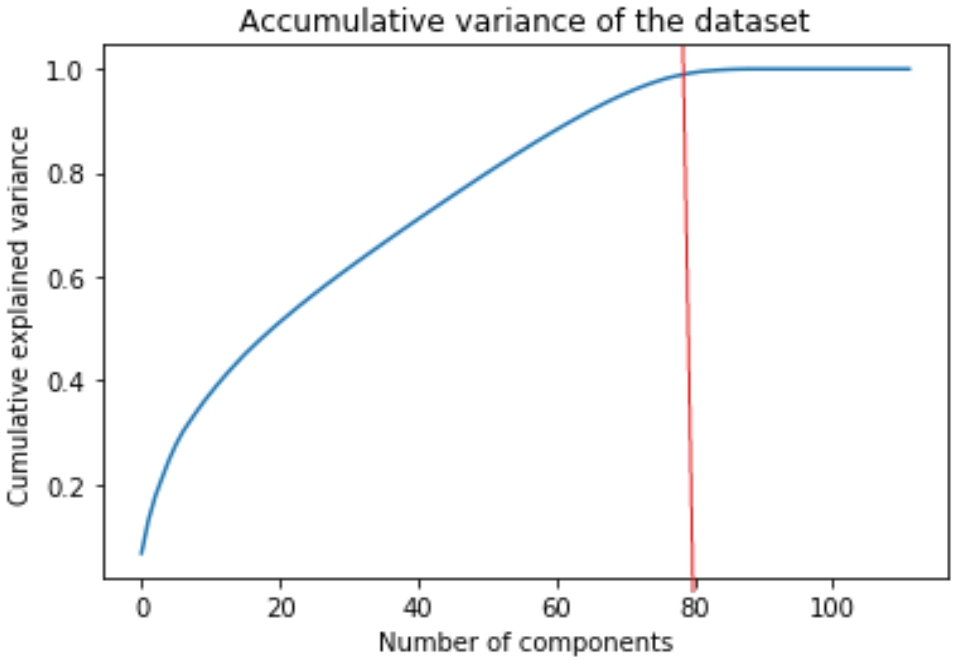



Figure 2.8: Accumulative variance in the data-set.

### 2.2.4 Summary

At the beginning of this chapter we introduced the data-set used in this project and we deviced different techniques to understand it better. First we developed the network of the variables that compose our data-set which expresses the knowledge that clinicians have about the data and after that we explored the correlation between the variables. In the second part of the chapter we employed a number of data preparation methods

to improve the quality of the data. We started with detecting the outliers and after we applied two imputation techniques to fill the 111.533 missing values. After applying our imputation techniques 61 values remained in the dataset, which we decided to omit to avoid importing bias. We also employed feature engineering to construct new features from the raw data. First we developed 86 dummy variables that binarize the 20 categorical variables we have in our data-set and after we constructed 9 new features derived from the dates of different events that we present in our dataset. In the end of the chapter we performed feature selection to reduce the dimenensionality of the data-set. The final dataset is composed of 10.277 complete transplantation examples and each example is composed of 80 principal components derived from 112 features in the postoperative model and 34 principal components derived fro the 48 features that form the pre-operative data-set.

# Chapter 3

# Neural Network

## 3.1 Background

Technological advances in computer science that have been developed in recent years, such as sensor technologies, in combination with the explosive use of the World Wide Web have led to an abundance of free available data [65][66]. However, looking into the raw data, it seems daunting to extract manually all the information included on them and make useful observations out of them. This emerges the need of demanding practical applications that are capable of dealing and interpreting the increased amount of data produced and yield meaningful insights[67]. For this reason, scientists introduced Machine Learning (ML) systems to provide the technical basis to extract useful knowledge from the raw data in a comprehensive form to understand the given data better but also to make predictions in new contexts [68]. For this purpose, ML algorithms mainly exploit the knowledge provided by computer science, statistics and engineering applications [65][68].

Many technological inventions that are widely used today have been inspired by natural procedures [69]. The ability of the human brain system to manipulate quickly, precisely and simultaneous highly complex processing tasks to accomplish perceptual tasks triggered by environmental signals have influenced scientists to introduce the intelligent models called Artificial Neural Networks (ANNs) [69][70][71]. Nevertheless, the conceptual knowledge gained by humans in rather simple problems requires a huge amount of data in order machines to perform similarly [72][73]. This is due to the different representations of the same object which human minds perceive in an noncompetitive manner than the machines do, even though sophisticated algorithms have been developed [74].

The reason why ANNs attracted so much attention in comparison with other traditional statistical methods techniques underlies their ability to extract knowledge from existing examples, whose dependencies are undetermined, and generalize this knowledge in unseen cases by making predictions [75]. The high achieved accuracy in predictions proposes them as a suitable system to be utilized in a crucial task such as clinical decision making [76]. Another important aspect of neural networks is their ability to manipulate different kind of data even in the same manner and make useful observations [66].

Breakthrough in neuroscience in the previous century discovered the functionality of the human brain. [77] The brain is the center of the nervous system, which is composed of millions of single neurons that are interconnected and through chemical reactions, information is processed, transmitted and received [71]. The graphical representation in Figure[1] 3.1 illustrates the architecture of such a neuron. The structure of the nerve cell is formed by the cell body, the axon, the dendrites and the synaptic terminals [71]. Dendrites are branch-like structures around the cell body which serve as information receptors from other cells, neurons or environmental signals [78]. The synaptic information received from dendrites is then transmitted to the cell body where it is transformed into an electrical signal via complex chemical processes [70]. The strength of the produced signal is responsible for whether the transmission is going to be fired and propagate the signal to the next neuron, or if it will be silenced and do nothing based on some threshold [71]. Next, the signals are transmitted to synaptic terminals through the more distant parts of the neuron called axons. During the procedure of synapses, the electrical impulses are propagated into other nerve cells through their dendrites. This procedure is triggered repeatedly according to the input signals from other cells and in turn it transmits its output signal to all the other cells that is connected with.

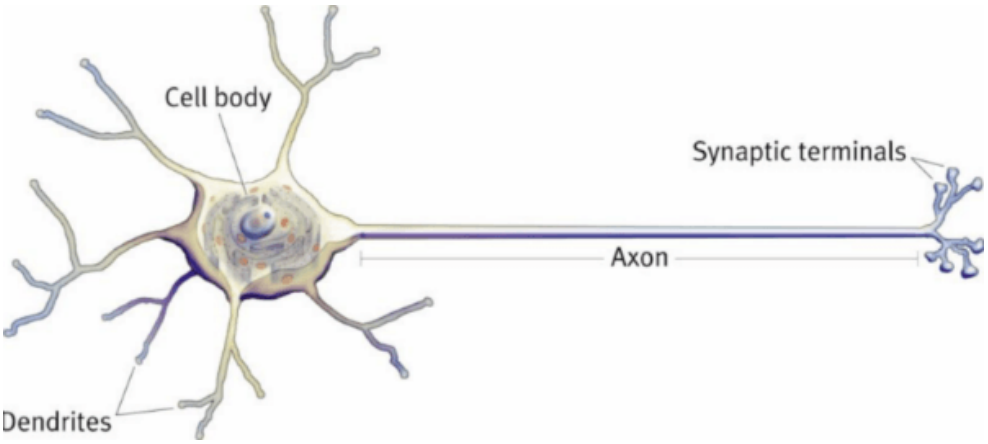



Figure 3.1: Model of the neuron.

In comparison with the biological neural network, artificial neural networks are way less complex. The architecture of ANNs is more abstract, exploiting only the general attributes of their biological relatives to preserve the principles of neuron computations. The fundamental unit for designing neural networks is the neuron which is often called perceptron and is organized in layers. Figure 3.2 depicts a single perceptron model. Every neuron receives an input vector of $n$ signals from other neurons in a previous layer or environmental signals and calculates an output. Each input signal of the perceptron is associated with its own synaptic weight for that specific neuron. This weight expresses the importance of the input value in comparison with the rest of the input values at this

[1] https://www.memorangapp.com/flashcards/104120/The+Nervous+System+I

neuron. For example the input signal at neuron $x1$ is multiplied at synapses with the assigned weight $w1x1$ to determine the strength of this connection in the output. The same procedure is repeated for all the input neurons as figure 3.2 shows.

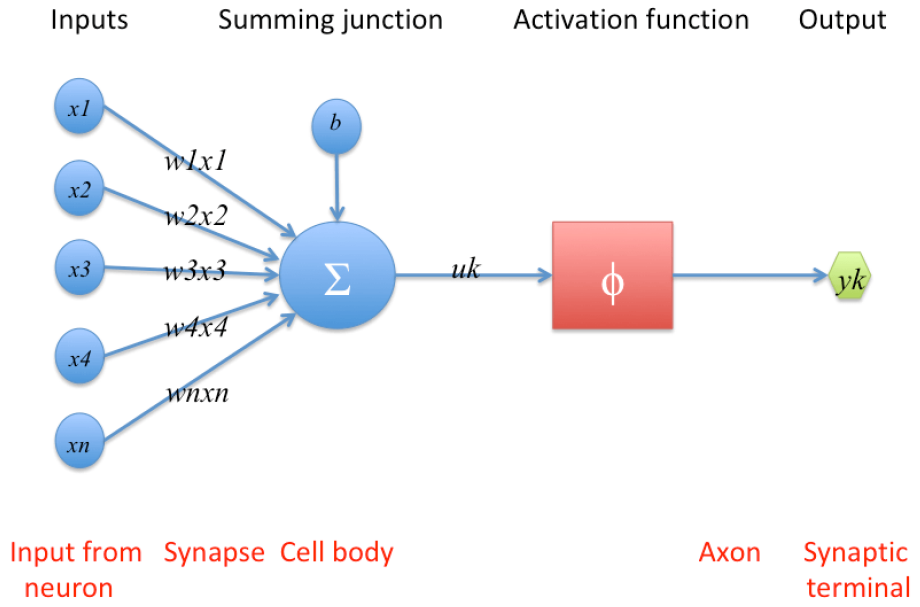



Figure 3.2: Simple perceptron.

Next, the weighted signal of the inputs of the neuron are summed as shown in equation 3.1.

$$u_k = \sum_{i=1}^{n} w_i x_i + b \tag{3.1}$$

The summing junction is a linear function that summarizes the $n$ weighted inputs plus the bias $b$ for k neurons in the network. The additional node of bias takes a constant value, which is usually 1 and plays an important role in order to fit the predicted output better with the data. The activation function at the last step of calculating the output of the neuron limits the output signals in an interval of some finite values [70]. The notion of non-Linearity that describes most of the real world data-sets is introduced by using the activation function. The activation functions $\phi$ perform a certain mathematical computation on its single input value from the junction function and yield the output of the model as it is shown in equation 3.2.

$$y_k = \phi(u_k) \tag{3.2}$$

Two different types of activation functions are identified [70]:

- *Heavyside function* derives a binary output of the neuron according to a certain threshold value. As equation 3.3 shows, the output of the model takes on the value of 1 for all the non-negative induced values and 0 for all the negative ones.

$$\phi(u_k) = \begin{cases} 1 & if\ u_k\ \geq\ 0 \\ 0 & if\ u_k\ <\ 0 \end{cases} \tag{3.3}$$

- *Sigmoid function* the most commonly used in neural networks. This is due to its ability to harmonize linear and non-linear behaviors of the data. This function derives an output between 0 and 1 based on its input signal $u_k$. The most common sigmoid function is the logistic function shown in equation 3.4. $\alpha$ is the slope parameter of its 'S' shaped graph.

$$\phi(u) = \frac{1}{1 + exp(au_k)} \tag{3.4}$$

It is important to note here that in contrast with the Heaviside function, Sigmoid activation function is differentiable. This feature makes the model learn better the data and generalize in new cases.

The learning process of neural network underlies on the best estimated synaptic weight for each input signal, so that it can reproduce the desired output for the training data based on some defined learning rule. Ultimately, the algorithm initializes the weights randomly and then evaluates if the desired output was yielded. If not the weights are updated until it reaches the desired result. In this way, the network is able to uncover linear separable tasks. However, as we will discuss in the following section, there more complex ways to update the weights and be able to tackle more difficult problems.

The knowledge obtained from a single perceptron is limited to some simple domain of some input signals. Nevertheless, when multiple perceptrons are interconnected to each other show their strength to deal with complex problems. In the following section we are going to discuss the architecture of a *Multilayer Perceptron*, where many single perceptons are connected to compose a powerful model for complex tasks [77].

## 3.2 Multilayer Perceptron

The main building block of complex ANN models is the artificial neuron as describe above. Depending on the architecture of the model, we can identify several types of NNs, such as Multilayer perceptons, Convolutional Networks, Recurrent Networks and Autoencoders. However in this project we only focus on Multilayer perceptrons (MLPs).

MPLs can be described as directed graphs where signals are propagated forward and backward and are composed of three different kind of node - neurons, described in the following paragraph [77]. The architecture of the network is divided into multiple layers, each of which consists of multiple neurons and it is fully connected with all the preceding

neurons in the network where all the connections are assigned with synaptic weights. A graphical representation of a multilayer perceptron is illustrated in figure 3.3.

The input neurons take signals vectors that are fed on the network or signals from other neurons and pass the information to subsequent neurons without any computational performance. Next, the neurons in the hidden layers receive signal of the input nodes or neurons from the preceding layers and each of them performs computations as shown in figure 3.2 and described in the formulas 3.1,3.2,3.3,3.4. The hidden notion is introduced due to the fact that the mediated actions that take place in these layers function in a such a way that the input signals are used thoroughly to produced meaningful outputs. It is important to mention here that the architecture of the network is of high importance and it effects its performance. In complex problems for example, the number of hidden layers used to train the algorithm may be higher than those used in simpler tasks. The input layer and the hidden layers include one extra node which represents the induced bias in the network. Then, the output of each neuron is propagated to the next layer and it becomes its input signal. Finally the output neurons in the final layer receive the response signal from the overall computations occurred in the neural network. The example model network presented in figure 3.3 is referred as a $n$ - $m$ - $k$ - 2 network. It is composed of a $n$ dimensional input vector plus one neuron for the bias, two hidden layers, one with $m$ hidden nodes plus one for the bias and the second one with $k$ hidden neurons plus one for the bias and it yields a two-dimensional output.

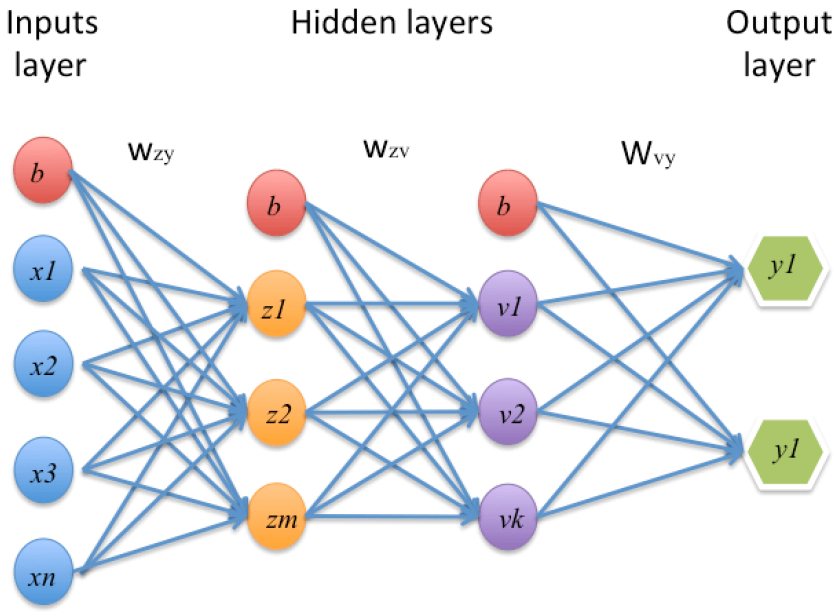



Figure 3.3: Multilayer perceptron

The ability of MLPs to outperform in comparison with other algorithms underlies not

only on their architecture, but also on their qualification to learn and generalize their knowledge. So far, we only mentioned the function signal that is propagated from each input signal and through computations feeds the algorithm with its output on the descending layers until it reaches the output layer. The second signal estimates the contribution of each neuron in order to achieve the desired output, called the error signal. The procedure of measuring the error signal has a backward direction regarding the direction of the function signal. The algorithm starts from the output layers and calculates the contribution of each neuron in the last hidden layer on the error with respect on the synaptic weights. The total error of the network is computed by the formula 3.5 ,

$$\varepsilon = \sum_{c} \sum_{j} (y_{jc} - d_{jc})^2 \tag{3.5}$$

where $c$ is the input - output index, $j$ is the output index, $d$ represents the desired output and $y$ the derived signal. In the next steps, for each derived error, the algorithm estimates the contribution of each neuron in the previous layer, repeating the procedure until it reaches the input layer. Figure 3.4 illustrates the directions of the signal flows in the network.

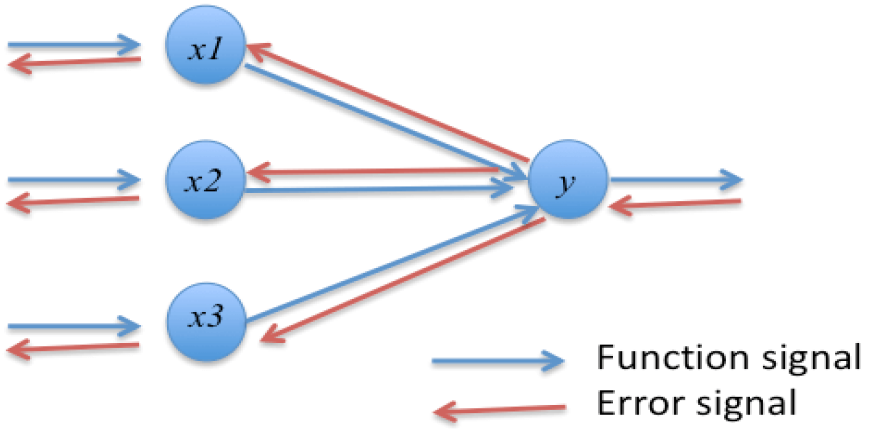



Figure 3.4: Propagation of Function signal and error signal.

The blue arrows represent the forward propagation of the weighted signal, while the red arrows show the backward direction of the error signals. The propagation of the two signal described above, compose the Backpropagation algorithm. The key importance of back propagating is to calculate the error rate through all the neurons in the network, so that we can adjust the synaptic weights in order to minimize error rate between the desired and the real output of the network [79]. This is achieved by calculating the partial derivative of the error $\varepsilon$ with respect to the weights of the network as it is shown in equation 3.6.

$$\frac{\partial \varepsilon}{\partial w_{ij}} = \frac{\partial \varepsilon}{\partial x_i \, \partial y_j} \tag{3.6}$$

This is a powerful property of supervised learning algorithms such backpropagation hold and allow the algorithm to learn from the input - output pairs of the data-set that is trained in. It is important to mention at this point that the initialization of the weights for the input layer takes random values that are uniformly distributed and have mean of 0[70].

The learning process however is a stepwise processes. To achieve that, the leaning rate $\eta$ parameter is introduced. The learning rate indicates the step of the weight update. Equation 3.7 shows the stepwise weight update from one layer to the next one.

$$\Delta w_{ij}^{k} = -\eta \frac{\partial E}{\partial w_{ij}} + \mu \Delta w_{ij}^{k-1} \tag{3.7}$$

To avoid local minima, the momentum $\mu$ term is introduced to stabilize the learning procedure by keeping track of the previous weights used.

Next, to the architecture of the networks, a well structured data-set is required. First of all, a proportional split of the data into training and test set is required. In most of the training set is composed of 70-80% of the data and the testing set of the remaining 20-30% [66][70][69]. Next, the structure of the training set needs to be identified. For each training example in the data-set we need to determine which information consist the features that must be learned during the learning procedure and which are the real outcomes, or the labels for the validation process. In medical diagnosis features mainly consisted of laboratory tests, observations made by clinicians, patient characteristics, clinicopathologic evaluation and information about their treatment. The target variable is the diagnosis for the specific patient given the features of their profile. Using the knowledge obtained from the training data, the algorithm is should be able to make predictions and generalize the rules in new examples. An extra step to improve the quality of the predictions is to standardize the data set in a Gaussian distribution with a mean equal to zero and variance equal to one [80].

In addition, the parameters of the network must be fine-tuned in order to enhance the accuracy of the predictions. The parameters for the research question at hand have to be chosen according to the nature of the data, the size of the data-set. The number of epochs are one them. Epoch is the number of forward and backward passes that the algorithm executes from all the training examples. Sometimes a large number of epochs lead to overfitting and the algorithm malfunctions in unseen data, while a small number causes underfitting. This is often happening because the algorithm learns the training examples by heart instead of learning to make predictions from the features of the data. The batch size parameter refers to the numbers of training examples used in each forward - backward epoch. The training examples are randomly shuffled in each epoch so that more information can be learned. Another crucial parameter in MLPs is the learning rate. This parameter indicates the degree of differentiation during the process of updating the synaptic weights. The activation function, as we mentioned previously, is the function

that computes the weighted sum for each neuron and according to that it makes a decision whether the transition is fire and the signal is propagated or if it remains silent and it is not transmitted. Another parameter that needs to be determined is the loss function. This function calculates the error propagation in the backpropagation phase in order to update the synaptic weights accordingly. Finally, the optimization algorithm has to be chosen. This algorithm is used to minimize the error of produced in the network and update the weights.

## 3.3 Previous work

A growing number of articles in ANNs used for medical decision making attracted scientists interest from the early stages of their development[81]. In combination with the powerful advances in computational abilities the last decades, ANNs were encouraged to support the diagnostic process that physicians need to go through and facilitate their workload to make correct decisions [82][76]. Despite the complexity that is involved in a medical decision, ANNs provide the appropriate ground to develop complex algorithms that can be integrated to enhance the highly required accuracy for predictive inference on such complex reasoning processes [76][13]. Databases with patients history, laboratory tests, physical examinations and medical imaging provide the information required for such artificial systems to learn from the existing cases and generalize for predicting new ones. The function of ANNs in health-care can be delimited into three main categories: prognosis, diagnosis and survival analysis. In the following paragraphs, we are going to report some interesting studies conducted to support medical decision using ANNs.

Er et al conducted a study in Tuberculosis disease diagnosis using MLP [83]. The authors developed two different architectures of a multilayer perceptron, one with one hidden layer and another one with two hidden layers. The data-set used in this study was composed of 150 individual cases where tuberculosis had been diagnosed and each example case had 38 features obtain from laboratory exams. The model with two hidden layers outperformed the previous studies in tuberculosis prediction with an accuracy of 95.08%.

A study in breast cancer prognosis was conducted by Chi et al in [32]. The usability of this model is to predict the probability of a patient who was diagnosed with breast cancer to survive from the disease for an interval of 0 - 10 years. Predicting the survival rate for a patient can then be useful for the clinicians to decide for the best treatment for each individual. The architecture on the MLP used in this study was composed of 30 input nodes, one hidden layer with 20 nodes and an output layer of 20 nodes as well. For the experiments the authors used two different data-sets , both of which are consisted of nuclear morphometric features which were acquired using the Xcyt image analysis program. The first data-set, called Wisconsin Prognostic Breast Cancer (WPBC) is composed of 198 samples while the second one, called Love, has 462 examples. The model shows a 95% confidence in the whole data-set by using cross validation for the predicted and the real outcome.

An additional promising study in cancer diagnosis and classification was conducted in [31]. Khan et al studied four different cancer that occur in chilhood, namely neuroblastoma (NB), rhabdomyosarcoma (RMS), nonHodgkin lymphoma (NHL) and the Ewing family of tumors (EWS). However, despite the fact that these types of cancer look very similar, they require completely different treatments to heal. In order to accurately classify them, the authors of this study propose an ANN model based on the gene expression signatures. Although the size of the data-set is quite small, composed of only 96 examples, the model shows an accuracy of 100 % in classification of these four types of cancer.

Another interesting field that ANNs have application is in medical imaging. Islam et al developed an ANN to predict benign-malignant tumors in mammograms [30]. Using the textile of a mammogram in combination with statistical and textual measures, the authors succeeded to extract interesting features that describe the topology of a tumor that could be fed in an ANN to make the predictions. The architecture of the proposed model is composed of 3 layers which receives 7 input signal for each mammogram and according to the calculations it predicts whether the tumor is benign or malignant. The data-set consists of 322 digital images with a resolution of 1024 $\times$ 1024 pixels. 90.91% of the malignant cases and 83.87% benign were classified correctly.

An early attempt to predict graft survival in liver transplantations using neural networks is presented in [29]. The purpose of this study is to predict the outcomes after the transplantation for appropriate graft allocation. Doyle et al developed a MLP composed of an input layer that feeds the system with 19 attributes obtained from preoperative and postoperative clinical tests, a hidden layer with 2 neurons and the output layer. The algorithm was implemented in a data-set with 155 learn examples in 10 different setups to achieve the optimal model. To proposed model was able to predict correctly liver survival with an accuracy of 96%.

ANNs were employed by Kumar et al to accurately diagnose kidney stones disease. Three different classifiers were developed to predict the creation of kidney stones based on seven symptoms that cause the disease, measured with laboratory tests. The architecture that outperforms among the three is a MLP with two hidden layers trained in 1.000 instances and tested in 150 examples. The model is capable of predicting whether an individual suffers from kidney stone disease with an accuracy of 98%.

Several studies have been focused on kidney transplantation outcomes. Brier et al employed ANNs to predict Delayed Graft Function (DGF) as postoperative result [28]. Two models were developed in this study, a logistic regression model and an ANN. The ANN model was trained in 304 cases, where patients had cadaveric renal transplants and it was able to predict correctly 68% of the examples. In addition, the authors achieved to predict the graft survival for one year without DGF with accuracy 81$\pm$ 3%. A research addressed at early acute graft rejection prediction using biopsies to learn the algorithm [33]. Graft rejection or immunologic rejection is the medical condition when the implanted organ is discarded by recipients immune system and the organ is destroyed [5]. To prevent such situations recipients are prescribed with immunosuppressive drugs. Furness et al employed three small data-sets to export useful information for organ loss, the first one was composed of 100 regulars cases, the second one of 21 well studied critical biopsies and the last one

contained 25 marginal examples, summing up in 146 examples. The network was fed with 12 attributes for each instance and used one hidden layer to perform the computations. This model was able to correctly identify 19/21 (90%) of the crucial biopsies that could cause graft rejection.

## 3.4 Experimental setup

The goal of this project is to build an Artificial Neural Network model capable of predicting graft loss based on the profiles of the recipients and the donors involved. The experiments have been conducted in the high-level programming language Python in combination with the popular deep learning framework Keras [84]. So far, in chapter 2 we described in detail the data-set of the Dutch Organ Transplant Registry of kidney transplantations which will be used in our experiments to provide all the information to train the model and make accurate predictions. In the following paragraphs we are going to discuss the experimental setups used in order to conclude in the best model for the research question.

The first step we need to take before setting up the architecture of the network is build an appropriate feature vector of the data which is going to be used as the outcome for our predictions. The *early_graft_loss* variable from our data-set is used for this purpose. As the plot in figure 3.5 shows, there are 7.500 cases where graft loss doesn't occur, 1.835 where graft loss is happening after 90 days after the transplantation and 928 examples that experienced transplant loss in less than 90 days.

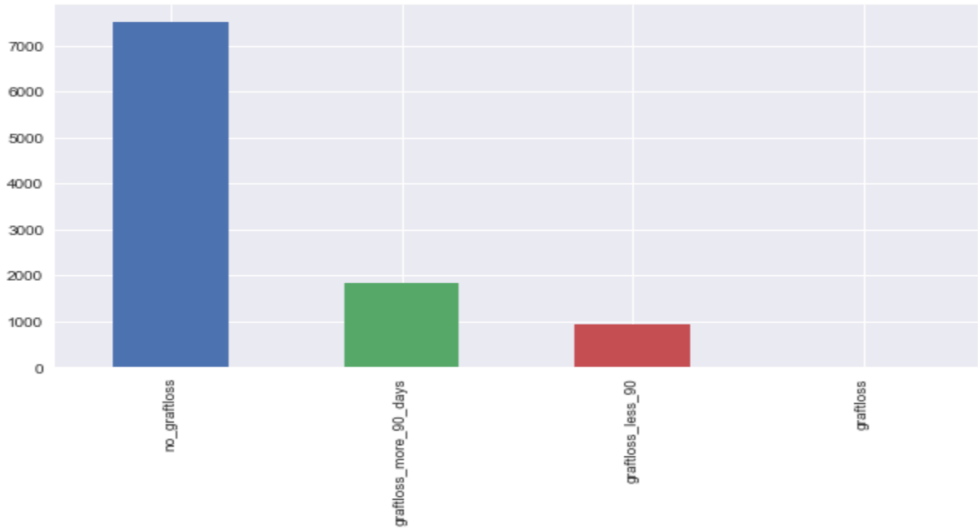



Figure 3.5: Graft loss distribution of the data-set.

Since our research question is to predict the graft loss, we constructed a new binary variable, where cases that do not experience graft loss have a value of zero and those which did with 1. This attribute is going to be used now as the training label in the learning

| | examples | attributes |
|---|---|---|
| **x-training** | 8.221 | 112 |
| **y-training** | 8.221 | 1 |
| **x-testing** | 2.056 | 112 |
| **y-testing** | 2.056 | 1 |

Table 3.1: Shape of the training and testing sets

phase and as the predictive outcome in the testing phase. The distribution of label variable is shown in figure 3.6. As we can see in this figure, the labels are non-uniform distributed with 7.500 0's and 2.764 1's. Later in this section we are going to discuss the influence in the predictive result caused by non-uniform distributed data in the classes of the label variable and how we can overtake it.

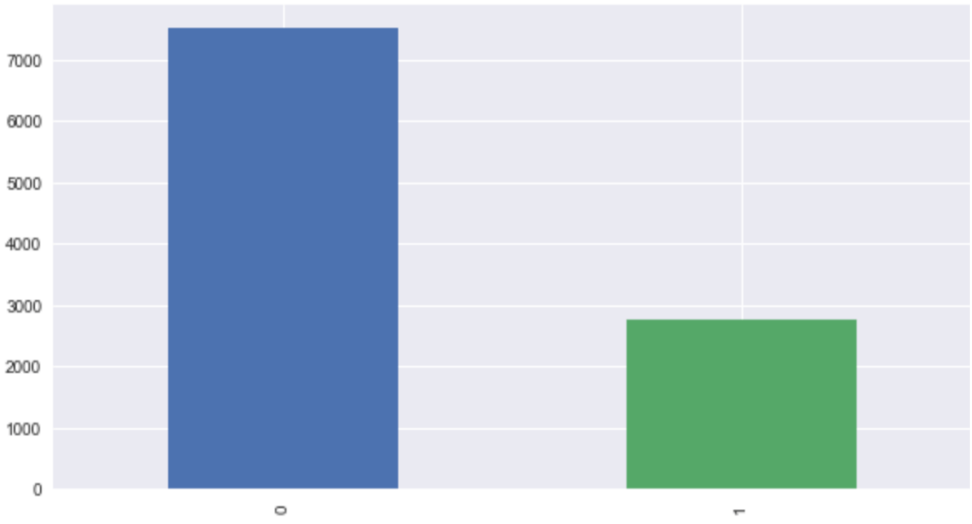



Figure 3.6: The distribution of the label variable.

Once the labels are defined, we need to construct four different subsets of the original data set, two for the training process (data and labels) and two for the testing (data and labels). The training set is composed of 80% of the data and the test set of the remaining 20 % [69]. Table 3.1 shows the shapes of the training and the testing sets. The x-training set is used to learn the system all the required information in order to be able to distinguish the two classes in the x-testing set. The respective y sets are later used to test the system in unseen examples and evaluate its performance based on the percentage of correct predictions.

In section 2.2.1 we show that our data-set is composed from three different types of data, such as continues variables, categorical variables and date related features. Each variable has different distribution and that has a significant negative effect in learning systems. For this reason the training data need to be transformed in a suitable form for the algorithm to

learn without being dominated by attributes with greater variance magnitude than other attributes [85][86][35]. *Standardscaler* is a data scaling method provided by the scikit learn toolbox for Python [86]. This algorithm transforms the feature vector in a Gaussian distribution, such that its distribution has standard deviation of 0ne and a mean of zero. The procedure of standardization is executed in each feature independently as equation 3.8 shows,

$$StandartScaler = \frac{x_i \ - \ mean(x)}{stdv(x)} \tag{3.8}$$

where $x_i$ is an instance of the variable x and *stdv(x)* is the standard deviation of the variable.

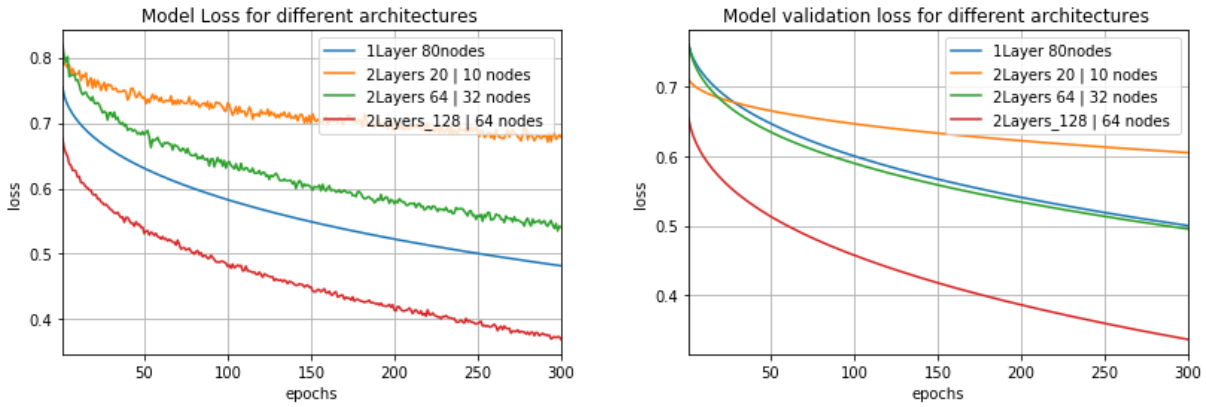
Earlier we mentioned that the distribution of the label variable is critical for the learning procedure. For a binary classification problem, if the distribution is not uniform then the system is dominated towards the value that occurs more often. In other words, the system learns to predict easier the value from the majority in the label vector, under-learning the features from the minority class. In machine learning this problem called imbalance of the training labels [87]. A number of approaches are referred in literature on how to overtake class imbalance [88][89]. One way to successfully leave behind this issue is to assign different weights in each class of the labels during the training process. The weights yielded from this function are then used to emphasize the examples in the minority class and train the algorithm equally in both classes.

### 3.4.1 Post-operative model

Accurate predictions in Neural Networks require a hyperparameter optimization to achieve the most accurate model. At this point we split these parameters into two categories to facilitate understanding. The first category will include parameters concerned in the model construction and those involved in the learning process.

The architecture of the network plays an important role during the learning process. Finding the optimal number of hidden layers and neuron in each layer is crucial. We implemented a number of experiments to identify the best set up for our network based on the loss of the model. Figure 3.7 illustrates the graphical representation of four representative architectures of the network. The set up for the blue lines in plot *a* and *b* were one hidden layer with 80 neurons, the same as the number of the input attributes. Our results confirm Sonetag's study in the number of layer optimization. The authors claim that for a binary MLP classifier, two hidden layers are required to generalize sufficient in new data [90]. The orange lines in figures 3.7 *a* and *b* were obtained with two hidden layers, 20 neurons in the first layer and 10 in the second layer. Two hidden layers with 64 neurons in the first and 32 in the second was the architecture of the network resulted in the green lines of the plot. The last network was composed of two hidden layers with 128 neurons in the first one and 64 in the second one and its loss is depicted with the red lines in the figure. From the four representative architectures we can see in the plots that the one that learns smoother and deeper, while the training loss and the validation loss converge better is the network with

two layers, composed with 128 nodes in the first and 64 in the second, represented with the red line in the plot.




(a) Model loss for varying architectures

(b) Model validation loss for varying architectures

Figure 3.7: Loss for different architectures of the network

Model construction parameters are part of the architecture of the model and include the activation functions used and the dropout parameter. The activation functions used for each layer in the network are responsible for computing the weighted sum of the signal and then decide whether a transition is fired or not. In the model we use two well employed activation functions, namely the rectified linear unit (RELU) and the sigmoid which was described earlier in equation 3.4. RELU is the most used activation function in ML. For a given weighted signal it returns a 0 for negative weighted signal and a positive number else, as equation 3.9 shows[91].

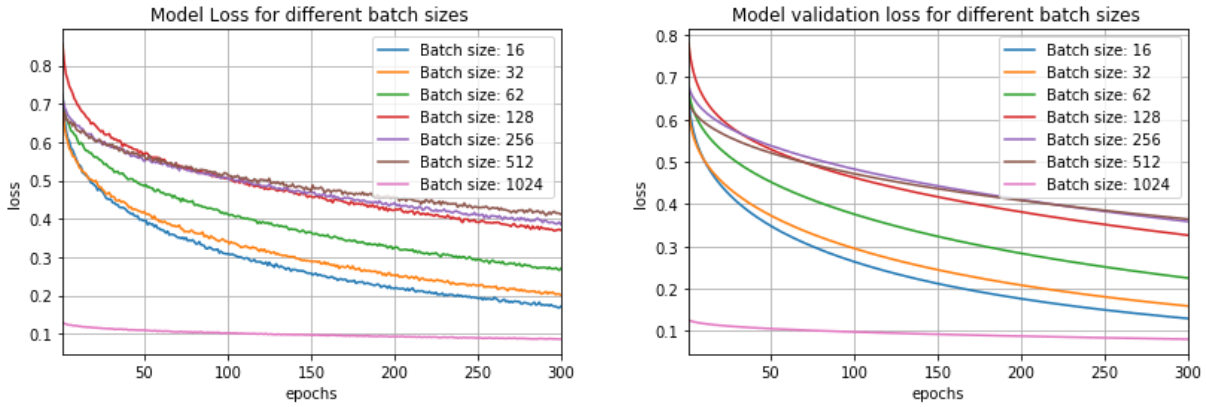
$$\phi(x) = max(0, x) \tag{3.9}$$

The dropout function was introduced by Srivastava et al to decrease potential overfitting by dropping randomly a number of training examples in each iteration [92]. For example, in our experiments we use a drop out of 20%, that means that 20% random input signals are ignored in each epoch and for every hidden layer. In this way the algorithm learns to generalize better in un-seen cases and improves the overall performance of the system [92].

The learning parameters that need to be fine tuned are the batch size on which the algorithm is trained on in each epoch, the learning rate that the algorithm learns, the optimization function and the loss function which is used to calculate the error rate. In the next paragraphs we are going to discuss each parameter separately looking at the loss and the validation loss of the model as comparison criterion.

Figure 3.8 shows the effect of 7 different batch sizes applied in the model during the experiments. We can easily notice that almost for all the batch sizes applied in the model the training and the validation loss converge and it learns the information provided by

the features of the training set but it also performs well on the validation set. However, as Hoffer et al discuss in their study in [93], higher batch sizes reduce the generalization capability of the model and at the same time it increases its computational performance. For these reasons, we choose as the optimal batch size for our model 32 trainable examples for each epoch. Another interesting study concerning batch size and droup out rate conducted by Ioffe et al in [94]. The study concluded that the number of examples during each iteration is reverse proportional with the percentage of example dropped out during the learning phase . This means that if we employ a low number of training instances in the learning process, we should also use very low dropout rate, or even do not use any at all.




(a) Loss

(b) Validation loss

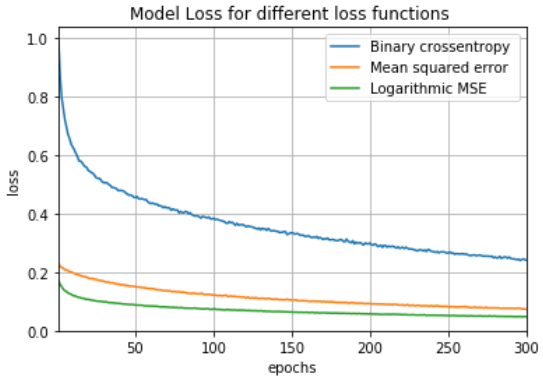
Figure 3.8: Loss and validation loss for different batch sizes

In our experiments to fine tune the loss function, we employed three loss functions, namely, binary crossentropy, mean squared error(MSE) and logarithmic mean squared error(LMSE). Due to the nature of our problem, not surprisingly we see in figure 3.16 that the optimal loss function is the binary cross entropy. We said not surprisingly because loss crossentropy is suggested as the most appropriate function for a two class classification task [95]. The calculation of binary cross entropy is perfomed by the formula 3.10.

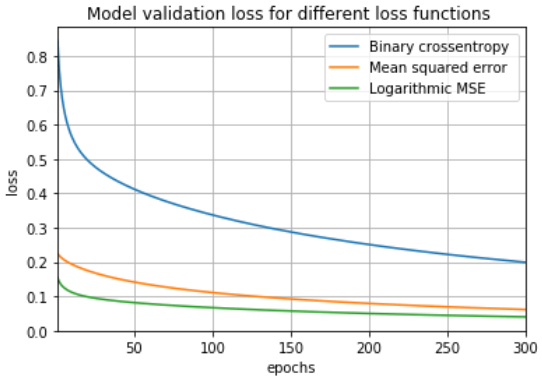
$$c = p_i \; log(\frac{1}{q_i}) \; + \; (1 - p_i) log(\frac{1}{(1 - q_i)}) \tag{3.10}$$

$q_i$ is the estimated probability of an example to belong on one of the classes, while $p_i$ is the actual probability. As the plots show, MSE and LMSE show an almost constant loss rate in the training and testing sets which indicate that the algorithm does not actually learn properly.

Next, we experiment with different learning rates. As we mentioned before, the learning rate indicates how fast the algorithm learns from the data in such a way that minimizes the error rate. Finding the optimal learning rate that fits the data the best is usually tricky. First we experiment with some constant values and later will use the time based decay



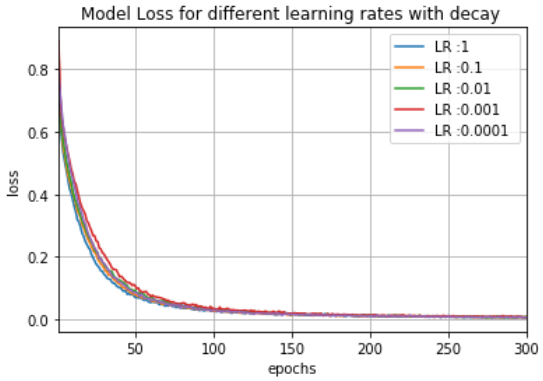

(a) Loss



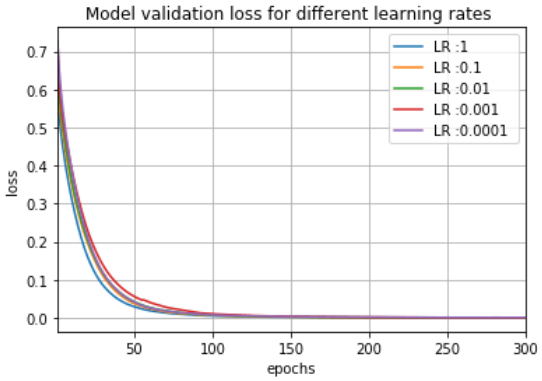

(b) Validation loss

Figure 3.9: Loss and validation loss for different loss functions

argument to decrease the learning rate in each epoch during the training process. Looking at the graphs in figure 3.10 we can see that the model performs equally for all the learning rates. According to the study conducted by Zhang et al in [96], sophisticated optimizers such as AdaGrad, RMSProp and Adam fine tune the learning rate individually for each feature that is learned in the model. Since we are using AdaGrad to optimize our model, we assume that is the reason why the performance of the model is constantly for all the learning rates applied.




(a) Loss

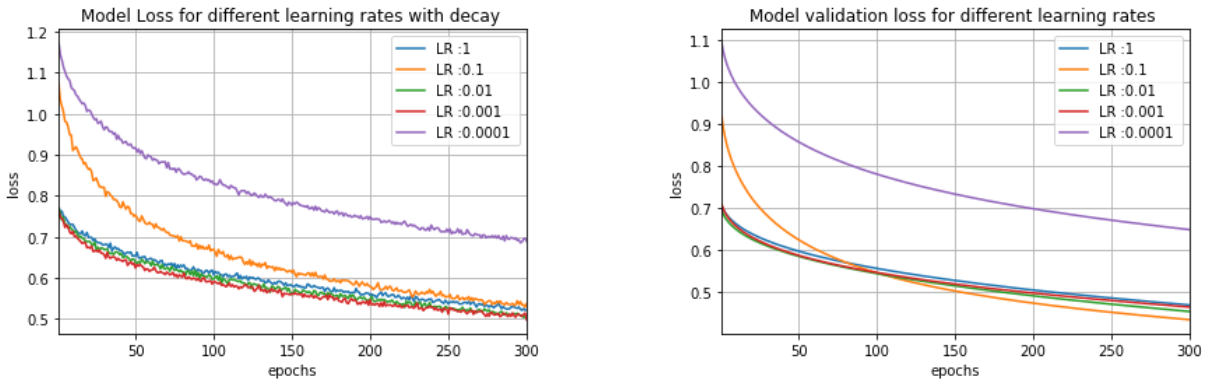



(b) Validation loss

Figure 3.10: Loss and validation loss for different constant learning rates.

Despite the controversial view that some scientists have concerning the use of decay in learning rate [97][98][94], we implemented a number of experiments to evaluate the performance of our system when the learning rate decreases. We used the learning rates employed in the previous experiments, so that we can compare the effect of the decay parameter. Decay in learning rate decreases the learning rate in function with the time. In some studies it is proposed to use a constant value independently of the overall set up of

the model while others suggest to set it as the quotient of the learning rate divided by the number of epochs used to train the model. Since the learning rate is divided continuously with a constant number at some point is tends to zero [98]. This very small number then, is very practical for the system to learn details of the data. Figure 3.11 depicts the losses for the five experiments. We see that the two of the five learning rates show an improvement in their performance when decay is involved in the learning process. However, the other three learning rates seem to adopt the information of the data better and show a bit lower loss during the process. From the results show in the two graphs in figure we concluded that the best learning rate for the model is that with value 0.001.

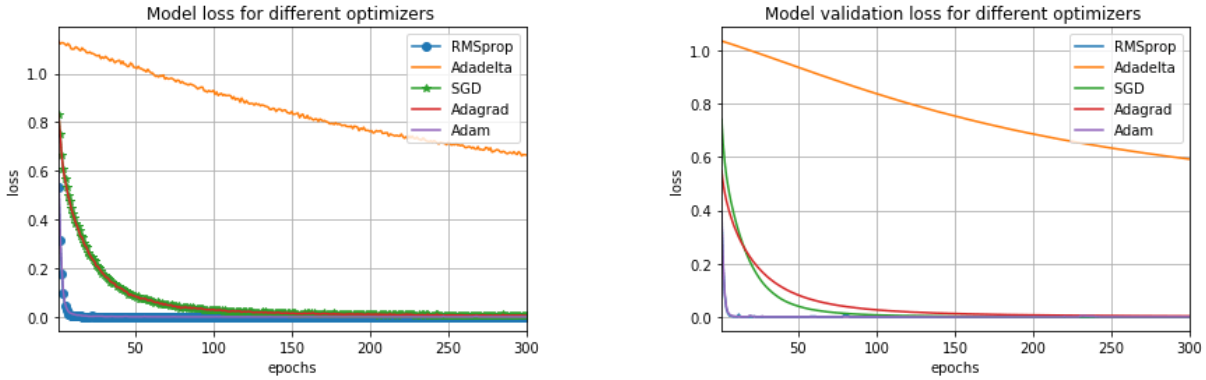



(a) Loss

(b) Validation loss

Figure 3.11: Loss and validation loss for different learning rates with constant decay.

The optimization algorithm used in the model is another parameter that needs to to be improved. The function of this algorithm is to appropriate update the internal parameters of the model after each iteration in function with the error minimization during the learning process. Figure 3.12 plots the losses for a number of optimizers. AdaGrad and SGD show a smoother learning loss and also converge better with the validation loss. Based on our experiments we concluded that AdaGrad fits better our data and leads to a better predictive results.




(a) Loss

(b) Validation loss

Figure 3.12: Loss and validation loss for different optimizer functions

## 3.5 Pre-operational model

The pre-operational predictive model for graft loss can be seen as of higher importance. This is because the learning model is trained only in the available information before the operation and according to this knowledge it predicts whether the recipient is going to experience graft loss or not. The data set is consisted of 48 features. Before starting the parameter optimization as we did previously, we will apply the PCA algorithm to reduce the dimensionality of our data-set and facilitate the learning process.

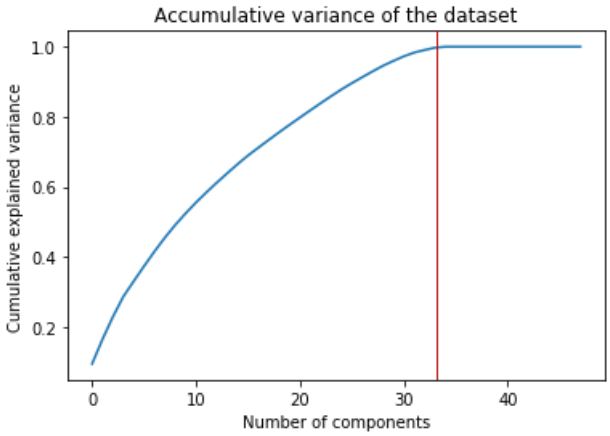
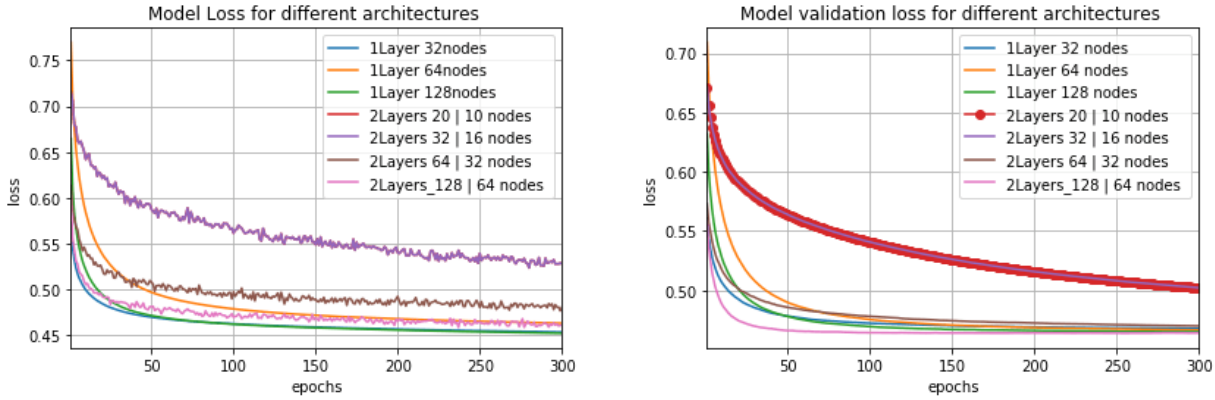



Figure 3.13: Dimensionality reduction with PCA.

As the plot in figure 3.13 indicates, our data-set with the pre-operative data for the patients involved can be reduced into 32 principal components that would carry all the knowledge of our data.

Next, we will perform parameter optimization to identify the framework that fits our data the best. We will start with the architecture of the network. We performed a number

of experiments and in figure 3.14 we present some results. As we mentioned earlier, the data set of the pre-operative characteristics of the donors and the recipients is much smaller from feature size perspective, which implies that we need a different architecture than the one we used before for the post-operative model. In contrast with the other model, we see that models consisted with only one hidden layer, perform pretty well. According to these results we concluded to utilize the model witch is consisted of one hidden layer composed of 32 nodes.




(a) Model loss for varying architectures

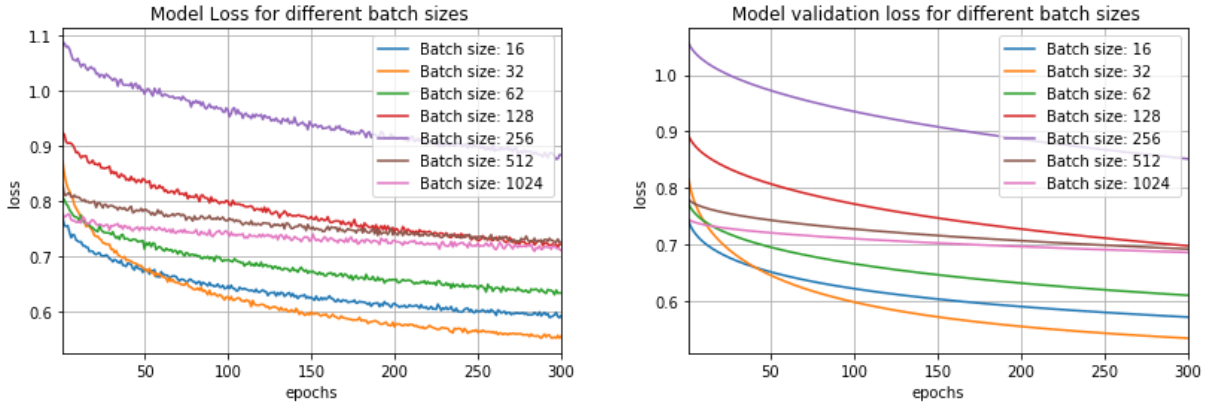
(b) Model validation loss for varying architectures

Figure 3.14: Loss for different architectures of the network

In figure 3.15 we can see the effect of using different batch sizes in the pre-operational predictive model. From the plots of the graph we can conclude that smaller batch sizes learn the data smother than larger ones. The number of training examples that fits the data better and outperforms the other set-ups according to these experiments is 32 training samples per epoch, with 16 and 62 coming second and third respectively. We chose to train our model with 32 training samples because it seems that its learning ability increases incrementally from the beginning of the learning process, in contrast with the model trained with 16 and 62 nodes which although perform a bit worse, their leaning ability is shown to be less sharp in the first 50 epochs.

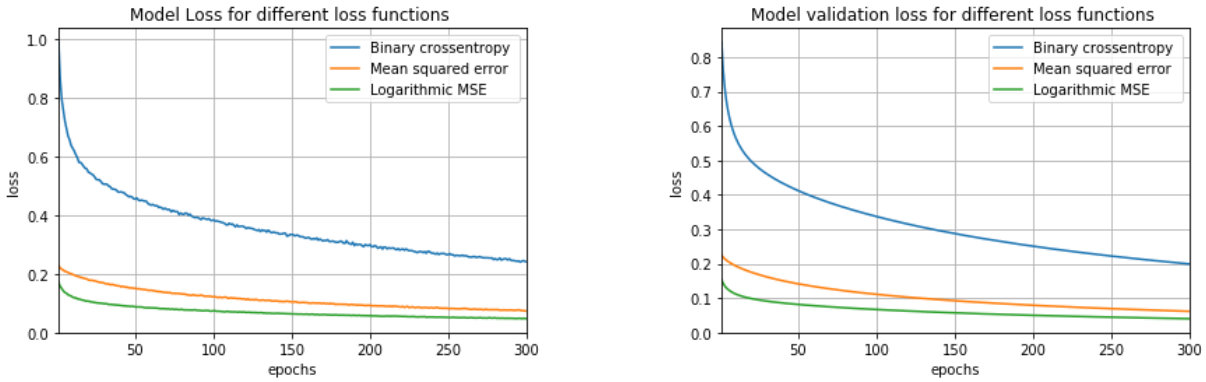
In addition, we made experiments to find the best loss function that suits our research question. As we mentioned before, since our problem is a binary classification problem, the loss function that should perform the best is binary cross-entropy. The results of the pre-operative model approve again this theory as it is shown in figure 3.16. We can see that binary cross entropy fits the data better during the learning process while the other two methods seem to overfit.

As we already mentioned earlier, the learning rate defines the incremental steps that are taken from the model to adopt the information involved in the data. From the experiments conducted we can see that when the learning rate is kept stable along the learning process, does not really differ in its predictive efficiency. For the five different learning rates executed in the model, we can see in figure 3.17 that its predictive abilities remain stable.




(a) Loss (b) Validation loss

Figure 3.15: Loss and validation loss for different batch sizes.

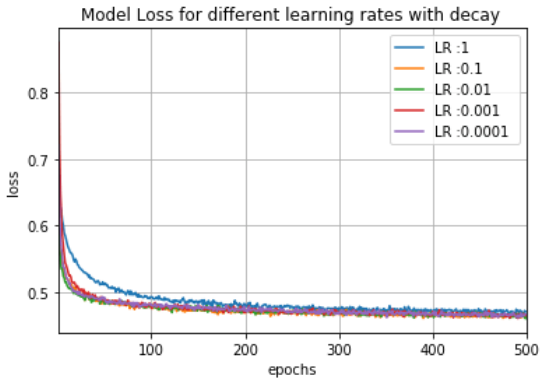



(a) Loss (b) Validation loss

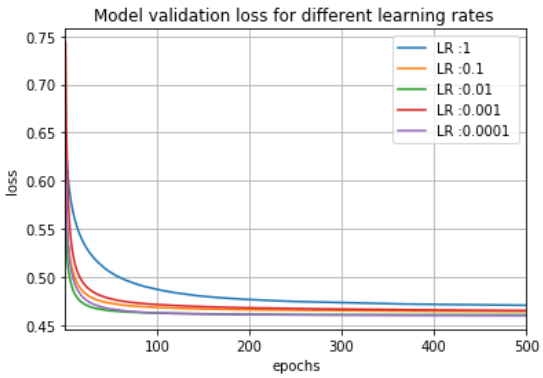
Figure 3.16: Loss and validation loss for different loss functions.

However, when the learning rate is decreased incrementally after each epoch executed, we see that the learning ability of the system show a significant improvement. From the plot in figure 3.18 we see that very large and very small learning rates perform a bit worse the the other learning rates. However, it is important to mention here that using high learning rates increase the risk of getting stuck in local minima which would have a negative effect in models performance. We conclude that the best learning rate in combination with the other parameters of our model is 0.001 with a decay rate of $10^{-6}$.

Before we conclude for the best set up for our model, we need to find which optimizer performs the best for our data and our model. The learning capabilities of the five optimizers tested in this model are shown in figure 4.4. Adadelta shows the worse performance among the five optimizers evaluated, displaying a very slow adaptation pace, while RMSprop, Adam, SGD and Adagrad fit the data better. SGD and Adagrad demonstrate an equal performance regarding the training loss, however, Adagrad outperforms in the

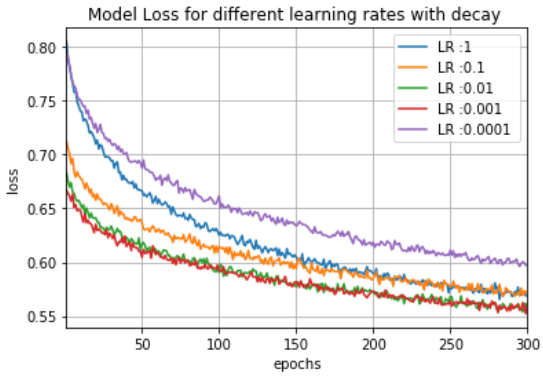



(a) Loss

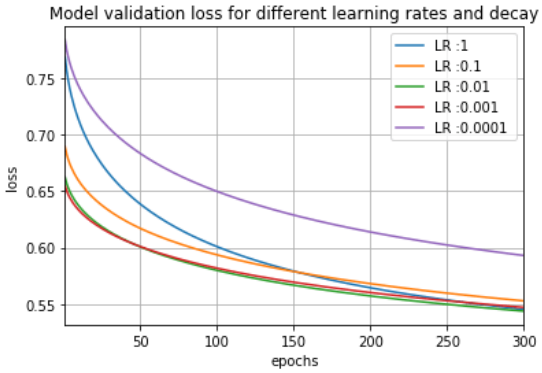



(b) Validation loss

Figure 3.17: Loss and validation loss for different constant learning rates.



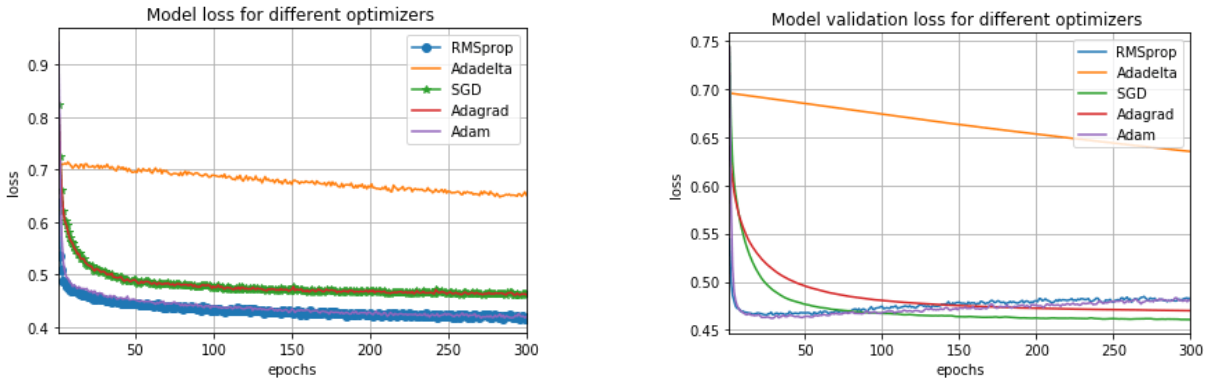

(a) Loss




(b) Validation loss

Figure 3.18: Loss and validation loss for different learning rates with constant decay.

validation loss and appears to be the best for our model.




(a) Loss

(b) Validation loss

Figure 3.19: Loss and validation loss for different optimizer functions

## 3.6 Results and Validation

### 3.6.1 The proposed post-operational model

After optimizing all the parameters of the model we conclude in the proposed model. The model is feed with the 112 features of the initial data-set , which are then reduced to 80 by applying the dimensionality reduction algorithm PCA. Figure 3.20 depicts a graphical representation of the model, where the input layer shows the variables that asre fed in the network before dimensionality reduction. Next, we can see the two hidden layers, the first one consisted of 128 nodes and the second one of 64 neurons and finally the output layer which expresses the predictive result, graft loss, or no graft loss.

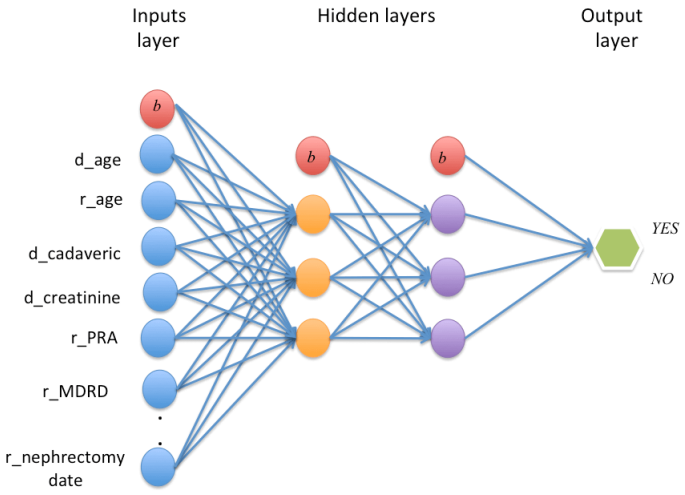



Figure 3.20: Model architecture.

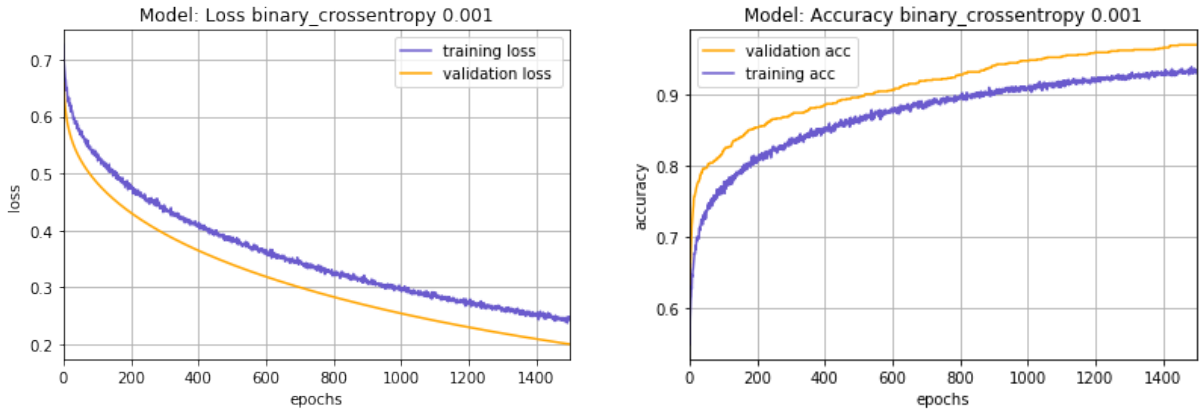
The selected loss function for our model is Binary cross entropy and the batch size that it learns is 32 training example in each epoch. The optimizer that outperformed in our examples is the Adagrad and the learning rate adopter is 0.001. During the training process of the model 18.689 parameters are optimized, so that the model can yield the best predictive outcome. Figure 3.21 shows the architecture of the model and also the number of parameters optimized in each layer, as well the total number of them.

| Layer (type) | Output Shape | Param # |
|---|---|---|
| dense_257 (Dense) | (None, 128) | 10368 |
| dropout_161 (Dropout) | (None, 128) | 0 |
| dense_258 (Dense) | (None, 64) | 8256 |
| dropout_162 (Dropout) | (None, 64) | 0 |
| dense_259 (Dense) | (None, 1) | 65 |

Total params: 18,689
Trainable params: 18,689
Non-trainable params: 0

Figure 3.21: Model architecture.

Based of the information provided on the model, it achieves an accuracy of 96.8% to correctly predict whether a recipient is going to experience a graft loss or not. The loss curve and the accuracy plot from both the learning and the validation process are shown in figure 3.22.




(a) Model loss

(b) Model accuracy

Figure 3.22: The accuracy and the loss of the proposed model

### *k* - fold cross validation

The predictive accuracy of a classifier in most of the cases is evaluated according to its error rate and its accuracy. However, classifier assessment is also required to prove what the classifier is actually learning and that it is able to generalize in new data [99] [100]. During the training process predictors learn the features of the data but they also adopt the noise involved on them which could lead either to over-fit the model or under-fit it [101]. Cross validation is a well known statistical strategy to estimate the overall learning skills of the model. The basic idea behind this algorithm is to get trained in individual subsets of the data that do not overlap and calculate the mean performance of the model [100]. To do so, the k fold cross validation algorithm divides the initial data-set into k non-overlapping sets. In each of the k iterations that are performed, the algorithm picks one of the k folds randomly as the validation set and the other k-1 sets are used to train the system. After k iterations, all the folds are used as a validation set. A graphical representesion of how the algorithm is functioning is shown in figure 3.23. The overall accuracy is then calculated by taking the mean of the k individual results. The reason why this technique is used to validate a predictor underlies on the fact that all the examples in the data-set have been used both in the testing process and in the training process. This implies significant reduction in bias and variance [100].

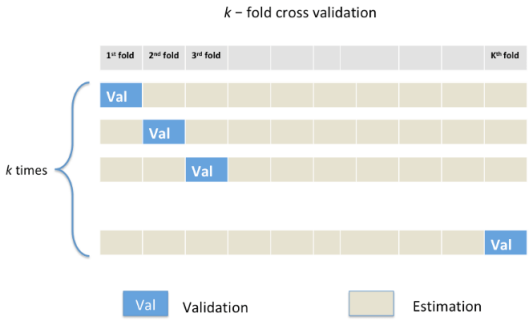
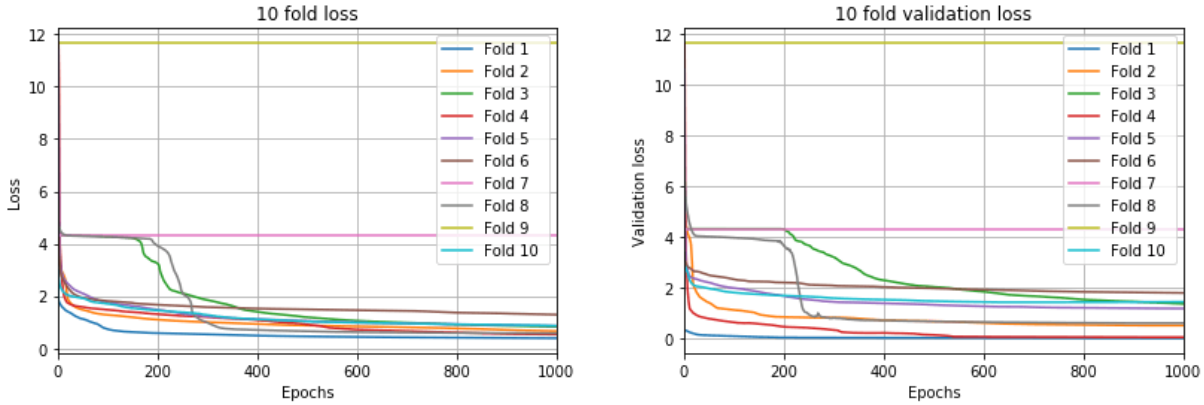



Figure 3.23: Graphical representation of the $k$-fold cross validation

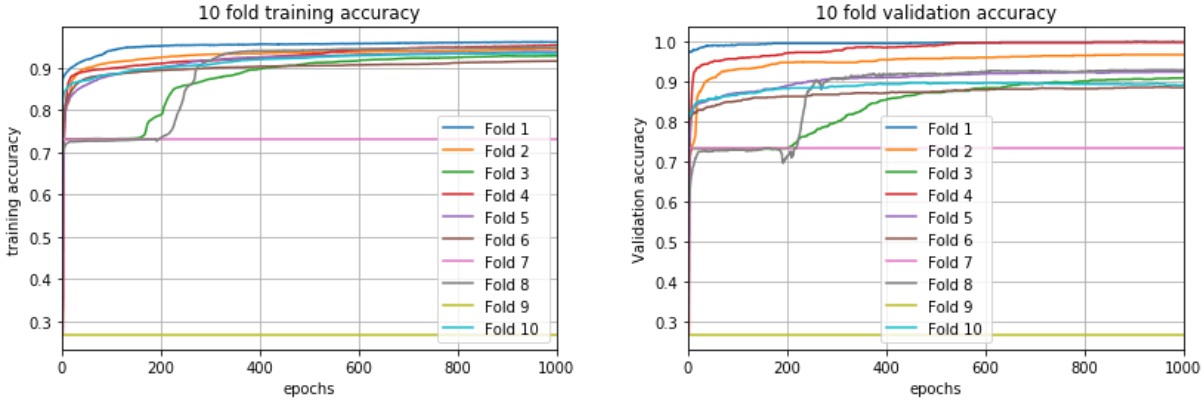
Looking at the results of our model shown in figures 3.24 and 3.25, we see the error rate and the accuracy of each fold both in the training and the validation phase of each fold performed. The fluctuation in the curves is a good indicator of the variance in the data-set, which 10 fold cross validation is able to detect. In both figures that the loss and the accuracy in the $7^{th}$ and $9^{th}$ fold remains stable from the beginning of the training process, this means that probably the algorithm got stuck at some local minima and it could proceed in the learning phase. The same happened at the $3^{rd}$ and $6^{th}$ fold at the begging of the process, but as we can see, the algorithm continued learning approximately after 200 epochs.

Figure 3.26 depicts the averaged performance of the 10 folds. The validated model achieved an accuracy of 89%. We can notice that the curves are fluctuating in comparison




(a) Loss (b) Validation loss

Figure 3.24: 10-fold cross validation loss.



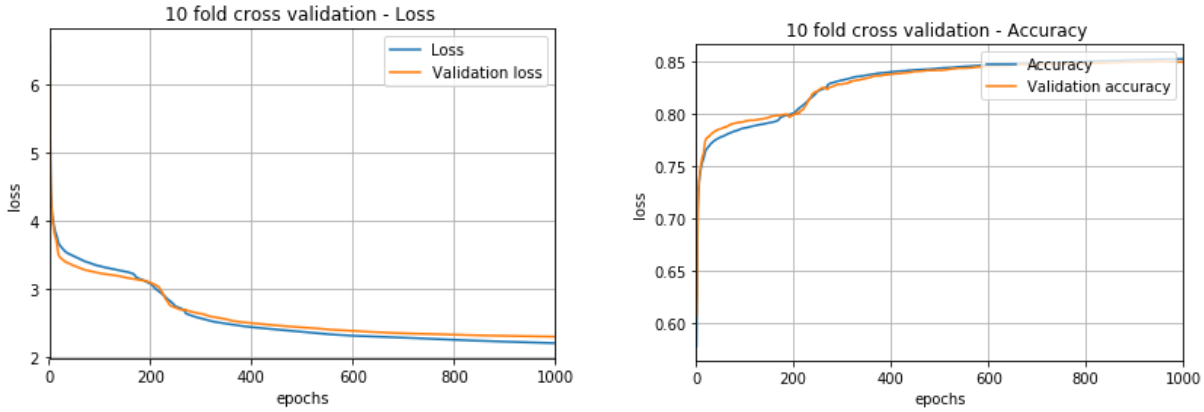

(a) Loss (b) Validation loss

Figure 3.25: 10-fold cross validation accuracy.

with the corresponding curves from the model itself. This can be explain due to the averaged result of the 10 folds.

### 3.6.2 The proposed pre-operational model

Having experimented with all the parameters and the architecture of the pre-operational model we can conclude in the model with the best performance for the task. The model is trained in 48 features which are reduced to 34 after applying PCA. The input layers is constisted of 34 nodes, one for each feature, plus one for the bias. Then, the hidden layer has 64 neurons and one for the bias and finally the output layer with one neuron expressing the predictive result. Figure 3.27 depicts a graphical representation of the models architecture.




(a) Traning / testing loss

(b) Traning / testing accuracy

Figure 3.26: 10-fold cross validation loss and accuracy.

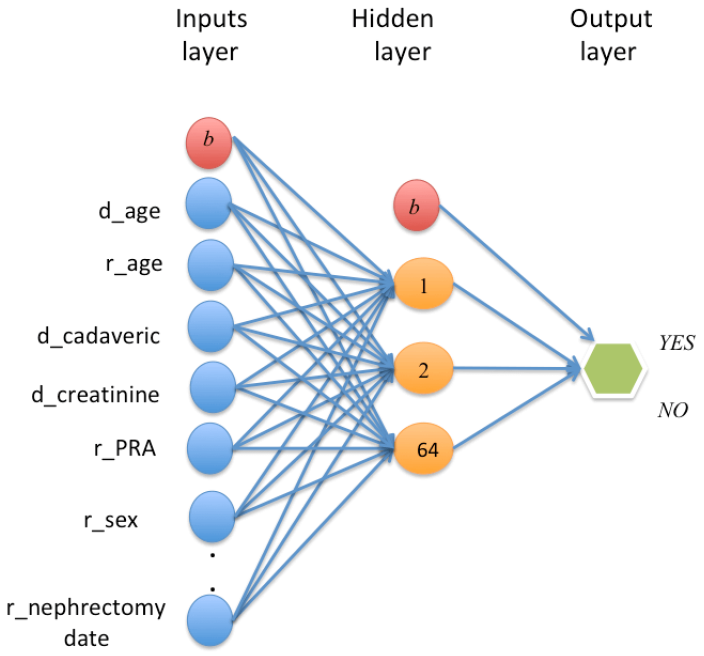



Figure 3.27: Pre-operational model architecture.

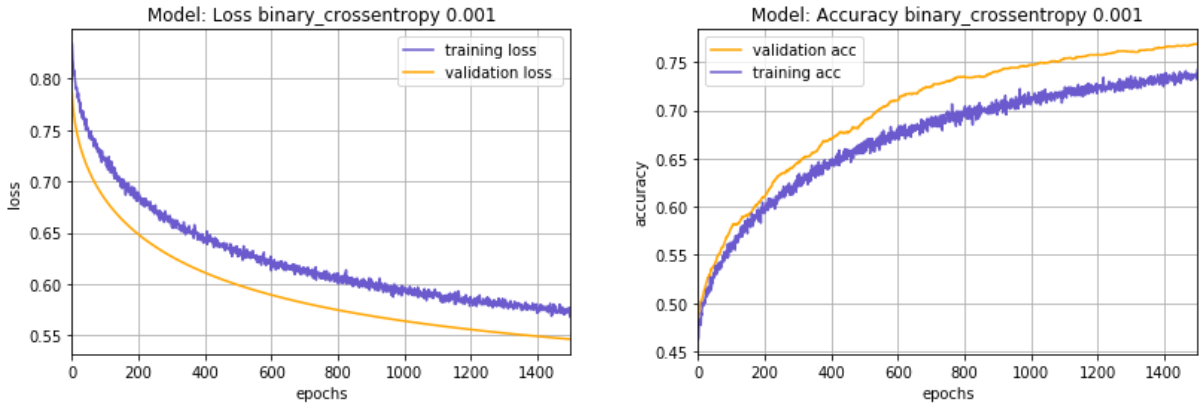
During the learning process, the model was estimating 2.240 trainable parameters in the hidden layers and 65 weights in the output layer in order to make the prediction, summing to 2.305 parameters in total. Figure 3.28

The proposed model achieves an accuracy of 76.8% in predicting whether the recipients is going to experience graft loss after the transplantation, given the characteristics of the donor and the recipient involved. Figures 3.29a and 3.29b depict the loss curves and the accuracy curves during the learning phase.

| Layer (type) | Output Shape | Param # |
|---|---|---|
| dense_159 (Dense) | (None, 64) | 2240 |
| dropout_78 (Dropout) | (None, 64) | 0 |
| dense_160 (Dense) | (None, 1) | 65 |

Total params: 2,305
Trainable params: 2,305
Non-trainable params: 0

Figure 3.28: Estimated parameters of the model.




(a) Model loss

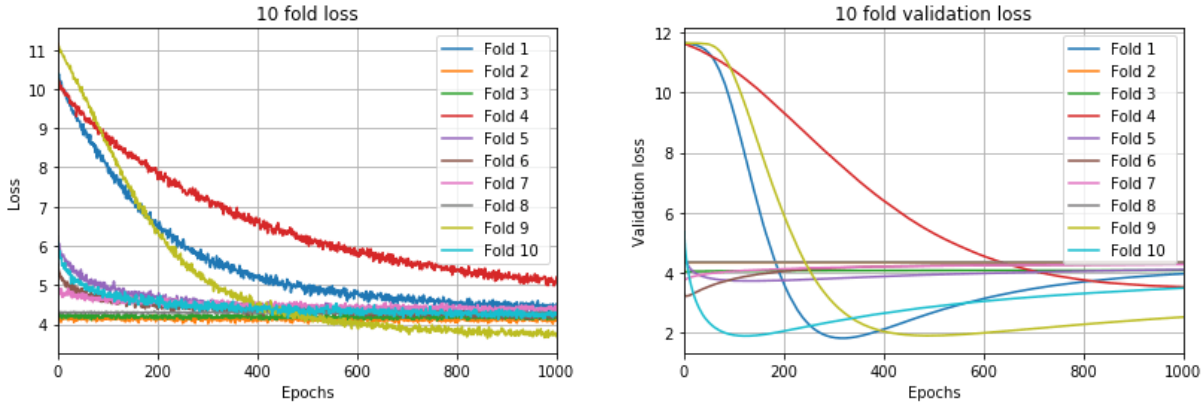
(b) Model accuracy

Figure 3.29: The accuracy and the loss of the proposed model based on the pre-operational data

### *10* - fold cross validation

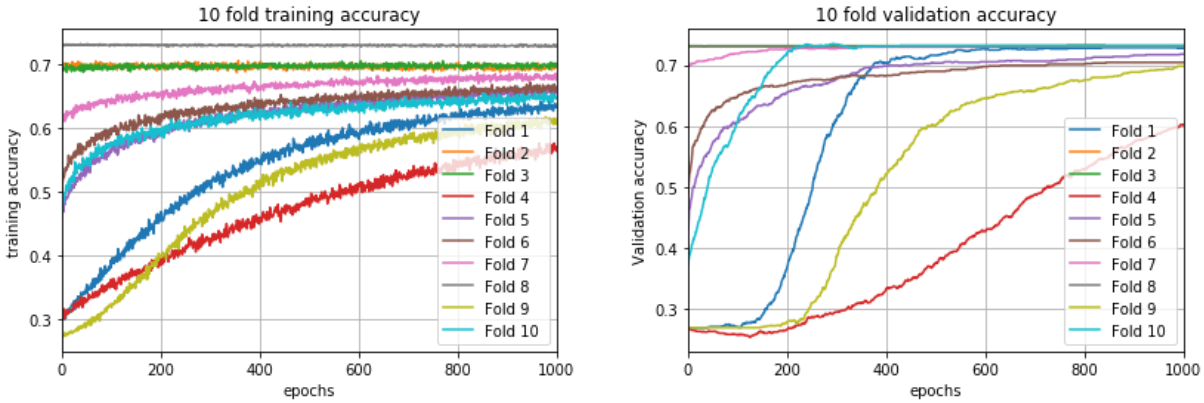
To verify the performance of the model we performed 10-fold cross validation on the proposed model. As we discussed earlier, the data-set was split into 10 folds and the it performed 10 iterations where each time one of the folds was employed as the testing set and the rest nine as the training set. In figure 3.30 we can see the loss and the validation loss for each of the 10 folds separately. It is noticeable that each fold performs differently due the fact that it is trained and validated in different data. Fold 1, 4 and 9 fit the data better, while the rest seven folds perform constantly. In figure 3.30b is important to mention that fold 1 and fold 10 seem to over-fit the data, since their validation loss starts increasing after a number of epochs.

In figure 3.31 we see the training accuracy and the validation accuracy. Again we can notice that the variance in the performance among the 10 folds. Folds 2 and 3 do not show any improvements in their predictive accuracy throughout the learning process, which may be caused by some local minima they got stuck and they did not manage to adopt any further information from the data.




(a) Loss (b) Validation loss

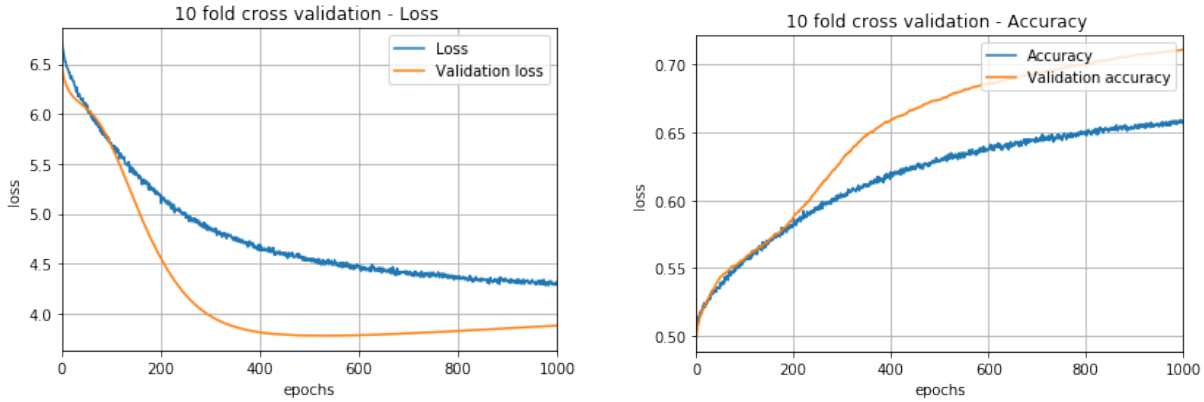
Figure 3.30: 10 fold cross validation loss in the pre-operational model




(a) Loss (b) Validation loss

Figure 3.31: 10 fold cross validation accuracy in pre-operational model

The results of the 10 folds in cross validation are then averaged and shown in figure 3.32. The cross validated model achieves an accuracy of 73.2% to predict the correct outcome.




(a) Training / testing loss

(b) Traning / testing accuracy

Figure 3.32: 10 fold cross validation loss and accuracy in pre-operational model

## 3.7 Summary

In this chapter we introduced the ANNs and we developed two predictive models to forecast graft survival for kidney transplantations. First we performed parameter and architecture optimization experiments in order to develop the most accurate models that fit our data. The pre-operative model, is a MLP composed of tho hidden layers, with 128 neurons in the first layer and 64 in the second while the optimal architecture for the pre-operative model is composed of only one hidden layer with 64 nodes. During our experiments to optimize the parameters of the model, we found out that the parameters that learn the algorithms the best, are the same in both models. This could be expected since the data originate from the same source and the only difference between them is the number of the features used. The performance of the post - operative model is 96.8% in predicting correctly graft survival, while the prognostic model show an accuracy of 76.8%. We also performed cross validation in the two models, where the performance was a bit lower. The post - operative model shown an accuracy of correctly predicting graft loss of 89% while the pre-operative model 73.2%.

# Chapter 4

# Semantic modeling

The tremendous amount of digital information produced nowadays by high throughput technologies in all disciplines, results in the need to develop metadata strategies that can effectively describe the information captured in the data [102][103]. In health-care, this problem is particularly apparent. The information captured in EHRs originates from diverse digital sources, collected from different professionals and divergent medical devices [104][103][105]. As a consequence, health care data usability is restricted from the perspective of information exchange between incompatible systems, reuse, integration and information accuracy. Having said that, superior technologies are required to integrate heterogeneous information dynamically, with flexibility and the ability to deal with the complexity involved in each domain [103]. For these reasons ontologies have been introduced to provide conceptual knowledge that can be shared between people or machine processing agents [106]. The significance of semantic modeling in supporting kidney allocation is discussed in the following sections.

## 4.1 Ontologies

An ontology is a knowledge representation model that describes a set of concepts in a domain of interest and expresses the relations among these concepts by using properties and restrictions [107][108]. The broad usage of ontologies underlies the explicit vocabulary and the structured information captured that allows humans and computers to share knowledge [107]. For this reason, domain knowledge captured in an ontology can be reused and integrated with other ontologies. This fact allows researchers to use semantics of different domains to analize and answer more complex tasks [107]. In addition, the easiness of ontologies to be modified and reasoning without expertise knowledge in contrast with other computational techniques, has attracted the scientific interest. Today, ontologies have been applied in diverse fields, from the web to nutrition and from agriculture to medicine. In this study we focus only on the health care sector and we propose an ontology for the recipient - donors pairs for kidney transplantations that adopts the particularities involved in the EHR of each pair and it could automatically provide the necessary information for

clinicians to operate with them. Then, this ontology can consist the ground of developing a personalized decision support system by integrating it with the results of the predictive models discussed in chapter 3.

### Ontology components

Web Ontology Language (OWL)[1] is a Semantic Web language used to represent wealthy concepts and the relations between these concepts in a such a way that can be computationally exploited. The building blocks that compose an OWL ontology are its individuals, properties and classes [109]. In the domain of interest, *individuals* are the representations of the objects that are involved in the domain while classes are concepts that individuals can be classified in according to some precise requirements. For instance, a recipient is a person who was implanted his/her right or left kidney. The properties of an OWL ontology express the relations between two concepts. For example, patient A *has donor* patient B. However, properties can be expressed reverse. Patient B *is the recipient of* patient A. Properties are distinguished into two categories: object properties, which describe relations between two individuals, and the data properties which express the relations between individuals and data values [109]. Classes in the domain of interest are sets of individuals that fulfill the explicit circular argument of a specific class.

Classes can express the domain of knowledge hierarchically, by employing super-classes and sub-classes. For instance, recipients and donors are physically people. Semantically recipients and donors (sub-classes) specialize the domain of people (Super-class). This taxonomic organization implies that all donors and all recipients are people, and the reverse, people can be either recipients or donors. Disjoint classes are an important aspect of semantic modeling in restricting individual to be part of specific class. In the transplantation ontology for instance, a person who is donor, can not be a recipients at this time. To model this relation, we make the two classes disjoint, so each individual should only belong to one of these two classes.

## 4.2 Previous work

Many studies have been conducted regarding the semantic modeling of medical data. Riano et al in [108] developed an ontology in the domain of chronically ill patients which represents the knowledge of the health care at home of these patients for 19 diseases 2 syndromes and 5 social issues. Based on this ontology the authors constructed two different models that provide clinicians meaningful knowledge for the profile of these patients. The first model implementation yields personalized clinical information for the diseases that patients suffer from and are necessary for clinicians to know about. While the second model integrates the personalized knowledge from the ontology with the interventions suggested for each disease and automatically suggests the potential treatment required for every case. Additionally, the authors built a personalized decision support system

[1] https://www.w3.org/OWL/

capable of identifying abnormalities in the patients record. Such peculiarities could be wrong diagnoses, comorbidities that were not diagnosed, missing information ect. Based on the clinical standards the proposed decision support system achieves 84% accuracy in the correct reasoning and it was evaluated for its applicability with 90%.

Another study that validates the power of semantic modeling to provide sufficient knowledge to support medical decision is discussed in [106]. Eccher et al introduced an ontology that covers the knowledge on 357 breast cancer therapies and models semantically the ambiguous involved between different EHRs. The knowledge captured in the OWL ontology was then applied to build classification rules that can distinguish the features of each individual therapies using the Semantic Web Rule Language (SWRL) rules. The efficiency of the model was evaluated according tho the domain knowledge provided by oncologists and it show a high accuracy in predicting the suggesting therapy with an error rate of 0.019% (7/357) while 26 of the therapies were not classified at all (5.4%).

An interesting study that may help people suffering from liver, breast and lung cancer world wide was conducted by Alfonse et al in [110]. In this study the authors developed a personalized decision support system that is capable of diagnosing cancer, identifying cancer stage and finally recommend a personalize treatment for the patient. For the three tasks mentioned above, the authors constructed three different modules, one for each task and all of them were interconnected to an ontology database. The ontology database was built by integrating the domain knowledge of three other ontologies, namely the lung cancer ontology, the liver cancer ontology and the breast cancer ontology. The three modules reasoned the database and suggested a personalized medical decision according tho their symptoms with an accuracy of 92%.

The significance of using ontologies and Semantic web instead of relational DataBases was studies by Martinez et al in [111]. In this study, the authors highlighted the importance of the open world assumption used in ontologies to integrate Semantic data from different sources online, which is an aspect that relational databases lack. Another basic difference between the two systems is that due to the schema that ontologies have, hidden knowledge for the domain of interest can easily extracted. However, concluding their study, they claim that the choice between the two system depends on the needs that someone have.

## 4.3 Transplantation Ontology

In this study we developed an ontology that can provide a formal personalized representation of all the concepts involved in a kidney transplantation. To construct the ontology we used the Web Ontology Language (OWL) and the ontology editor Protégé[2]. Patients characteristics, clinical assessment, laboratory tests and medical outcomes for the donors and the recipients involved in a transplantation are the concepts that our semantic model captures but also the relationships and the restrictions associated with these concepts. The proposed ontology was built based on the available data provided by the Netherlands Organ

[2]https://protege.stanford.edu/

Transplant Registry (NOTR) aiming to clarify the concepts involved in the transplantation cases in hand.

The main concepts involved in the transplantation ontology can be distinguished in three hierarchical levels. Figure 4.1 shows a graphical representation of the hierarchy of these concepts. Circles represent the concepts of the domain of interests, arrows show the relation between these concepts. Starting from the OWL thing, which is the domain of transplantations, we can identify the super-classes of the ontology(dark green circles). These are the characteristics of the patient, the screening of the organ before and after the operation, information for the recipients death, the information for donors death, the people involved in the transplantation, potential diseases that recipients and donors may suffered from and the information about the transplantation procedure. Next, the second and third hierarchical levels shown in the graph with lighter and lightest shade of green circles respectively, represent more specific concepts than the class they originated from. For example the super-class Disease has two sub-classes that describe its conceptual knowledge, the subclass Hypertension and the Diabetes.

In addition, we can define restrictions that should be satisfied from the individuals in order to be classified on them. For example, a donor is defined as the person who has some recipient, but in order to be a donor should be defined the type of cadaveric. Figure 4.2 shows how this restriction was made in Protégé.

Similarly, the NHB categories, which stands for Non Heart Beating donors and describes the type of cadaveric each patient had, have to be restricted only to patients who were cardiac dead, since brain dead donors do not have this aspect. In addition the subclass of the class NHB category were made disjoint since a patient can be classified only in one of the four sub-classes of this concept, as figure 4.3b shows.

As we mentioned earlier, object properties describe the relation between two individuals. For example an individual who belongs to the class donor can have a property *isdonorof* to describe the relation this individual have with some other individual who belong to the class recipient. This relation can be expresses also with its inverse property *hasrecipient*.

The properties that explain the relation between two classes sometimes may have some sub-properties as it is shown in figure 4.4a, to express more specific properties between the two classes. For instance the object property *hasOrganReaction* has five sub-properties, namely *hasCauseofgraftFailure*, *hasDelayedGraftFunction*,*hasEarlyGraftLOss*, *hasGraftLoss* and *hasGraftFailure* which express the potential reaction of the graft after the operation.

In our ontology the data properties describe the relations between individuals and the concepts that characterize them in the data. For example the property *hasage*, describes the relation of each individual in the data-set with the variable age. Figure 4.4b depicts the list with the data properties and the sub-properties employed in this ontology.

To summarize, in this chapter we introduced a personalized semantic model that represents the conceptual knowledge from the data-set of transplantations we worked with in this project. The ontology is built based on sever super-classes, 21 middle level concepts and 46 second level concepts. To properly represent the relations between the concepts of the domain of kidney transtrantation we used 13 object properties and 30 data proper-

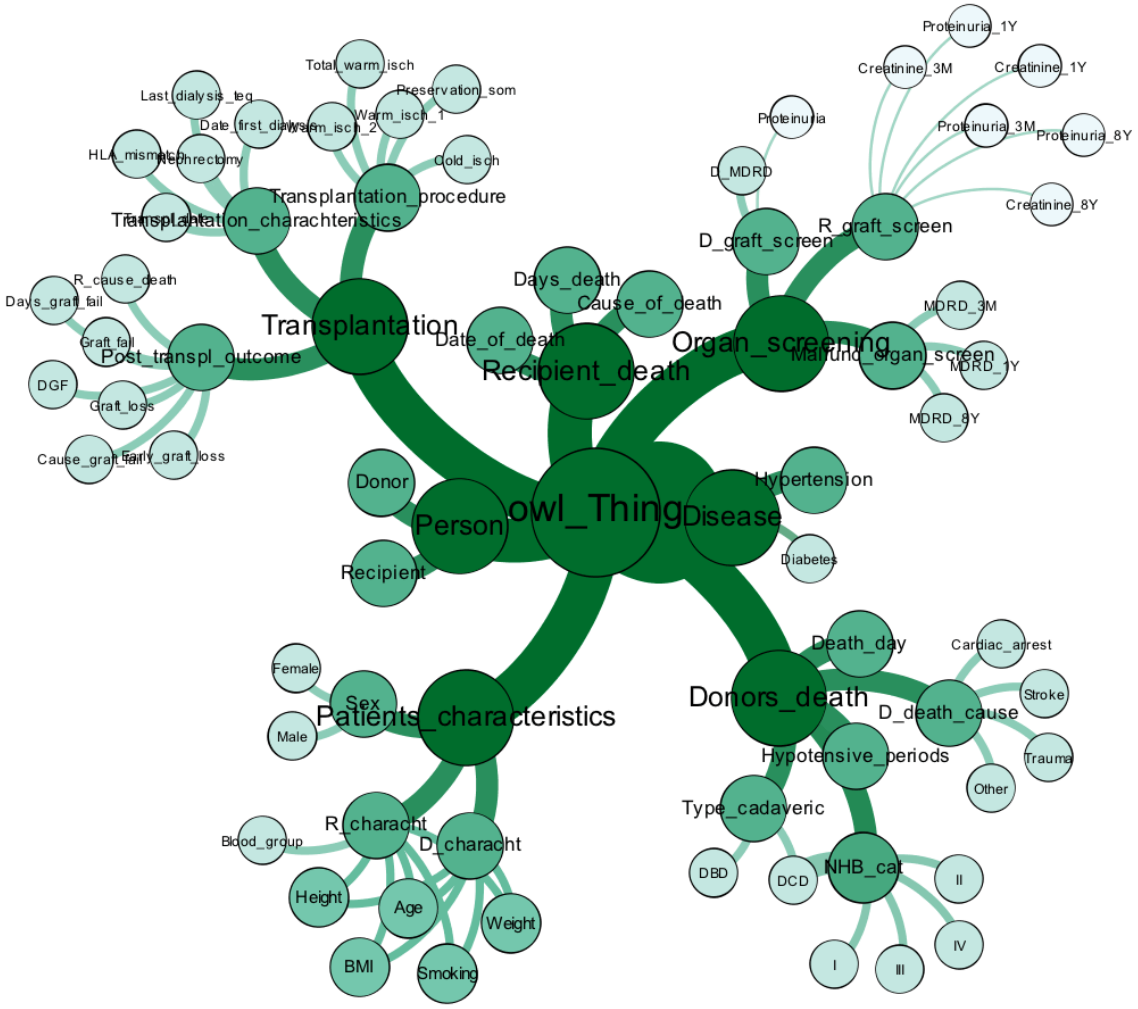



Figure 4.1: Visualization of the Owl ontology.

ties. The constructed ontology together with the predicted outcomes from our models can provide the ground knowledge that is required to construct a personalized decision support system. The development of such a system could help clinicians to make more accurate decision in graft allocation. In addition clinicians can reason the ontology and identify the effect that each parameter has in graft survival.

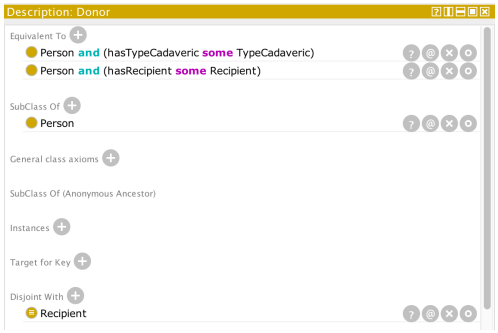
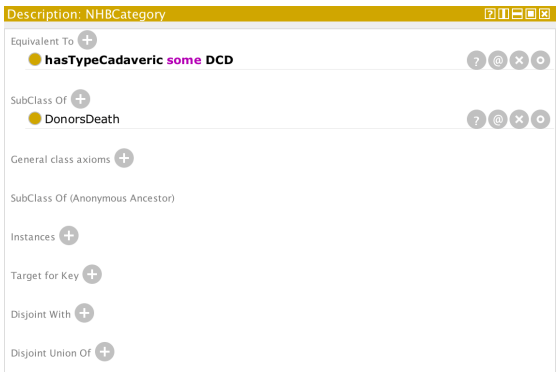
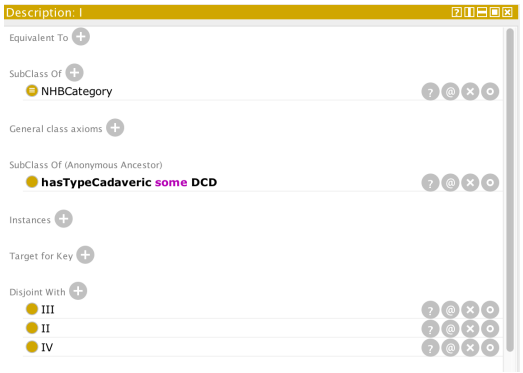



Figure 4.2: Donor class description.

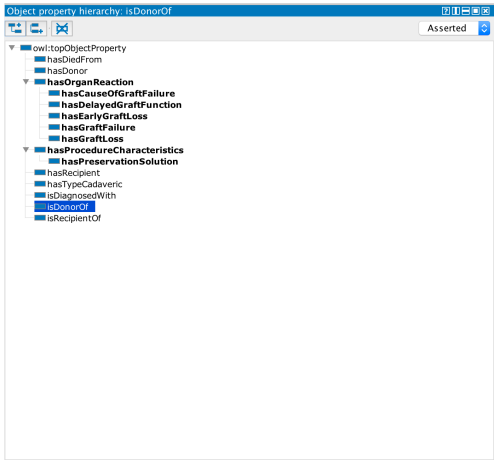



(a) NHB class restriction

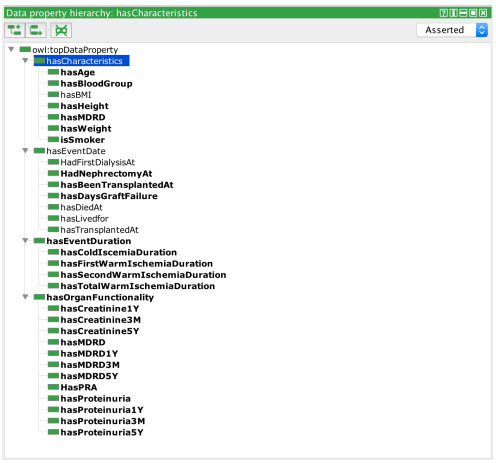



(b) NHB class properties

Figure 4.3: Class description for NHB




(a) Object properties




(b) Data properties

Figure 4.4: Properties of the transplantation ontology.

# Chapter 5

# Conclusions

Despite the fact that the data used in this project were incomplete and noisy, we were able to to apply a set of data science methods to clean and prepare the data. This enabled the construction of two predictive models that are capable of forecasting graft survival after transplantation by exploiting all the information provided for the donor-recipients pairs. Although ANNs have the ability to cope with an increased number of factors during their learning phase, a number of parameters in function with the architecture of the network need to be optimized in order to enhance the efficiency of the predictive outcome. Correct predictions in graft survival can then be used to make appropriate decision on the allocation of the grafts, by distributing each organ to the patients who has the optimal characteristics that can ensure a successful transplantation and at the same time long term graft survival. To validate the performance of our models we employed 10-fold cross validation. The last but not least objective of this project was to construct the ontology that cover the conceptual knowledge of the data-set on hand and provide ground for future development of a Personalized decision Support System that may help clinicians to make more accurate decision in organ allocation. The proposed pre-operative model achieves an accuracy of 76.8% in correctly predicting graft survival, whilst the post-operative model is 96.8% accurate.

## 5.1 Research questions

At the introduction (section 1.4.1) of this project we addressed some research questions. After completing this project we can conclude to the answers of these questions.

**What data mining techniques can be applied to deal with incomplete and class imbalanced medical data ?**

As we already discussed multiple times throughout this report, medical data are characterized by their inconsistency and data incompleteness. To this end, machine learning methods and specifically Neural Networks require a complete dataset in order to function. For this reason we imputed the missing information in a way that the variance of the data

remains high, while the imported bias is kept as low as possible. The main approach to impute the missing information was the multiple conditional mean. However, we also employed mathematical calculation to fill missing values for variables that could be achieved from other variables.

Class imbalanced data is an important factor in predictive models, which if it is not taken into consideration can mislead the predictive outcome. In our study about 80% of the examples are classified in the class that represents no graft loss and the remaining 20% is representing graft loss. To deal with that, we assigned different weight on each class, so that during the training phase of the algorithm the two classes can be faced equally.

**Are Artificial Neural Networks capable of predicting graft survival, given recipient - donor information ?**

The two models we developed using ANNs show a highly accuracy in predicting the correct outcome based on the provided real outcome. In order to make high accurate predictions, the neural network models learn all the provided information and according to them classify the examples. In contrast clinicians make decisions based on a limited number of parameters. Having said that, the proposed model can effectively support medical decision and enhance the probabilities of allocating grafts efficient.

**How do the models perform ? Do the results of cross validation show similar results?**

The pre-operative model shows an accuracy of 76.8% and the post-operative model an accuracy of 96.8%. The difference in the accuracy between the models is due to the fact that the post-operative model is trained with larger amount of information that help the classifier make more precise predictions, while the information provided for the pre-operative model are not descriptive enough to generalize better. These results can be confirmed with other prognostic and diagnostic models proposed in literature that employ ANNs to correctly forecast the desired outcome. The results of the cross validation prove the high accuracy of the two proposed models. The cross validated pre-operative model achieves an accuracy of 73.2% and the post-operative model 89%. This difference in accuracy can be explained by the nature of how cross validation works. In other words, the 10-fold cross validation is using all the data to get trained and tested while the individual model is trained in a random set of 80% of the data and validated in te remaining 20%.

**How will the proposed ontology be useful for the problem ?**

The proposed ontology covers the conceptual knowledge of the transplantations data and provides individual information for the recipients and the donors. This knowledge can be the basis to develop a Personalized Decision Support System to assist clinicians in the graft allocation process and support their decisions. To this end, in section 2.1.2 we discussed about the clinicians knowledge about the data-set which was illustrated in the graph 2.2. According to this knowledge, kidney transplantations appears to be elaborate.

However, the knowledge obtained from the semantic model for the same data is depicted in the graph 4.1, expresses the data simpler and more comprehensible.

## 5.2 Future work

As we discussed in chapter 4, we constructed a semantic models that provides the conceptual knowledge of the domain of kidney transplantation for deceased donors. This ontology can consist the ground of developing a Personalized Decision Support System by integrating the outcome of the the proposed predictive models as a probability value relating to graft survival. This system can then be used by clinicians to access and interpret these results and aid them to make more accurate decision regarding kidney allocation.

As we discussed before, the preoperative model performs worse than the model trained in the whole dataset. However, if we had more information about the donors and the recipients history, we could improve the performance of the model. For example, according to the allocation protocol provided by Eurotransplant, the first criterion for graft allocation is the compatibility of donors and recipients blood group. Nevertheless, in our data-set the variable that represent the donors blood group did not exist. In addition, if we had more information about the recipients history before the transplantation, probably this model would perform better. To this end, if the recorded data from the dialysis registry were integrated in our dataset the predictive performance of our models may improve.

The predictive results of this study can be encouraging to develop models to answer other research questions in the complex topic of transplantations. For example, we have read many studies that discusses the importance of DGF in graft survival or the PNF. Similar models can be developed to predict these outcomes as well and incorporate all these models to support clinical decisions.

To conclude this project, we suggest the establishment of a strict protocol regarding the recording of patients information in EHRs. In this way, the captured information would be more valid and complete, allowing researchers to gain more accurate insights from the collected data. In addition, if the data are homogeneous and structured in a same way, different data-set can be integrated and provide more realistic knowledge in the medical sector.

# Bibliography


[1] G Karam et al. “Guidelines on Renal Transplantation”. In: *Renal Transplantation - European Association of Urology* March (2009), pp. 327–337. ISSN: 0028-4793. DOI: 10.1056/NEJM199408113310606.

[2] Iván Ortega-Deballon, Laura Hornby and Sam D. Shemie. “Protocols for uncontrolled donation after circulatory death: A systematic review of international guidelines, practices and transplant outcomes”. In: *Critical Care* 19.1 (2015). ISSN: 1466609X. DOI: 10.1186/s13054-015-0985-7. URL: http://dx.doi.org/10.1186/s13054-015-0985-7.

[3] Jacqueline Van De Wetering, Arjan D Van Zuilen and H L Maarten. “Equivalent long-term transplantation outcomes for kidneys donated after brain death and cardiac death: Conclusions from a nationwide evaluation”. In: (), pp. 1–22.

[4] R. A. Wolfe et al. “Comparison of Mortality in All Patients on Dialysis, Patients on Dialysis Awaiting Transplantation, and Recipients of a First Cadaveric Transplant”. In: *New England Journal of Medicine,* 342.23 (1999), pp. 1725–1730. DOI: doi:10.1056/nejm199912023412303.

[5] Janet M. Torpy, Nicholas Barnett and Nizam Mamode. “Kidney transplantation”. In: *Surgery - Oxford International Edition* 305.6 (2011). ISSN: 0263-9319. DOI: 10.1001/jama.305.6.634. URL: http://dx.doi.org/10.1001/jama.305.6.634.

[6] A. Peres Penteado et al. “Kidney transplantation process in Brazil represented in business process modeling notation”. In: *Transplantation Proceedings* 47.4 (2015), pp. 963–966. ISSN: 18732623. DOI: 10.1016/j.transproceed.2015.03.044. URL: http://dx.doi.org/10.1016/j.transproceed.2015.03.044.

[7] A. J. Collins et al. “Mortality risks of peritoneal dialysis and hemodialysis”. In: *American Journal of Kidney Diseases* 34.6 (1999), pp. 1065–1074. ISSN: 02726386. DOI: 10.1016/S0272-6386(99)70012-0. URL: http://dx.doi.org/10.1016/S0272-6386(99)70012-0.

[8] AS Go and GM Chertow. “Chronic kidney disease and the risks of death, cardiovascular events, and hospitalization”. In: *New England Journal ...* (2004), pp. 1296–1305. ISSN: 1533-4406. DOI: 10.1056/NEJMoa041031. URL: http://www.nejm.org/doi/full/10.1056/NEJMoa041031.

[9] Robert a Metzger et al. "Expanded criteria donors for kidney transplantation." In: *American journal of transplantation : official journal of the American Society of Transplantation and the American Society of Transplant Surgeons* 3 Suppl 4 (2003), pp. 114–125. ISSN: 1600-6135. DOI: 10.1034/j.1600-6143.3.s4.11.x.

[10] Michael Abecassis et al. "Kidney transplantation as primary therapy for end-stage renal disease: A National Kidney Foundation/Kidney Disease Outcomes Quality Initiative (NKF/KDOQI) conference". In: *Clinical Journal of the American Society of Nephrology* 3.2 (2008), pp. 471–480. ISSN: 15559041. DOI: 10.2215/CJN.05021107.

[11] Adnan Sharif and Richard Borrows. "Delayed graft function after kidney transplantation: The clinical perspective". In: *American Journal of Kidney Diseases* 62.1 (2013), pp. 150–158. ISSN: 02726386. DOI: 10.1053/j.ajkd.2012.11.050. URL: http://dx.doi.org/10.1053/j.ajkd.2012.11.050.

[12] Canadian Transplant foundation. "Clinical Guidelines for Kidney Transplantation". In: June (2018), pp. 1–66.

[13] Tommaso Di Noia et al. "An end stage kidney disease predictor based on an artificial neural networks ensemble". In: *Expert Systems with Applications* 40.11 (2013), pp. 4438–4445. ISSN: 09574174. DOI: 10.1016/j.eswa.2013.01.046. URL: http://dx.doi.org/10.1016/j.eswa.2013.01.046.

[14] Eurotransplant Foundation. "Eurotransplant Manual 4.0". In: (2016). URL: https://www.eurotransplant.org/cms/mediaobject.php?file=H1+Introduction+July+28+20161.pdf.

[15] Ch Legendre and H. Kreis. "A tribute to jean hamburger's contribution to organ transplantation". In: *American Journal of Transplantation* 10.11 (2010), pp. 2392–2395. ISSN: 16006135. DOI: 10.1111/j.1600-6143.2010.03295.x.

[16] Friedrich K. Port et al. "Donor characteristics associated with reduced graft survival: An approach to expanding the pool of kidney donors". In: *Transplantation* 74.9 (2002), pp. 1281–1286. ISSN: 00411337. DOI: 10.1097/00007890-200211150-00014.

[17] Ravi Parasuraman et al. "Primary Nonfunction of Renal Allograft Secondary to Acute Oxalate Nephropathy". In: *Case Reports in Transplantation* 2011 (2011), pp. 1–4. ISSN: 2090-6943. DOI: 10.1155/2011/876906. URL: http://www.hindawi.com/journals/crit/2011/876906/.

[18] Frans J. van Ittersum et al. "Increased risk of graft failure and mortality in Dutch recipients receiving an expanded criteria donor kidney transplant". In: *Transplant International* 30.1 (2017), pp. 14–28. ISSN: 14322277. DOI: 10.1111/tri.12863.

[19] Karthik K. Tennankore et al. "Prolonged warm ischemia time is associated with graft failure and mortality after kidney transplantation". In: *Kidney International* 89.3 (2016), pp. 648–658. ISSN: 15231755. DOI: 10.1016/j.kint.2015.09.002. URL: http://dx.doi.org/10.1016/j.kint.2015.09.002.

[20] Jacqueline M A Smits et al. "Evaluation of the Eurotransplant Senior Program. The Results of the First Year on behalf of all the Eurotransplant Senior Program Centers d". In: *American Journal of Transplantation* 2 (2002), pp. 664–670. ISSN: 1600-6135.

[21] Lutz Fritsche et al. "Old-for-Old Kidney Allocation Allows Successful Expansion of the Donor and Recipient Pool". In: *American Journal of Transplantation* 3.11 (2003), pp. 1434–1439. ISSN: 16006135. DOI: 10.1046/j.1600-6135.2003.00251.x.

[22] A. B. Massie et al. "Early Changes in Kidney Distribution under the New Allocation System". In: *Journal of the American Society of Nephrology* 27.8 (2016), pp. 2495–2501. ISSN: 1046-6673. DOI: 10.1681/ASN.2015080934. URL: http://www.jasn.org/cgi/doi/10.1681/ASN.2015080934.

[23] Philippe Lambin et al. "Decision support systems for personalized and participative radiation oncology". In: *Advanced Drug Delivery Reviews* 109 (2017), pp. 131–153. ISSN: 18728294. DOI: 10.1016/j.addr.2016.01.006. URL: http://dx.doi.org/10.1016/j.addr.2016.01.006.

[24] Michael W. Kattan. "When and how to use informatics tools in caring for urologic patients". In: *Nature Clinical Practice Urology* 2.4 (2005), pp. 183–190. ISSN: 17434270. DOI: 10.1038/ncpuro0144.

[25] Kidney Disease Improving Global Outcomes Kdigo Transplant Work Group. "KDIGO clinical practice guideline for the care of kidney transplant recipients". In: *American journal of transplantation* 9.3 (2009), S1–155. ISSN: 1600-6143. DOI: 10.1111/j.1600-6143.2009.02834.x. URL: papers2://publication/doi/10.1111/j.1600-6143.2009.02834.x.

[26] Tim Lustberg et al. "Implementation of a rapid learning platform: Predicting 2-year survival in laryngeal carcinoma patients in a clinical setting". In: *Oncotarget* 7.24 (2016). ISSN: 1949-2553. DOI: 10.18632/oncotarget.8755. URL: http://www.oncotarget.com/fulltext/8755.

[27] I. P. Vaughan and S. J. Ormerod. "Improving the Quality of Distribution Models for Conservation by Addressing Shortcomings in the Field Collection of Training Data". In: *Conservation Biology* 17.6 (2003), pp. 1601–1611. ISSN: 08888892. DOI: 10.1111/j.1523-1739.2003.00359.x.

[28] Michael E. Brier, Prasun C. Ray and Jon B. Klein. "Prediction of delayed renal allograft function using an artificial neural network". In: *Nephrology Dialysis Transplantation* 18.12 (2003), pp. 2655–2659. ISSN: 09310509. DOI: 10.1093/ndt/gfg439.

[29] Howard Doyle et al. "Predicting Outcomes After Liver Transplantation". In: 219.4 (1994), pp. 408–415.

[30] Mohammed J. Islam, Majid Ahmadi and Maher A. Sid-Ahmed. "An Efficient Automatic Mass Classification Method In Digitized Mammograms Using Artificial Neural Network". In: *International Journal of Artificial Intelligence & Applications* 1.3 (2010), pp. 1–13. ISSN: 09762191. DOI: 10.5121/ijaia.2010.1301. arXiv: 1007.5129. URL: http://arxiv.org/abs/1007.5129%7B%5C%%7D5Cnhttp://www.airccse.org/journal/ijaia/papers/0710ijaia1.pdf.

[31] J Khan et al. "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks." In: *Nature medicine* 7.6 (2001), pp. 673–9. ISSN: 1078-8956. DOI: 10.1038/89044. URL: http://dx.doi.org/10.1038/89044.

[32] Chih-Lin Chi, W Nick Street and William H Wolberg. "Application of artificial neural network-based survival analysis on two breast cancer datasets." In: *AMIA ... Annual Symposium proceedings. AMIA Symposium* 2007 (2007), pp. 130–4. ISSN: 1942-597X. URL: http://www.ncbi.nlm.nih.gov/pubmed/18693812%7B%5C%%7D5Cnhttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2813661.

[33] P Furness et al. "A neural network approach to the biopsy diagnosis of early acute renal transplant rejection". In: *Histopathology* 35.5 (1999), pp. 461–467. URL: http://ovidsp.ovid.com/ovidweb.cgi?T=JS%7B%5C&%7DPAGE=reference%7B%5C&%7DD=emed4%7B%5C&%7DNEWS=N%7B%5C&%7DAN=1999392964.

[34] James E Tcheng et al. *Optimizing strategies for clinical decision support summary of a meeting series*. National Academy of Medicine, 2017, p. 96. ISBN: 9781947103092. URL: https://www.healthit.gov/sites/default/files/page/2018-04/Optimizing%7B%5C_%7DStrategies%7B%5C_%7D508.pdf.

[35] Kiawei Han, Micheline Kamber and Jian Pei. *Data mining : Concepts and techniques*. Elsevier, 2012. ISBN: 978-0-12-381479-1.

[36] Hyun Kang. "The prevention and handling of the missing data". In: *Korean Journal of Anesthesiology* 64.5 (2013), pp. 402–406. ISSN: 20056419. DOI: 10.4097/kjae.2013.64.5.402.

[37] Mohamed S. Barakat et al. "The effect of imputing missing clinical attribute values on training lung cancer survival prediction model performance". In: *Health Information Science and Systems* 5.1 (2017), p. 16. ISSN: 2047-2501. DOI: 10.1007/s13755-017-0039-4. URL: http://www.ncbi.nlm.nih.gov/pubmed/29255599%7B%5C%%7D0Ahttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5718991%7B%5C%%7D0Ahttp://link.springer.com/10.1007/s13755-017-0039-4.

[38] Brian J Wells et al. "Strategies for handling missing data in electronic health record derived data." In: *EGEMS (Washington, DC)* 1.3 (2013), p. 1035. ISSN: 2327-9214. DOI: 10.13063/2327-9214.1035. URL: http://www.pubmedcentral.nih.gov/

articlerender.fcgi?artid=4371484%7B%5C&%7Dtool=pmcentrez%7B%5C&%7Drendertype=abstract.

[39] Pedro J. García-Laencina, José-Luis Sancho-Gómez and Aníbal R. Figueiras-Vidal. "Pattern classification with missing data: a review". In: *Neural Computing and Applications* 19.2 (2010), pp. 263–282. ISSN: 0941-0643. DOI: 10.1007/s00521-009-0295-6. arXiv: arXiv:1011.1669v3. URL: http://link.springer.com/10.1007/s00521-009-0295-6.

[40] Eurotransplant. "Eurotranplant - Data Policy". In: (2017). URL: https://www.eurotransplant.org/cms/mediaobject.php?file=Eurotransplant+Data+policy.pdf.

[41] Mathieu Bastian, Sebastien Heymann and Mathieu Jacomy. "Gephi: An Open Source Software for Exploring and Manipulating Networks". In: *Third International AAAI Conference on Weblogs and Social Media* (2009), pp. 361–362. ISSN: 14753898. DOI: 10.1136/qshc.2004.010033. URL: http://www.aaai.org/ocs/index.php/ICWSM/09/paper/view/154%7B%5C%%7D5Cnpapers2://publication/uuid/CCEBC82E-0D18-4FFC-91EC-6E4A7F1A1972.

[42] Loet Leydesdorff. "Betweenness Centrality as an Indicator of the Interdisciplinarity of Scientific Journals". In: *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE AND TECHNOLOGY,* 58.8 (2007), pp. 1303–1319. DOI: https://doi.org/10.1002/asi.20614.

[43] K. J. Halazun et al. "Warm Ischemia in Transplantation: Search for a Consensus Definition". In: *Transplantation Proceedings* 39.5 (2007), pp. 1329–1331. ISSN: 00411345. DOI: 10.1016/j.transproceed.2007.02.061.

[44] M M Mukaka. "A guide to appropriate use of Correlation coefficient in medical research." In: *Malawi Medical Journal* 24.3 (2012), pp. 69–71. ISSN: 1995-7270. DOI: 10.1016/j.cmpb.2016.01.020.

[45] Hrishikesh Chakraborty. "A mixed model approach for intent-to-treat analysis in longitudinal clinical trials with missing values". In: March (2009). DOI: 10.3768/rtipress.2009.mr.0009.0903.

[46] Katya L. Masconi et al. "Reporting and handling of missing data in predictive research for prevalent undiagnosed type 2 diabetes mellitus: A systematic review". In: *EPMA Journal* 6.1 (2015). ISSN: 18785085. DOI: 10.1186/s13167-015-0028-0.

[47] Geert J.M.G. van der Heijden et al. "Imputation of missing values is superior to complete case analysis and the missing-indicator method in multivariable diagnostic research: A clinical example". In: *Journal of Clinical Epidemiology* 59.10 (2006), pp. 1102–1109. ISSN: 08954356. DOI: 10.1016/j.jclinepi.2006.01.015.

[48] Denis Cousineau and Sylvain Chartier. "Outliers detection and treatment: a review." In: *International Journal of Psychological Research* 3.1 (2015), pp. 58–67. ISSN: 2011-7922. DOI: 10.21500/20112084.844.

[49] Marco Aste et al. “Techniques for dealing with incomplete data: a tutorial and survey”. In: *Pattern Analysis and Applications* 18.1 (2014), pp. 1–29. ISSN: 14337541. DOI: 10.1007/s10044-014-0411-9.

[50] Jonathan Sterne et al. “Multiple imputation for missing data in epidemiological and clinical research: potential and pitfalls”. In: *BMJ* 339 (2009), p. 144. ISSN: 14685833. DOI: 10.1136/bmj.b2393.

[51] A. Rogier T. Donders et al. “Review: A gentle introduction to imputation of missing values”. In: *Journal of Clinical Epidemiology* 59.10 (2006), pp. 1087–1091. ISSN: 08954356. DOI: 10.1016/j.jclinepi.2006.01.014.

[52] Ulpu Remes et al. “Bounded conditional mean imputation with observation uncertainties and acoustic model adaptation”. In: *IEEE Transactions on Audio, Speech and Language Processing* 23.7 (2015), pp. 1198–1208. ISSN: 15587916. DOI: 10.1109/TASLP.2015.2424322.

[53] Joseph L. Schafer and Nathaniel Schenker. “Inference with Imputed Conditional Means”. In: *Journal of the American Statistical Association* 95.449 (2000), pp. 144–154. ISSN: 1537274X. DOI: 10.1080/01621459.2000.10473910.

[54] Burak Sayin et al. “Comparison of preemptive kidney transplant recipients with nonpreemptive kidney recipients in single center: 5 years of follow-up”. In: *International Journal of Nephrology and Renovascular Disease* 6 (2013), pp. 95–99. ISSN: 11787058. DOI: 10.2147/IJNRD.S42042.

[55] Selwyn Piramuthu, Harish Ragavan and Michael J. Shaw. “Using Feature Construction to Improve the Performance of Neural Networks”. In: *Management Science* 44.3 (1998), pp. 416–430. ISSN: 0025-1909. DOI: 10.1287/mnsc.44.3.416. URL: http://pubsonline.informs.org/doi/abs/10.1287/mnsc.44.3.416.

[56] Huimin Zhao, Atish P. Sinha and Wei Ge. “Effects of feature construction on classification performance: An empirical study in bank failure prediction”. In: *Expert Systems with Applications* 36.2 PART 2 (2009), pp. 2633–2644. ISSN: 09574174. DOI: 10.1016/j.eswa.2008.01.053. URL: http://dx.doi.org/10.1016/j.eswa.2008.01.053.

[57] Matthew G. Smith and Larry Bull. “Genetic programming with a genetic algorithm for feature construction and selection”. In: *Genetic Programming and Evolvable Machines* 6.3 (2005), pp. 265–281. ISSN: 13892576. DOI: 10.1007/s10710-005-2988-7.

[58] Daniel Martin Katz et al. “A general approach for predicting the behavior of the Supreme Court of the United States”. In: *PloS one* 12.4 (2017), e0174698. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0174698. arXiv: arXiv:1407.6333v1. URL: http://www.ncbi.nlm.nih.gov/pubmed/28403140%7B%5C%%7D0Ahttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5389610.

[59] Philippe Leray and Patrick Gallinari. "Feature Selection with Neural Networks Feature Selection with Neural Networks". In: *Behaviormetrika* 26 (1998), pp. 16–6. ISSN: 0385-7417. DOI: 10.2333/bhmk.26.145. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.54.4570.

[60] Isabelle Guyon and André Elisseeff. "An Introduction to Variable and Feature Selection". In: *Journal of Machine Learning Research (JMLR)* 3.3 (2003), pp. 1157–1182. ISSN: 00032670. DOI: 10.1016/j.aca.2011.07.027. arXiv: 1111.6189v1.

[61] Isabelle Guyon and Andre Elisseeff. "An Introduction to Variable and Feature Selection". In: *Journal of Machine Learning Research (JMLR)* 3 (2003), pp. 1157–1182. ISSN: 00032670. DOI: 10.1016/j.aca.2011.07.027.

[62] Matthew Shardlow. "An Analysis of Feature Selection Techniques". In: *The University of Manchester* (2016), pp. 1–7.

[63] Sunita Parashar and Sharuti Sogi. "Finding skewness and deskewing scanned document". In: 3 (2012), pp. 1619–1624.

[64] Jonathon Shlens. "A Tutorial on Principal Component Analysis". In: *Google Research* (2014). DOI: 10.1.1.115.3503.

[65] Peter Harrington. *Machine Learning in Action*. January. Manning Publications Co, 2012. ISBN: 9781617290183.

[66] Filippo Amato et al. "Artificial neural networks in medical diagnosis". In: *Journal of Applied Biomedicine* 11.2 (2013), pp. 47–58. ISSN: 12140287. DOI: 10.2478/v10136-012-0031-x.

[67] Stergios Theodoridis and Konstantinos Koutroumbas. *Pattern Recognition*. Vol. 4th Edition. Elsener, 2009. ISBN: 9781597492720.

[68] Ethem Alpaydin. *Introduction to Machine Learning*. Vol. 3rd Edition. MIT Press, 2014. ISBN: 978-0-262-01243-0.

[69] Aurlien Geron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 1st. O'Reilly Media, Inc., 2017. ISBN: 1491962291, 9781491962299.

[70] S. Haykin. *Neural Networks and Learning Machines*. 2008, p. 906. ISBN: 9780131471399. DOI: 978-0131471399. arXiv: arXiv:1312.6199v4.

[71] W KinneBrock. "Neural Networks". In: *Oldenburg Verlag* (1992). ISSN: 08936080. DOI: 10.1016/0893-6080(94)90051-5. arXiv: arXiv:1411.3159v1.

[72] Brenden M. Lake, Ruslan Salakhutdinov and Joshua B. Tnenbaum. "Program Induction". In: *Science* 350.6266 (2015), pp. 1332–1338. DOI: 10.1126/science.aab3050.

[73] Wiesław Chmielnicki and Katarzyna Stąpor. "Investigation of Normalization Techniques and Their Impact on a Recognition Rate in Handwritten Numeral Recognition". In: *Schedae Informaticae* 19.-1 (2011), pp. 53–77. ISSN: 0860-0295. DOI: 10.2478/v10149-011-0004-y.

[74] Ahmed Akl, Amani M. Ismail and Mohamed Ghoneim. "Prediction of Graft Survival of Living-Donor Kidney Transplantation: Nomograms or Artificial Neural Networks?" In: *Transplantation* 86.10 (2008), pp. 1401–1406. ISSN: 00411337. DOI: 10.1097/TP.0b013e31818b221f.

[75] Daniel J Sargent. "Comparison of artificial neural networks with other statistical approaches". In: *Cancer* 91.S8 (2001), pp. 1636–1642. ISSN: 1097-0142. DOI: 10.1002/1097-0142(20010415)91:8+<1636::AID-CNCR1176>3.0.CO;2-D.

[76] Paulo J. Lisboa and Azzam F.G. Taktak. "The use of artificial neural networks in decision support in cancer: A systematic review". In: *Neural Networks* 19.4 (2006), pp. 408–415. ISSN: 08936080. DOI: 10.1016/j.neunet.2005.10.007.

[77] Richard Dybowski and Vanya Gant. "Clinical applications of artificial neural networks". In: *System* (2001). DOI: 10.1017/CBO9780511543494. URL: http://ebooks.cambridge.org/ref/id/CBO9780511543494.

[78] Bruce Alberts et al. *Molecular Biology of the Cell*. Vol. 6th Edition. 2014. ISBN: 9780815344537.

[79] D. Rumelhart, G. Hinton and R. William. "Learning Internal Representations by Error Propagation". In: *CALIFORNIA UNIV SAN DIEGO LA JOLLA INST FOR COGNITIVE SCIENCE* (1985).

[80] Lars Buitinck et al. "API design for machine learning software: experiences from the scikit-learn project". In: *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*. 2013, pp. 108–122.

[81] Amelia J. Averitt and Karthik Natarajan. "Going Deep: The Role of Neural Networks for Renal Survival and Beyond". In: *Kidney International Reports* 3.2 (2018), pp. 242–243. ISSN: 24680249. DOI: 10.1016/j.ekir.2017.12.006. URL: https://doi.org/10.1016/j.ekir.2017.12.006.

[82] Robert S. Ledley and Lee B. Lusted. "Reasoning foundations of medical diagnosis". In: *Science* 130.3366 (1959), pp. 9–21. ISSN: 00368075. DOI: 10.1126/science.130.3366.9. arXiv: arXiv:1011.1669v3.

[83] Orhan Er, Feyzullah Temurtas and A. Çetin Tanrfffdfffdkulu. "Tuberculosis Disease Diagnosis Using Artificial Neural Networks". In: *Journal of Medical Systems* 34.3 (2010), pp. 299–302. ISSN: 0148-5598. DOI: 10.1007/s10916-008-9241-x. URL: http://link.springer.com/10.1007/s10916-008-9241-x.

[84] François Chollet et al. *Keras*. https://keras.io. 2015.

[85] Lars Buitinck et al. "API design for machine learning software: experiences from the scikit-learn project". In: (2013), pp. 1–15. arXiv: 1309.0238. URL: http://arxiv.org/abs/1309.0238.

[86] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[87] Maciej A. Mazurowski et al. "Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance". In: *Neural Networks* 21.2-3 (2008), pp. 427–436. ISSN: 08936080. DOI: 10.1016/j.neunet.2007.12.031. arXiv: NIHMS150003.

[88] Shoujin Wang et al. "Training deep neural networks on imbalanced data sets". In: *2016 International Joint Conference on Neural Networks (IJCNN)* (2016), pp. 4368–4374.

[89] N Chawla et al. "{SMOTE}: {S}ynthetic minority over-sampling technique". In: *Journal of Artificial Intelligence Research* 16 (2002), pp. 321–357. ISSN: 10769757. DOI: 10.1613/jair.953. arXiv: 1106.1813.

[90] Eduardo D. Sontag. "Feedback Stabilization Using Two-Hidden-Layer Nets". In: *IEEE Transactions on Neural Networks* 3.6 (1992), pp. 981–990. ISSN: 19410093. DOI: 10.1109/72.165599.

[91] Vinod Nair and Geoffrey E Hinton. "Rectified Linear Units Improve Restricted Boltzmann Machines". In: *Proceedings of the 27th International Conference on Machine Learning* 3 (2010), pp. 807–814. ISSN: 1935-8237. DOI: 10.1.1.165.6419. arXiv: 1111.6189v1.

[92] Nitish Srivastava et al. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *Journal of Machine Learning Research* 15 (2014), pp. 1929–1958. ISSN: 15337928. DOI: 10.1214/12-AOS1000. arXiv: 1102.4807.

[93] Elad Hoffer, Itay Hubara and Daniel Soudry. "Train longer, generalize better: closing the generalization gap in large batch training of neural networks". In: (2017). ISSN: 10495258. arXiv: 1705.08741. URL: http://arxiv.org/abs/1705.08741.

[94] Sergey Ioffe and Christian Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: (2015). ISSN: 0717-6163. DOI: 10.1007/s13398-014-0173-7.2. arXiv: 1502.03167. URL: http://arxiv.org/abs/1502.03167.

[95] Christos Baziotis, Nikos Pelekis and Christos Doulkeridis. "DataStories at SemEval-2017 Task 6: Siamese LSTM with Attention for Humorous Text Comparison". In: *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)* (2017), pp. 389–394. URL: http://www.aclweb.org/anthology/S17-2065.

[96] Jian Zhang and Ioannis Mitliagkas. "YellowFin and the Art of Momentum Tuning". In: (2017), pp. 1–27. arXiv: 1706.03471. URL: http://arxiv.org/abs/1706.03471.

[97] Samuel L. Smith et al. "Don't Decay the Learning Rate, Increase the Batch Size". In: 2017 (2017), pp. 1–11. arXiv: 1711.00489. URL: http://arxiv.org/abs/1711.00489.

[98] Tijmen Tieleman. "Training restricted Boltzmann machines using approximations to the likelihood gradient". In: *Proceedings of the 25th international conference on Machine learning - ICML '08* (2008), pp. 1064–1071. ISSN: 21576904. DOI: 10.1145/1390156.1390290. URL: http://portal.acm.org/citation.cfm?doid=1390156.1390290.

[99] A Krogh and J Vedelsby. "Neural network ensembles, cross validation, and active learning". In: *Advances in neural network processing systems* 7 (1995), pp. 8–231. ISSN: 10495258. DOI: 10.1.1.37.8876. URL: internal-pdf://kroghandvedelsby-4069001218/KroghandVedelsby.pdf.

[100] Tzu Tsung Wong. "Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation". In: *Pattern Recognition* 48.9 (2015), pp. 2839–2846. ISSN: 00313203. DOI: 10.1016/j.patcog.2015.03.009. URL: http://dx.doi.org/10.1016/j.patcog.2015.03.009.

[101] Juan Diego Rodríguez, Aritz Pérez and Jose Antonio Lozano. "Sensitivity analysis of kappa-fold cross validation in prediction error estimation". In: *IEEE transactions on pattern analysis and machine intelligence* 32.3 (2010), pp. 569–575. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2009.187. arXiv: 9605103 [cs].

[102] Frederico Freitas, Stefan Schulz and Eduardo Moraes. "Biology_medicine". In: *Reciis* 3.1 (2009). ISSN: 1981-6286. DOI: 10.3395/reciis.v3i1.239en. URL: http://www.reciis.cict.fiocruz.br/index.php/reciis/article/view/239/249.

[103] Furkh Zeshan and Radziah Mohamad. "Medical ontology in the dynamic healthcare environment". In: *Procedia Computer Science* 10 (2012), pp. 340–348. ISSN: 18770509. DOI: 10.1016/j.procs.2012.06.045. URL: http://dx.doi.org/10.1016/j.procs.2012.06.045.

[104] Miguel-Ángel Sicilia. "Metadata, semantics, and ontology: providing meaning to information resources". In: *Int. J. Metadata, Semantics and Ontologies J. Metadata, Semantics and Ontologies* 1.1 (2006), pp. 83–86. ISSN: 1744-2621. DOI: 10.1504/IJMSO.2006.008773.

[105] a. Jovic, M. Prcela and D. Gamberger. "Ontologies in Medical Knowledge Representation". In: *2007 29th International Conference on Information Technology Interfaces* (2007), pp. 535–540. ISSN: 1330-1012. DOI: 10.1109/ITI.2007.4283828.

[106] Claudio Eccher et al. "An ontology of cancer therapies supporting interoperability and data consistency in EPRs". In: *Computers in Biology and Medicine* 43.7 (2013), pp. 822–832. ISSN: 00104825. DOI: 10.1016/j.compbiomed.2013.04.012. URL: http://dx.doi.org/10.1016/j.compbiomed.2013.04.012.

[107] Natalya F. Noy and Deborah L. McGuinness. "Ontology Development 101: A Guide to Creating Your First Ontology". In: *Stanford Knowledge Systems Laboratory* (2001), p. 25. ISSN: 09333657. DOI: 10.1016/j.artmed.2004.01.014. arXiv: 1304.1186.

[108] David Riaño et al. "An ontology-based personalization of health-care knowledge to support clinical decisions for chronically ill patients". In: *Journal of Biomedical Informatics* 45.3 (2012), pp. 429–446. ISSN: 15320464. DOI: 10.1016/j.jbi.2011.12.008.

[109] Matthew Horridge et al. *A Practical Guide To Building OWL Ontologies Using Protege 4 and CO-ODE ToolsCopyright c TMarch 13, 2009*. Vol. 1.3. 2009, pp. 0–108.

[110] Marco Alfonse, Mostafa M. Aref and Abdel-Badeeh M. Salem. "Alfonse_Cancer". In: *International Journal of Information Engineering and Electronic Business* 6.6 (2014), pp. 55–63. ISSN: 20749023. DOI: 10.5815/ijieeb.2014.06.07. URL: http://www.mecs-press.org/ijieeb/ijieeb-v6-n6/v6n6-7.html.

[111] Carmen Martinez-Cruz, Ignacio J. Blanco and M. Amparo Vila. "Ontologies versus relational databases: Are they so different? A comparison". In: *Artificial Intelligence Review* 38.4 (2012), pp. 271–290. ISSN: 02692821. DOI: 10.1007/s10462-011-9251-9.

# Appendices

# Appendix A

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| **d_cadaveric_type** | 10410.0 | 0.705 | 0.455 | 0.0 | 0.0 | 1.0 | 1.0 | 1.0 |
| **d_NHB_cat** | 3064.0 | 3.987 | 31.155 | 3.0 | 3.0 | 3.0 | 3.0 | 999.0 |
| **d_death_cause_cat** | 10410.0 | 2.075 | 0.982 | 1.0 | 1.0 | 2.0 | 2.0 | 4.0 |
| **d_age** | 10409.0 | 46.338 | 16.436 | 0.0 | 36.0 | 50.0 | 59.0 | 86.0 |
| **d_sex** | 10410.0 | 0.458 | 0.498 | 0.0 | 0.0 | 0.0 | 1.0 | 1.0 |
| **d_height** | 9725.0 | 173.144 | 12.446 | 60.0 | 168.0 | 175.0 | 180.0 | 210.0 |
| **d_weight** | 9748.0 | 75.067 | 16.583 | 4.0 | 65.0 | 75.0 | 85.0 | 185.0 |
| **d_BMI** | 9712.0 | 24.810 | 4.304 | 9.693 | 22.491 | 24.489 | 26.573 | 66.597 |
| **d_diabetes** | 5474.0 | 3.898 | 0.489 | 1.0 | 4.0 | 4.0 | 4.0 | 5.0 |
| **d_smoking** | 7744.0 | 0.552 | 0.564 | 0.0 | 0.0 | 1.0 | 1.0 | 2.0 |
| **d_hypotensive_periods_duration** | 3369.0 | 53.388 | 99.259 | 0.0 | 10.0 | 20.0 | 60.0 | 999.0 |
| **d_creatinine_highest** | 10364.0 | 105.699 | 429.626 | 0.09 | 63.0 | 78.0 | 91.0 | 9989.2 |
| **d_creatinine_last** | 10363.0 | 105.005 | 429.878 | 0.09 | 56.0 | 72.0 | 92.0 | 13260.0 |
| **d_MDRD** | 10150.0 | 101.875 | 44.563 | 4.214 | 72.423 | 95.181 | 122.349 | 579.90 |
| **preservation_solution_type_cat** | 10410.0 | 5.927 | 1.532 | 1.0 | 4.0 | 7.0 | 7.0 | 10.0 |
| **retransplant** | 10410.0 | 0.279 | 0.664 | 0.0 | 0.0 | 0.0 | 0.0 | 6.0 |
| **r_age** | 10410.0 | 50.353 | 14.428 | 12.0 | 41.0 | 52.0 | 62.0 | 85.0 |
| **r_sex** | 10410.0 | 0.399 | 0.489 | 0.0 | 0.0 | 0.0 | 1.0 | 1.0 |
| **r_height** | 8271.0 | 171.163 | 10.467 | 59.0 | 165.0 | 171.0 | 178.0 | 210.0 |
| **r_weight** | 8345.0 | 73.860 | 15.241 | 23.0 | 63.0 | 73.0 | 83.0 | 176.0 |
| **r_BMI** | 8038.0 | 25.168 | 4.386 | 10.810 | 22.052 | 24.690 | 27.770 | 48.442 |
| **r_initial_disease_recurrent** | 9958.0 | 1.001 | 0.726 | 0.0 | 0.0 | 1.0 | 2.0 | 2.0 |
| **r_PRA** | 10407.0 | 6.590 | 18.290 | 0.0 | 0.0 | 0.0 | 2.0 | 100.0 |
| **mismatch_DR** | 10377.0 | 0.584 | 0.592 | 0.0 | 0.0 | 1.0 | 1.0 | 2.0 |
| **mismatch_A** | 10391.0 | 0.764 | 0.664 | 0.0 | 0.0 | 1.0 | 1.0 | 2.0 |
| **mismatch_B** | 10391.0 | 0.929 | 0.670 | 0.0 | 0.0 | 1.0 | 1.0 | 2.0 |
| **r_last_dialysis_technique_cat** | 10228.0 | 3.840 | 1.670 | 1.0 | 3.0 | 4.0 | 4.0 | 12.0 |
| **r_pre_emptive_transplant** | 8011.0 | 0.144 | 0.484 | 0.0 | 0.0 | 0.0 | 0.0 | 2.0 |
| **ischaemic_period_warm_1** | 9563.0 | 5.819 | 9.498 | 0.0 | 0.0 | 0.0 | 13.0 | 90.0 |
| **ischaemic_period_cold** | 9749.0 | 1243.375 | 479.997 | 0.0 | 890.0 | 1198.0 | 1521.0 | 3211.0 |
| **ischaemic_period_warm_2** | 9654.0 | 34.776 | 12.898 | 0.0 | 26.0 | 33.0 | 40.0 | 180.0 |
| **r_delayed_graft_function** | 8080.0 | 1.882 | 0.694 | 0.0 | 1.0 | 2.0 | 2.0 | 3.0 |
| **r_creatinine_M3** | 8313.0 | 160.092 | 87.980 | 18.0 | 113.0 | 141.0 | 181.0 | 2373.0 |
| **r_proteinuria_M3** | 6433.0 | 0.296 | 1.451 | -1.0 | 0.1 | 0.1 | 0.3 | 66.0 |
| **r_MDRD_M3** | 8305.0 | 46.206 | 19.413 | 4.70 | 33.369 | 44.153 | 56.502 | 212.073 |
| **r_creatinine_Y1** | 7551.0 | 148.609 | 69.500 | 41.0 | 108.0 | 134.0 | 170.5 | 1377.0 |
| **r_proteinuria_Y1** | 5742.0 | 0.325 | 2.170 | 0.0 | 0.0 | 0.1 | 0.2 | 99.0 |
| **r_MDRD_Y1** | 7547.0 | 48.672 | 19.028 | 5.199 | 35.776 | 46.594 | 59.383 | 193.296 |
| **r_creatinine_Y5** | 4499.0 | 150.781 | 76.721 | 45.0 | 104.0 | 133.0 | 175.0 | 1318.0 |
| **r_proteinuria_Y5** | 3100.0 | 0.325 | 1.729 | 0.0 | 0.0 | 0.1 | 0.3 | 90.0 |
| **r_MDRD_Y5** | 4497.0 | 48.786 | 20.310 | 5.094 | 34.201 | 46.745 | 60.932 | 165.753 |
| **r_graft_fail_cause_cat** | 10410.0 | 1.809 | 3.184 | 0.0 | 0.0 | 1.0 | 3.0 | 16.0 |
| **graftloss** | 10410.0 | 0.269 | 0.443 | 0.0 | 0.0 | 0.0 | 1.0 | 1.0 |
| **early_graftloss2** | 10410.0 | 1.551 | 0.777 | 0.0 | 1.0 | 2.0 | 2.0 | 3.0 |
| **r_death_cause** | 4428.0 | 2.514 | 2.474 | 0.0 | 0.0 | 2.0 | 6.0 | 6.0 |
| **r_dead** | 10410.0 | 0.425 | 0.494 | 0.0 | 0.0 | 0.0 | 1.0 | 1.0 |
| **r_days_death** | 4431.0 | 2775.493 | 2160.764 | 0.0 | 1016.5 | 2373.0 | 4054.5 | 9970.0 |

Table 1: Statistics for all the variables in the dataset. For each variable the non missing values are counted, but also is calculated the mean, standart deviation, min , max and the $25^{th}$,$50^{th}$, $75^{th}$ percentile.

# Appendix B

| **Initial variable** | **Dummy-variables** |
|---|---|
| Type cadaveric | d_cadaveric_DBD<br>d_cadaveric_DCD |
| NHB category | d_nhb_Brain_dead<br>d_nhb_Cardiac_arrest_brain_stem_death<br>d_nhb_Awaiting_Cardiac_arrest |
| Donors' death cause | d_death_cause_Trauma<br>d_death_cause_Stroke<br>d_death_cause_Cardiac_Arrest<br>d_death_cause_Other |
| Donors' sex | d_sex_male<br>d_sex_female |
| Donor hypertension | donor_hypertension_yes<br>donor_hypertension_no |
| Donor diabetes | d_diabetes_type_1<br>d_diabetes_type_2<br>d_diabetes_type_unknown<br>d_diabetes_No |
| Donor smoking | d_smoking_yes<br>d_smoking_no |
| Preservation solution | preservation_solution_Bretshneider<br>preservation_solution_Celsior<br>preservation_solution_Eurocollins<br>preservation_solution_HTK<br>preservation_solution_IGL_1<br>preservation_solution_Modifies_UW<br>preservation_solution_UW<br>preservation_solution_Other<br>preservation_solution_Unknown<br>preservation_solution_Hartmann |
| Retransplantations | retranspl_Yes_only_one_tranps<br>retranspl_Yes_1st_retranp<br>retranspl_Yes_2nd_retranp<br>retranspl_Yes_3rd_retranp<br>retranspl_Yes_4th_retranp<br>retranspl_Yes_5th_retranp<br>retranspl_Yes_6th_retransp |
| Recipient sex | r_sex_male<br>r_sex_female |
| Initial disease recurrency | r_initial_disease_recurr_yes<br>r_initial_disease_recurr_no<br>r_initial_disease_recurr_unknown |
| Initial preemptive transplantation | r_pre_emptive_transplant_no<br>r_pre_emptive_transplant_yes |

| **Initial variable** | **Dummy-variables** |
|---|---|
| Recipient blood group | r_blood_group_O<br>r_blood_group_A<br>r_blood_group_B<br>r_blood_group_AB |
| Combined transplantations | r_combined_transplants_Rki<br>r_combined_transplants_Lki |
| Delayed graft function | delayed_graft_function_PNF<br>delayed_graft_function_delayed<br>delayed_graft_function_direct<br>delayed_graft_function_unknown |
| Graft failure cause | graft_fail_cause_not<br>graft_fail_cause_died_with_functioning_graft<br>graft_fail_cause_stop_immunosuppressive_drugs<br>graft_fail_cause_rejection_while_in_immunosupressive<br>graft_fail_cause_hyperacute_rejection<br>graft_fail_cause_non_viable_kidney<br>graft_fail_cause_permanent_non_function<br>graft_fail_cause_recurrent_primary_renal_disease<br>graft_fail_cause_infection_non_graft_related<br>graft_fail_cause_infection_of_graft<br>graft_fail_cause_thrombosis<br>graft_fail_cause_technical_problems<br>graft_fail_cause_vascular_ureteric_operative_prob<br>graft_fail_cause_vascular_non_operative<br>graft_fail_cause_removal_of_functioning_graft<br>graft_fail_cause_other<br>graft_fail_cause_unknown |
| Graft loss | graft_loss_yes<br>graft_loss_no |
| Early graft loss | early_graft_loss_graftloss_more_90_days<br>early_graft_loss_graftloss_less_90<br>early_graft_loss_no_graftloss<br>early_graft_loss_graftloss |
| Recipients death cause | r_death_cause_alive<br>r_death_cause_unknown<br>r_death_cause_pulmonary<br>r_death_cause_cardiovascular<br>r_death_cause_cerebrovascular<br>r_death_cause_gi_liver<br>r_death_cause_renal_dialysis_related<br>r_death_cause_other |
| Recipients death | r_death_yes<br>r_death_no |

Table 2: Dummy variables created from the transplantation dataset.