University of Bergen

Department of Information Science and Media Studies

Web-based Data Mining Tool for Total Knee Arthroplasty

Candidate
Sølve Ånneland

Supervisor
Professor Ankica Babic

June 15, 2021

1 Acknowledgements

I have received fantastic support and help from several people during the project. Firstly, I want to thank my supervisor Professor Ankica Babic for her great guidance. She has always been available to answer questions and provide valuable feedback.

I would thank my fellow master students for a great collaboration. It has been a pleasure working with all of you.

In addition, I will also give a special thanks to Dr. Peter Ellison for his help and feedback. Throughout the year, Peter has been available to answer any questions concerning the clinical aspects of arthroplasty.

I want to also thank the Norwegian Arthroplasty Register for meeting us, and providing insight into their work and organisation.

I would also like to thank everyone willing to set aside their time to provide feedback on the prototype. Your insight and contribution helped.

Lastly, I would thank my beloved family for always supporting and believing in me.

2 Abstract

Every year, thousands of Norwegian citizens undergo knee surgery in Norway. Taking a knee operation is not without risk. From previous studies, some products have shown worse quality than others. A lower lifespan of prostheses has consequences for patients and for society (pain, missing days of work, additional costs of re-operations and new prostheses).

The Norwegian Arthroplasty Register (NAR) collects data for 95% of the knee and hip surgeries in Norway, which for the knee is 6,000 surgeries each year. The purpose of the NAR register is to monitor and detect inferior prostheses as quickly as possible.

The typical way for detecting inferior prostheses is to use survival analysis whose advantage is that it uses time as a factor. For instance, the Kaplan Meier method can measure if one prosthesis has a consistently higher survival time over eight years as compared to another prosthesis. There are additional methods applicable such as Cox PH regression, which also can take into consideration other factors such as gender and age. These methods are preferred by clinicians.

In this thesis, we have developed a Web API that makes it possible to perform various methods of data analysis. An additionally implemented feature was the minimal front-end interface to graphically present results. Both are the building blocks of a functional prototype for performing various analyses including among others Kaplan Meier and Cox PH.

The advantage of the functional prototype is that it can be used online at the convenience of users. And, with minimal modification it could be directly connected to the NAR registry's database. Such a setup enables calculating outcomes frequently on regularly updated data.

The prototype was evaluated by biomedical experts as well as IT experts. Respective System Usability Scores (SUS) were calculated for each expert group, resulting in an average score of 86, which is excellent. The evaluators also suggested adding additional functionality like PCA analysis and a summary statistics table.

Contents

List of Figures

List of Tables

3 Introduction

This master thesis is part of a collaboration with three other master students Solheim, 2021; Stolt-Nielsen, 2021; T.Hufthammer, 2021. The purpose of the collaboration was to produce a data exploration and data mining tool for the arthroplasty domain. We wanted to explore and implement methods that could aid in assessing the survival outcome of prostheses in the arthroplasty domain.

Two students focused on the front-end development - Solheim, 2021 focused on visualisations and Stolt-Nielsen, 2021 on providing an appealing User Interface (UI) experience. The back-end team focused on data mining and is split between two focus areas - is Total Hip Arthroplasty (THA) and Total Knee Arthroplasty (TKA). The focus of this thesis have been on TKA.

The data is coming from the Norwegian Arthroplasty Registry (NAR) that is relying mainly on Kaplan Meier and Cox PH methods for this purpose. Our goal was to explore possibilities for applying data mining methods in a user friendly manner. We belived this would allow a greater flexibility for calculating patient outcomes. This work was academic, but inspired by the real challenges on one side, and the available technologies to solve it on the other side.

The first objective was to develop a back-end web-based Application Programming Interface (Web API) (*What is an API?* 2021) and minimal front-end in Flask which used the Web API to give a better graphical representation of the analyses and results.

The purpose of the Web API is to serve as a platform for data mining methods for detecting underperforming total knee prostheses.

This kind of research is on the rise, as can be seen with the Danish National Registry, which published results of data mining. They have in El-Galaly et al. (2020) shown applications of predictive modeling resulting in a better understanding of which features (variables) within TKA data may be suitable for building predictive machine learning models.

In detail, El-Galaly et al. (2020) used among 25 000 patient data records from 2012 to 2015 of primary TKA surgeries and analyzed them using data mining methods (El-Galaly et al., 2020, pp. 1). The research aimed to develop a model capable of predicting early TKA revision (El-Galaly et al., 2020, pp. 2). Early TKA revision happens when a patient needs a new prosthesis implant within two years after the first surgery due to problems with prostheses, which is considered a poor patient outcome (El-Galaly et al., 2020, pp. 1).

The prediction models were created using the four machine learning methods: logistic regression, random forest, gradient boosting, and neural network (El-Galaly et al., 2020, pp. 4). The ten most important features for each of the different prediction models were listed. Those features varied, but all the models considered feature age to be an important predictor of outcome (revision surgery) (El-Galaly et al., 2020, pp. 8).

For a prediction model to be clinically useful, the Area Under the Curve (AUC) score should be 0.7 or higher. The results of the prediction models were that all had an AUC score that ranged between 0.57 and 0.60. As such, none of the prediction models reached the AUC threshold goal of 0.7, and the models were, therefore, deemed clinically not useful (El-Galaly et al., 2020, pp. 12). Finally, the study mentions that the prediction models might be better if more data were available, like if the data contained an anonymous surgeon identifier (El-Galaly et al., 2020, pp. 13).

4 Research questions

The following research questions were formulated for this thesis:

- Is it possible to develop a minimal web-based functional prototype that utilizes data-mining to assist medical professionals making decisions regarding total knee arthroplasty?
- Which data-mining algorithms have the highest potential for predicting underperforming total knee prostheses?
- Is survival analysis, as the gold standard, the best predictor of under-performing total knee prostheses?

4.1 Overview of the thesis

This Chapter contains the outline of the research conducted within the master thesis project:

Chapter 5: Background contains the medical theory about knee arthroplasty prostheses, data registries, and research challenges. Chapter 5.2 covers the literature and related work, which is relevant for this project.

Chapter 6: Method explains the method used in this project and their contribution.

Chapter 7: Technologies explains the technologies used for creating a data-mining tool for knee arthroplasty.

Chapter 8: Data explains the test knee data set that were used to generate results in the data-mining tool.

Chapter 9: Identifying user needs and establishing requirements explains the requirements and how they were identified.

Chapter 10: Development iterations explains the four development iterations and different findings about development.

Chapter 11: Artifact explains the data-mining tool which consists of a Web API that runs data analysis and a Web based front-end that displays the results.

Chapter 12: Evaluation explains the evaluation of the data-mining tool.

Chapter 13: Discussion answers the research questions and discusses the limitations of the study.

Chapter 14: Conclusion and future work summarizes the project and provides recommendations for future work.

5 Background

5.1 Arthroplasty

5.1.1 Knee arthroplasty

Arthroplasty surgeries are conducted on different parts of the body, such as the hip, knee, shoulder, elbow, or ankle (Healio, 2012). In this thesis, the focus will be on total knee arthroplasty. This section starts with general information about knee arthroplasty and ends with more details about total knee replacement surgery.

”Arthroplasty is a surgical procedure performed by an orthopedic surgeon that alters or completely replaces a joint in the body, usually to restore normal motion and relieve pain in a malformed or diseased joint” (Healio, 2012).

People usually get knee replacement surgery because of pain caused by osteoarthritis (*Knee replacement* 2017). Osteoarthritis is a degenerative joint disease that 30 million suffer from in the USA. It occurs most often in hands, feet, spine, hips, and knees. Osteoarthritis occurs typically due to the degeneration of cartilage. Cartilage is a rubbery tissue at the end of bones that helps bones to connect to each other so that they can move more smoothly (Holland, 2018).

Knee replacement surgery can fix pain and functionality of diseased knee joints. To figure out if a knee replacement surgery is necessary, the doctor checks the knee’s range of motion, stability, and strength and uses x-rays. Factors the doctors can take into consideration can be age, weight, health, activity level, knee size, and shape. The risks associated with this type of surgery are infection, blood clots, stroke, heart attack, and nerve damage (*Knee replacement* 2017).

There are advantages of surgical therapy, but the drawbacks of getting a knee replacement is that a knee replacement can wear out sooner if the user is doing high-impact activities. Therefore, it is recommended to avoid high impact activities such as jogging and sports that involve jumping and instead do low impact activities such as walking or swimming. Another type of problem that can occur are infections in the knee replacement. If that happens, the patient will need to get a new knee replacement surgery. The lifespan of a knee replacement varies, but it is usually over 15 years (*Knee replacement* 2017). Similar results were also found by Evans et al. (2019), who investigated how long a knee replacement lasted. The findings were that for primary total knee replacement, 82% lasted 25 years, and for unicondylar knee replacement, 72% lasted 25 years (Evans et al., 2019, pp. 655). As with many health issues, older individuals are also more prone to a need for a knee replacement than younger individuals (*Knee replacement* 2017).

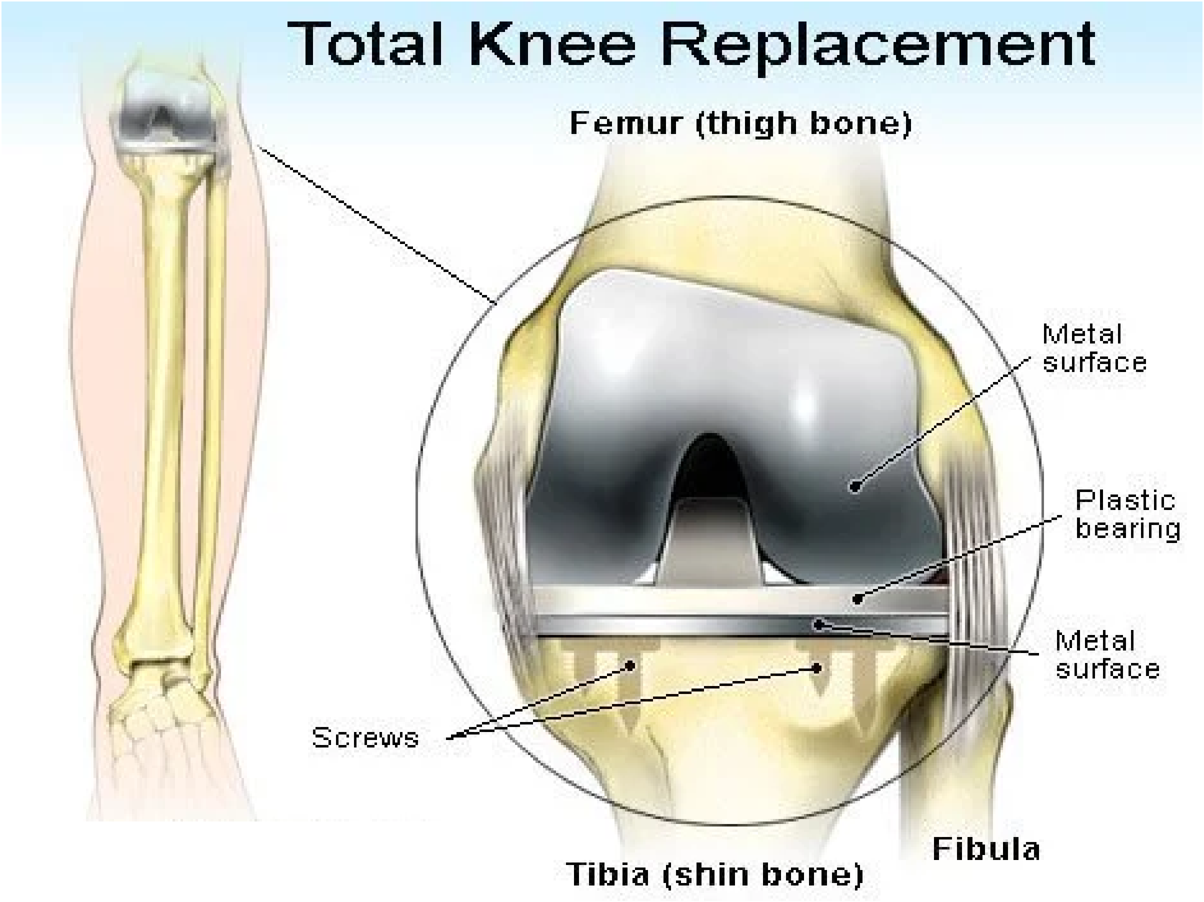


Figure 1: Total Knee Replacement (Shiel, 2019)

Total knee replacement is a surgical procedure that usually takes under 2 hours (*Total Knee Replacement* n.d.). The first thing the surgeon does in the surgery is to remove damaged cartilage and bones. Afterward, the surgeon places implants into the knee. These implants can be made of metal or plastics (*Total Knee Replacement* n.d.).

Different parts of the knee are replaced in total knee replacement surgery. The parts that are always replaced with artificial components are the femur bone and the lower leg bone (tibia). A part of the knee that is only replaced if the condition is bad enough is the kneecap portion of the knee joint. All these artificial components combined are called a prosthesis (Shiel, 2019).

In total knee replacement surgery, the surgeon can use cemented, uncemented, or hybrid (cemented tibia and cementless femur) between the bone and the implants. However, there is no conclusion of which type works best (Brown, Harper, and Bjorgul, 2013).

5.1.2 Arthroplasty registries

Arthroplasty registries exist to improve the treatment of patients that are undergoing joint replacement surgery. Arthroplasty registries are common in the Nordics, and they exist in Denmark, Sweden, Finland, and Norway. (Pedersen and Fenstad, 2015, p.5).

However, the countries differ in statistical methods, data being collected, implant survival, etc. Because of these differences, results between the countries are not completely comparable (Pedersen and Fenstad, 2015, p5).

The Nordic Arthroplasty Register Association (NARA) was created in 2007 To improve the quality of research and understanding of outcomes in joint replacement surgery (Pedersen and Fenstad, 2015, p5). The NARA association members are arthroplasty registers from Denmark, Finland, Norway (The Norwegian Arthroplasty Register), and Sweden (Pedersen and Fenstad, 2015, p6-7).

One of the NARA association‘s goals is to create one minimal Nordic dataset that contains results for total joint replacement surgery. This dataset should make it possible to study smaller subgroups of patients that are too small to investigate only in a single country (Pedersen and Fenstad, 2015, p6). The current NARA association dataset contains 20 variables for knee data that are collected from all member countries from the year 1997 (Pedersen and Fenstad, 2015, p8). The NARA knee dataset contains 390 000 primary knee arthroplasty surgeries from 1997-2012, where most surgeries are from Sweden (150 000) and only 49 000 or 12.5% are from Norway (Pedersen and Fenstad, 2015, p9).

The Norwegian arthroplasty registry was founded in the 1980s and was from 2009 a nationwide register in Norway (*norwegian arthroplasty register* n.d.). Every year the register gathers over 95% of the knee and hip arthroplasties conducted in Norway. For the knee arthroplasties, that is a total of 6000 each year. The register gathers information from when the first time the prosthesis is inserted and also for later revisions for failed prostheses. For the patient to be included in the register, the patient must write a written consent (*norwegian arthroplasty register* n.d.). From 1994 to 2018, 97022 knee replacements have been registered in the Norwegian Arthroplasty Register (NAR). The most common reasons for reoperations of knee arthroplasty are infection, instability, loose tibial component, and pain (*Norwegian national advisory unit on arthroplasty and hip fractures: annual report* 2019, pp. 65). Reoperations can be caused by poor surgery outcomes (*Norwegian national advisory unit on arthroplasty and hip fractures: annual report* 2019, pp. 65) . Nevertheless, for total knee arthroplasty survival curves have been improved from 1994 to 2018 (*Norwegian national advisory unit on arthroplasty and hip fractures: annual report* 2019, pp. 65) . In other words, the patient's prosthesis lasts longer today than earlier.

In this thesis, I got access to an experimental dataset from the Norwegian Arthroplasty Registry containing knee arthroplasty surgery data.

Lastly, the primary purpose for the Norwegian Arthroplasty Register (NAR) registry is:

"The main purpose of the register is to function as a surveillance tool to identify inferior implants as early as possible" (*The Norwegian Arthroplasty Register* 2021).

The NAR registry uses the survival analysis methods Kaplan Meier and Cox PH, which is considered the standard to identify inferior prostheses. In the next section, inferior prostheses are identified using these methods.

5.1.3 Baseline prosthesis

Gjøtesen (2013) used survival analysis to investigate the rates and cause of revision of seven different brands of cemented primary Total Knee Replacement (TKR). The data in the study was NAR register data from 1994 to 2009. For comparing different prostheses, there was a need for a benchmark or a so-called baseline prosthesis. The Profix knee prostheses, the most commonly used total knee prostheses in Norway, were chosen as the baseline prosthesis. And the rest of the prostheses results were compared relative to this prosthesis (Gjøtesen, 2013, pp. 636).

For detecting inferior prostheses, there is a need for a so-called standard / baseline that can indicate if a prosthesis is inferior or not. For instance, if the baseline prosthesis has a mean survival year of 15 years, and another prosthesis has a mean survival year of 12 years, this other prosthesis performs below the baseline. It is therefore, clearly an inferior prosthesis. The previous example is only valid if the poorly performing prosthesis has a proper sample size. If the sample is too small, there would be no way to draw a reliable conclusion based on this example. For instance, if there only exists 20 patients for the poor prosthesis while the data for the baseline is 2000 prosthesis, I would argue that the small sample size of 20 is too limited to draw a proper conclusion. However, even though the sample size is small, there might be a good idea to investigate why the prosthesis is performing this badly. It is in the patients interest to address the safety concerns. If the prosthesis is new, there might be a learning phase for the surgeons (Peltola, Malmivaara, and Paavola, 2013). Hence the results might improve over time. There also might be certain types of more active patients who have had this prosthesis so it is their lifestyle that led to a lower survival rate than normal.

The performance of a prosthesis also depends on the surgeon's skill level (Badawy et al., 2017, pp. 1) and the patients' health (Boyce et al., 2019, pp. 557) and activity (Seyler et al., 2006). Because of all of these different factors, there is a need for a sample size that is large enough to eliminate the randomness of surgeon skill and patient health and lifestyle.

Our investigation also needs a baseline prosthesis that will be used as a benchmark for detecting inferior prostheses. As Gjøtesen (2013) suggested, Profix will be used as the baseline prosthesis in this paper for survival analysis.

5.2 Literature Review

When conducting the literature review, I only included articles within the field of healthcare that were created after the year 2000.

The literature review is primarily oriented around literature in healthcare that has used data mining. During the review, I examined the workings of previous students that have collaborated with the NAR registry. Additionally, I was studying research that has used data mining algorithms in the field of knee arthroplasty to find algorithms with good potential. However, I also included research from other healthcare fields to get a broader knowledge of data mining algorithms' potential.

The literature review starts with previous master theses of students who collaborated with the NAR register. Next, I cover relevant research employing data mining to arthroplasty and healthcare data. And finally I addressed literature from other healthcare fields.

5.2.1 Similar research

This section includes previous master theses that dealt with different aspects of the NAR registry.

Iden (2020) investigated how data mining could be used to gather insight into total knee arthroplasty using NAR registry data. For this purpose, he used various machine learning and statistical methods (Iden, 2020, pp. 49). The approach he used was Knowledge discovery in databases (KDD), which is a process of discovering patterns in data. The KDD method consist of the different steps, such as data selection, preprocessing, transformation, data mining and interpretation/evaluation (Iden, 2020, pp. 10).

For his research, he explored the variables young age, bilateral surgery, perioperative surgery, implant product that lies in the NAR registry dataset. Additionally, he made predictions about the cause and type of revision surgery and the time difference between first and second bilateral surgery (Iden, 2020, pp. 11).

The NAR registry data used for his research contained among 40 000 patient records from 1994 to 2018. The NAR dataset consisted of only popular devices with over 500 records, and most of the data was of primary surgeries and a limited amount of revision surgeries (Iden, 2020, pp. 13).

The technologies used for preprocessing were Pandas (*Pandas* 2021) and Numpy (*Numpy* 2021). For machine learning and transformation of data, he used Scikit learn (scikit, 2021a) (Iden, 2020, pp. 13). For normalizing the data, he used Scikits MinMaxScaler (Iden, 2020, pp. 13).

For exploring patterns in the data, he used Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), and k-means cluster analysis (Iden, 2020, pp. 16-17). Additionally, he explored the data using descriptive statistical techniques such as mean values and displayed value distribution in boxplots (Iden, 2020, pp. 18-19).

The findings from LDA were mixed. It performed poorly for clustering the time difference between bilateral surgeries and ASA classifications. However, for clustering deep infection, the performance was good (Iden, 2020, pp. 38-39).

The findings from clustering using PCA were that the results were more stable, as it created explanatory clusters across the applied variables. Interestingly, he managed to see from PCA that LCS Complete had a positive correlation to serious ASA classifications. However, the best result was found in the time difference between the bilateral surgeries (Iden, 2020, pp. 39).

The findings from k-means clustering were that it produced mixed results. Often it produced unevenly sized clusters, but some cluster results were of interest (Iden, 2020, pp. 39). Interestingly, it is possible to see that the LCS complete device has a connection to reduced survival rate (Iden, 2020, pp. 44). Another finding was that the best k-means algorithm was k-means with cosine distance (Iden, 2020, pp. 40).

For classification and regression, he followed the same approach. The approach consists of first making a feature selection with the help of Scikits SelectKBest method. Second, the algorithms was evaluated using 10-fold cross-validation that splits data into pairs, i.e training and test sets. Third, visualize and compare the results of algorithms in boxplot graphs. Lastly, the best algorithm was applied for hyperparameter optimization using Scikit GridSearchCV (Iden, 2020, pp. 19-20). He tested 9 different classification algorithms and 6 regression algorithms (Iden, 2020, pp.19-20).

For classification of revision cause, gradient boosting classification gave the best result with 46.6 percent accuracy (Iden, 2020, pp. 46). However, a problem with the classifier is that rare revision causes like bone fracture are never predicted correctly (Iden, 2020, pp. 47). To improve the classifier's performance, loosening of distal and loosening of the proximal component was combined into the value general aseptic loosening. After these values were combined, the results improved, and there were fewer failed predictions (Iden, 2020, pp. 47).

For classification of revision type, several algorithms performed similarly. However, gradient boosting was the best algorithm (Iden, 2020, pp. 49). The classification results using gradient boosting were that it had a macro average of 41 percent for precision and an F1 macro average of 26 percent (Iden, 2020, pp. 51).

For predicting the time difference between bilateral surgeries, the goal was to predict time divided into two and four time frames (Iden, 2020, pp. 51). The gradient boosting classifier gave the best result for predicting two time frames with a median accuracy of 77 percent and four time frames with a median accuracy of 56 percent (Iden, 2020, pp. 53).

For predicting the exact time difference between bilateral surgeries, no cases where time was above 4000 days were predicted correctly (Iden, 2020, pp. 56). The best algorithm for predicting time difference was the k-nearest neighbors regressor that had an adjusted r-squared between 0.48 and 0.55 during 10-fold testing (Iden, 2020, pp. 54-55).

Additionally, he also conducted an online recorded interview with experts that consisted of a biostatistician, arthroplasty surgeon, and prosthesis engineers. In the interview, he asked them how they consider the results from the descriptive and the predictive modeling. He also asked how the work can be developed further and their thoughts on the KDD method in knee arthroplasty (Iden, 2020, pp.24). The participants in the interview thought the results were of high interest. However, it was mentioned in the interview that there is a need for close collaborations between data scientists and physicians in a way that physicians gain a bigger understanding of the models and the data scientists gain a larger clinical understanding (Iden, 2020, pp.57).

Ertkjern (2015) produced a prototypical artefact with emphasis on HCI as part of his master thesis. He created a web application that includes a search engine and functionality for surgeons to submit records (Ertkjern, 2015, pp. 1-2). The intended user groups for this artefact are researchers, surgeons, and public users (Ertkjern, 2015, p. 84).

The artefacts search engine was developed to promote Open Data and to use Big Data in a useful way in health research to detect prostheses with higher risk of failure. The search engine pulled data from different sources that are relevant for the different user groups. One of those data sources is The Norwegian Arthroplasty Register that provides searchable records of hip and knee implants. However, since researchers at Haukeland University Hospital also use other data sources for research, additional sources were included, such as MAUDE, Clinical Trials website, and PubMed. Besides search functionalities, the artefact provided surgeons the opportunity to submit and monitor records and view statistics, which should encourage research contributions by surgeons (Ertkjern, 2015, pp. 1-2).

Regarding the representation of data, the author highlights the importance of visualizing data in a pleasing way. For example, by using map, bar and pie charts rather than plain data tables (Ertkjern, 2015, pp. 12-14). The evaluation was done using heuristic evaluation by expert users and using the system usability scale method on end-users. Results indicated satisfactory performance (Ertkjern, 2015, p. 88).

5.2.2 Data mining in arthroplasty

The first paper is investigating reasons for failures of revision knee arthroplasty.

Leta et al. (2015) investigated reasons for failures of revision knee arthroplasty. To gather more knowledge on the subject, they investigated survival rate and modes of failures of revision knee arthroplasty. They examined data collected between 1994 and 2011 from the Norwegian Arthroplasty Register. For analysis, Kaplan-Meier and Cox regression were used. Results from the study found infection, instability, loose tibial component, and pain to be the most common reasons for re-revision (Leta et al., 2015, pp. 48). In terms of survival rate for revision knee arthroplasties, it was 85 percent for five years and 71 percent for 15 years (Leta et al., 2015, pp. 48).

Some factors, such as sex and age, increased the likelihood of re-revision. For instances, more males than females underwent re-revision. Also, a larger portion of patients under 60 years of age as compared to those over 60 had re-revision (Leta et al., 2015, pp. 48). For patients under 60, authors suggest higher activity levels might be the cause. One other likely reason is that surgeons might have an easier time to ask younger patients for re-revision than older ones. One other factor that increases the risk of re-revision is sex, more specifically, male. Males had twice the prospect of re-revision than females (Leta et al., 2015, pp. 54).

The study had some strengths and weaknesses. An advantage of the study was that it used a nationwide registry with high external validity. A deficiency of the study was that factors such as surgeon skills and lifestyle and additional health issues were not taken into consideration.

Lewis et al. (2013) investigated how to predict 10-year knee replacement. The study used patient data from Australia that included over 1400 women over 70-year-old. The study found that certain variables led to increased risk. These variables were body mass index, knee pain, previous knee replacement, analgesia use for joint pain, and age (Lewis et al., 2013, pp. 1). The model also showed that the inclusion of body mass index into the prediction model improved the prediction significantly (Lewis et al., 2013, pp. 6).

To analyze the patient data, the researchers used IBM SPSS, STATA, and SAS (Lewis et al., 2013, pp. 3). In the prediction model, the variables joint pain, age, body mass index, previous knee replacement, and analgesia use for joint pain were selected. The results showed a C-statistics score of 0.79 for predicting 10-years knee replacement (Lewis et al., 2013, pp. 5).

The paper did not precisely mention which type of prediction method they used in the study. Still, it is highly likely that the paper used a type of regression analysis because of the inclusion of regression coefficients in the table on page 4 (Lewis et al., 2013, pp. 4). Even though the study did not mention the kind of prediction method, it gave valuable indications by identifying variables that might have predictive power for knee arthroplasty.

The limitation of the study is that it only included elderly women. Because of that, the same prediction model and outcome might not necessarily apply to other parts of the population.

5.2.3 Data mining in healthcare

Lorenzoni et al. (2019) compared different machine learning techniques to predict hospitalization in heart failure patients from Italy. To make the prediction, they used data from an ongoing project in Italy named the Gestione Integrata dello Scompenso Cardiaco (GISC) study. The GISC project is a heart failure assistance program where the heart failure patients are enrolled. The GISC project uses an online platform that also shares data with healthcare staff (Lorenzoni et al., 2019, pp. 1). The dataset in the GISC study contained data from 380 patients that were included in the program between 2011-2015. In the dataset, only 110 patient records had no missing data (Lorenzoni et al., 2019, pp. 3).

Before running the algorithms on the dataset, they had to figure out which kind of data to include as predictors. Consequently, they included hospitalization, numerical variables such as Body Mass Index (BMI), age, heart rate, etc. The data that were considered for categorical variables were gender, the occurrence of myocardial infarction, etc. After selecting variables, all the predictors in the dataset were transformed into numerical variables, and continuous variables were transformed into values between -1 and 1 (Lorenzoni et al., 2019, pp. 3).

For prediction, the following algorithms were tested and compared: Logistic regression, Generalized Linear Model Net (GLMN), classification and regression tree, random forest, adaboost, logitboost, support vector machine, and neural networks (Lorenzoni et al., 2019, pp. 1). The study found that the best prediction method was GLMN. It had an average accuracy of 81.2%, a positive predictive value of 87.5% and a negative predictive value of 75% (Lorenzoni et al., 2019, pp. 9). As for prediction, the best performance was achieved when patients with one or more missing values were removed from the dataset (Lorenzoni et al., 2019, pp. 6).

A drawback of the study is that it contains a very limited dataset. Nevertheless, it does give some advice on how data cleaning can improve performance and which type of algorithms could be useful for prediction.

Endo, Shibata, and Tanaka (2007) tried to predict the five-year survival rate of breast cancer using different algorithms. The dataset that they used contained over 87000 records from the US. The dataset should, according to the paper, be representative of the whole US population. The paper used over 37000 records from the dataset. In the preprocessing part, male patients were removed, such that the dataset contained only female patients (Endo, Shibata, and Tanaka, 2007, pp. 11-12). The records contained patients diagnosed with breast cancer (Endo, Shibata, and Tanaka, 2007, pp. 11). The dataset contained variables such as race, marital status, age, surgery performed, the reason for no surgery, radiation, etc. Patients with survival over five years were assigned the number 1, while the rest were assigned number 0 (Endo, Shibata, and Tanaka, 2007, pp. 12).

For the prediction of breast cancer survival, seven different algorithms were tested. Those were artificial neural network, naive bayes, bayes net, decision trees with naive bayes, decision trees, decision trees, and logistic regression model (Endo, Shibata, and Tanaka, 2007, pp. 12). The metrics used for measuring how the algorithms performed were sensitivity, specificity, and accuracy. Specificity means the probability of predicting death, sensitivity is the probability of predicting survival, and accuracy is the probability of predicting both survival and death. The metrics measured survival and death within a time span of 5 years, that is, if patients are dead or alive within five years (Endo, Shibata, and Tanaka, 2007, pp. 14). The result when comparing the algorithms, were that logistic regression had the highest accuracy (85.8%), decision trees had the highest sensitivity (97.1%), and the artificial neural network had the highest specificity(50.9%) (Endo, Shibata, and Tanaka, 2007, pp. 14-15). In the discussion of the paper, the author suggests that the best algorithm for predicting breast cancer survival might be a combination of decision trees and logistic regression (Endo, Shibata, and Tanaka, 2007, pp. 16).

An advantage of the paper is that it uses a large dataset for comparing different algorithms in the medical domain.

Voznuka et al. (2004) Assistme is a decision support system with the purpose to support thoracic surgeons (Voznuka et al., 2004, pp. 497). The system uses data from previous surgeries where most patients have implanted a mechanical device like for instance Left Ventricular Assist Device(LVAD), which makes the heart recover from ventricular failure (Voznuka et al., 2004, pp. 497-498).

The Assistme system is able to generate predefined reports about information about mean values of age, weight and the mortality of different devices, etc. (Voznuka et al., 2004, pp. 503). Additionally the system can generate reports using Case Based Reasoning (CBR) and Cluster analysis (Voznuka et al., 2004, pp. 498). For visualizing the reports, bar charts and tables were used since they were preferred by physicians (Voznuka et al., 2004, pp. 508).

The potential user groups for the system is surgeons or physicians, administrative, and patients. These groups were identified in meetings with physicians. Each user group has different needs for reports. The more advanced users like surgeons or physicians need to be able to generate customised reports by choosing their own parameters. Less knowledgeable groups like patient users need more simple reports like predefined reports (Voznuka et al., 2004, pp. 499).

The advantage of the Assistme system is that it can generate reports and summary of registers with updated information. Compared to today‘s systems this could be seen as an advantage as it usually takes several months before reports are generated (Voznuka et al., 2004, pp. 508).

5.3 Research Challenges

When doing research within the medical field, certain research challenges may arise. One of those challenges concerns surgery practices, which may differ between countries.

Due to differences in legislation and practices across nations, research from one country may not be applicable in others. For instance, knee arthroplasty practices differ between Norway, Sweden, and Denmark. Robertsson et al. (2010) compared the differences between knee arthroplasty in Sweden, Norway, and Denmark using the Norwegian Arthroplasty Register, the Swedish Knee Arthroplasty Register and the Danish Knee Arthroplasty Register. They looked into the demographics, methods, and overall results. The study found that there exist differences between the countries across different factors. One of the factors that differ was in the use of uncemented or hybrid fixation components. This class of prostheses was used 14% in Norway, 2% in Sweden, and 22% in Denmark (Robertsson et al., 2010, pp. 82). According to the study, there were also more reoperations after total knee arthroplasty for osteoarthritis in Norway and Denmark compared to Sweden (Robertsson et al., 2010, pp. 82). This may be due to the higher use of uncemented or hybrid prostheses in Norway and Denmark compared to Sweden. Additionally, the implant brands also differed a lot between the countries. In the conclusion of the study, the authors state that Sweden has the lowest revision rate and that further research is needed (Robertsson et al., 2010, pp. 82).

As mentioned above, different practices exist in both equipment brands and surgery practices between countries. Therefore, comparing studies directly can be challenging. In addition to that, implants have likely improved in quality, and the same goes for surgical procedures and equipment as well. Consequently, comparing past studies to more recent ones can be challenging. Older studies may use one type of equipment, while current studies are using newer and more advanced equipment. Due to technological advancements, previous studies may not be of the same relevance as more recent studies.

Changes in surgery practices and equipment brands are likely to influence historical datasets like those from the Norwegian Arthroplasty Register (NAR). As such, even if the proposed artifact can make high predictions applied on the NAR dataset, these predictions will be influenced by historical changes in the equipment and implants. These historical changes that lie in the NAR dataset are a limitation of this thesis research, and it is essential to be aware of it.

6 Method

6.1 Design Science

Much research in academia is never being used or known in business organizations. One of the reasons for this is that much research is not relevant to professionals in organizations (Dresch, Lacerda, and Jose, 2015, pp. 1). A type of knowledge that can increase relevance is Mode 2 knowledge. Mode 2 knowledge aims to solve problems that occur related to an application, and this is also interdisciplinary. Mode 1 knowledge is academic and occurs in one discipline (Dresch, Lacerda, and Jose, 2015, pp. 3). Mode 2 knowledge is similar to the goals of design science, which is: "The mission of developing knowledge that can be used by professionals to solve their day-to-day problems"(Dresch, Lacerda, and Jose, 2015, pp. 3).

Mode 2 knowledge uses different people among industries that work together to create the best possible solution where each person contributes different expertise to the project (Dresch, Lacerda, and Jose, 2015, pp. 63).

Hefner et al. (2004) proposed seven guidelines for Design Science research. These guidelines are design as an artifact, problem relevance, design evaluation ,research contributions, research rigor, design science as a search process and communication of the research (Hefner et al., 2004, pp. 86). Listed in Table 1 is an explanation of how these guidelines relate to this thesis.

Guideline	**Description**
1. Design as an artifact	The artifact is a web-based API that generates knowledge for knee arthroplasty using data mining.
2. Problem relevance	The goal of the artifact is guided by professionals in the healthcare industry.
3. Design evaluation	Feedback from the primary users of the artifact is needed to assure that the design is relevant for them.
4. Research contributions	The research contribution is the artifact itself and how well it performs.
5. Research rigor	The artifact has been developed using the DSDM methodology, which is a method that is suitable for small teams. In addition to that, several other methods were used to develop the artifact.
6. Design science as a search process	Several prototype iterations have been made before completing the artifact.
7. Communication of the research	The master thesis has been written in a way that both management and technology professionals will understand it. Also it will be available at the Open Source portal of the university.

Table 1: Design Science guidelines from (Hefner et al., 2004, pp. 86), with an explanation of how they relate to this thesis.

6.2 Knowledge Discovery in Databases (KDD)

The directions of turning data into useful knowledge is described by the KDD process. A high level overview of the KDD process is found in Figure 2.

”KDD is the organized process of identifying valid, novel, useful, and understandable patterns from large and complex data sets.”(Maimon and Rokach, 2005, pp. 1)

The core step in the KDD process concerns data mining, which concerns the use of algorithms to find and explore patterns in data (Maimon and Rokach, 2005, pp. 1).

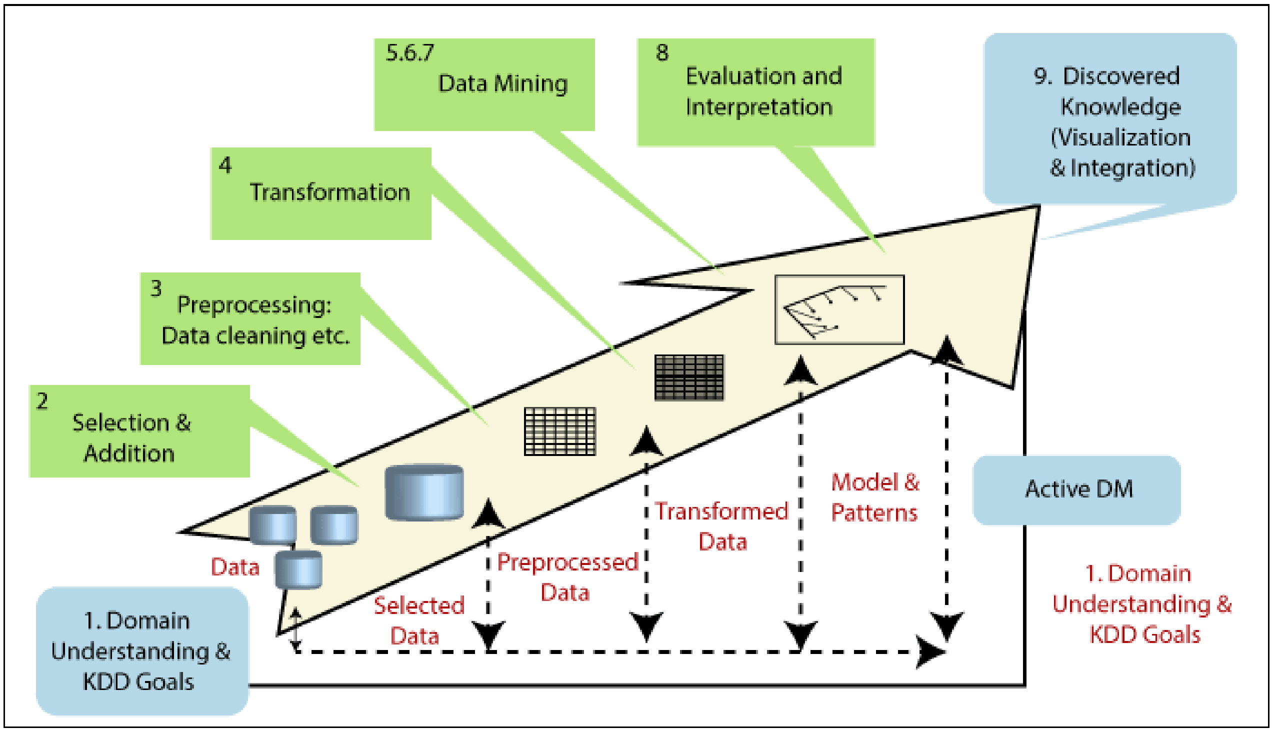

5.6.7
Data Mining
8
Evaluation and Interpretation
9. Discovered Knowledge (Visualization & Integration)
4
Transformation
3
Preprocessing: Data cleaning etc.
2
Selection & Addition
Active DM
Model & Patterns
Transformed Data
Preprocessed Data
Selected Data
Data
1. Domain Understanding & KDD Goals
1. Domain Understanding & KDD Goals

Figure 2: The Process of Knowledge Discovery in Databases. The figure has been adopted from (*KDD Process in Data Mining* 2021)

The KDD process consists of nine steps that are listed in Figure 2 (Maimon and Rokach, 2005, pp. 2). The first step is developing an understanding of the domain and defining goals (Maimon and Rokach, 2005, pp. 3). Step 2 is about identifying the data that should be included in the dataset and integrate all the selected data into one data set. Step 3 is the cleaning of the dataset that consists of the handling of missing values, noise, and outliers. Missing data can be handled in different ways, where one way is to predict missing values. Step 4 is about preparing and generating good data for data mining using, for instance, feature extraction (Maimon and Rokach, 2005, pp. 4). Step 5 is the selection of the data mining algorithm based on the goal of the project. The algorithm can be, for instance, regression, classification, or clustering. Step 6 is about finding the exact data mining algorithm to use for the project. Step 7 is a step at which algorithms are running several times before finding acceptable results. This step can demand adjusting parameters in order to find optimal results. Step 8 is the evaluation part, where the mined patterns are interpreted and evaluated based on the goals(Maimon and Rokach, 2005, pp. 5). Step 9 is where the knowledge is used in another system (Maimon and Rokach, 2005, pp. 6).

The steps in the KDD process have been used as a guide in this research.

6.2.1 Domain understanding and KDD goals

I have been in multiple meetings with professionals working at the Norwegian Arthroplasty Register. In these meetings, I have learned that one of the most important tasks for the register is to detect underperforming prostheses. I have also learned from these meetings that Kaplan Meier analysis and Cox proportional hazard analysis are commonly used for detecting underperforming prostheses.

Besides having meetings with professionals, I have also viewed existing web applications that generate knowledge using arthroplasty registry data. One typical example of these web applications is the prototype from the Finnish Arthroplasty Register (*FAR Finnish Arthroplasty Register* 2021). This prototype has functionality that is limited to only basic statistics. The prototype we are building goes beyond basic statistical methods since it includes survival analysis methods and clustering.

6.2.2 Preprocessing

Before running algorithms on a dataset, it is useful to gain more knowledge about the current dataset.

One way to gain more insight is to run a correlation analysis. Correlation analysis is used to find relationships that exist in the data, and it determines the strength of the relationship between two item sets. A negative value indicates a negative relationship, and a positive value indicates a positive relationship. The higher the value, the stronger the relationship is. If a strong relationship exists, the independent variable can be a strong predictor of the dependent variable. A drawback of correlation analysis is that it does not guarantee causality (Kumar and Chong, 2018, p. 5).

6.2.3 Data mining

Data mining uses a broad range of various discovery methods for finding patterns in data. These methods include prediction and description methods (Maimon and Rokach, 2005, p. 6). Of prediction methods, supervised and unsupervised methods are the main groups. Unsupervised methods group instances without a dependent attribute. Supervised methods discover relationships between variables (Maimon and Rokach, 2005, p. 7). Among supervised methods, there are regression models that can predict values and classification models that group data into classes (Maimon and Rokach, 2005, p. 8). Another type of method class is verification methods. These are using statistics to evaluate hypotheses such as a t-test or multivariate variance analysis called ANOVA (Maimon and Rokach, 2005, p. 7). In the next section, machine learning is explained in more detail.

6.2.4 Machine learning

The project's development requires powerful algorithms that can use historical data for prediction or classification of new cases. Machine learning algorithms are powerful and capable of doing this. As compared to statistics they can run many times to meet user needs and to allow a broader exploration of data.

Machine learning (ML) is a part of artificial intelligence (AI). ML is currently being used to solve problems in vision, speech recognition, robotics, and many other fields. The way ML works is that it uses past experience (data) to make predictions in the future, gain knowledge from the data, or both. In practice, ML uses data that is called training data to build a model (Ayodele, 2010, p. 2). In simple terms, the ML algorithm gets input data, and the ML algorithm creates a program by itself by recognizing patterns in the data (Sodhi, Awasthi, and Sharma, 2019, p. 1357). To test the models accuracy, the model is tested on test data, which is data that has not been used for building the model (Sodhi, Awasthi, and Sharma, 2019, p. 1359).

Problems like classification, regression, similarity can be solved by ML. In classification, the ML algorithm puts objects in different categories. In regression, the ML algorithm predicts the probability of something (Sodhi, Awasthi, and Sharma, 2019, p. 1357). In similarity, the ML algorithm finds similar objects (Sodhi, Awasthi, and Sharma, 2019, p. 1358).

In supervised learning, the ML algorithm is provided with labeled input and output data for training. After the training, the algorithm has created a mapping function that can identify an output for a given input. The training process continues until the algorithm has reached an acceptable accuracy. An example of a supervised learning algorithm is linear regression (Sodhi, Awasthi, and Sharma, 2019, p. 1358).

In unsupervised learning, the ML algorithm gets unlabeled input data, and then identifies the hidden patterns in the dataset without former training. It is most often used in clustering and association problems. An example of this type of algorithm is k-means for clustering (Sodhi, Awasthi, and Sharma, 2019, p. 1358).

The next sections show different measures of how to evaluate the performance of machine learning algorithms.

6.2.5 Evaluating algorithmic performance

Different performance measures like sensitivity (1), specificity (2), precision (3) and accuracy (4), are used to evaluate the performance of algorithms. These metrics are important for comparing different algorithms and for evaluating the overall usefulness of algorithms. This section includes the performance measures that will be used in the final master thesis when conducting data mining on knee arthroplasty.

Sensitivity measures how well the algorithm recognizes positive samples (Maimon and Rokach, 2005, p. 666).

$$Sensitivity = \frac{true\_positive}{positive} \tag{1}$$

Specificity measures how well the algorithm understands negative samples (Maimon and Rokach, 2005, p. 666).

$$Specificity = \frac{true\_negative}{negative} \tag{2}$$

Precision measures how well the algorithm works on positive cases in calculating the percentage of positive cases that are, in reality, positive (Maimon and Rokach, 2005, p. 666).

$$Precision = \frac{true\_positive}{true\_positive + false\_positive} \tag{3}$$

Accuracy uses the combination of sensitivity and specificity to calculate the accuracy of the algorithm (Maimon and Rokach, 2005, p. 666).

$$Accuracy = sensitivity + specificity \tag{4}$$

The Receiver Operating Characteristic (ROC) Curve

The Receiver Operating Characteristic (ROC) Curve is being used to predict how well diseased patients are distinguished from healthy individuals. For this calculation, it uses sensitivity and '1 - specificity' (Hajian-Tilaki, 2013, pp. 627).

The ROC curve Y-axis shows the true positive fraction, calculated using TP/(TP+FN), also called sensitivity. And on its X-axis, it shows the false-negative fraction calculated using FN/TP+FN, which can also be calculated using '1 - specificity' (Hajian-Tilaki, 2013, pp. 628–630).

A ROC curve with a value of 0.5 means that the predictions are completely random. A score of 1 means that all predictions are correct and all diseased patients are successfully classified. Lastly, A score of 0 means that all predictions are wrong (Hajian-Tilaki, 2013, pp 631).

6.3 Survival Analysis

New treatments in healthcare need to be tested in trials to see if they are efficient. One way to analyze data from clinical trials is to use statistical models that use *the time until an event occurs.* Examples of such events are death, negative reaction to treatment, etc. (R. Singh and Mukhopadhyay, 2011).

The investigators using survival analysis do this to get the probability of surviving x amount of years, which can be one, two, or more. Additionally, they may use survival analysis to compare different groups' survival (Jager et al., 2008, pp. 561).

6.3.1 The Kaplan Meier method

The most popular method in survival analysis is the Kaplan Meier method (Jager et al., 2008, pp. 560). Since survival analysis is popular in medical research, many researchers have used and written about the Kaplan Meier method. A PubMed search for the term "Kaplan Meier" in April 2021 returned over 77000 hits (*kaplan+meier - Search Results - PubMed* 2021).

The Kaplan Meier method is usually presented as a plot using cumulative survival (Jager et al., 2008, pp. 563), as seen in Figures 3 and 4, which are actually Kaplan Meier plots. However, some authors prefer to show Kaplan Meier plots using cumulative mortality (Jager et al., 2008, pp. 564). Cumulative mortality is calculated using 1 - cumulative survival as seen in Table 2.

When the Kaplan-Meier method is used together with the log-rank test, it can give survival probabilities and be used to compare survival between groups (Jager et al., 2008, pp. 565). Cumulative survival, the log-rank test, and P-values are explained below.

6.3.2 Cumulative survival

One way to compare survival between different groups is to use cumulative survival at some particular time (Jager et al., 2008, pp. 563). The cumulative survival is the proportion surviving on this day multiplied by the cumulative survival over the previous period (Jager et al., 2008, pp. 562).

An illustrative example with sample data over cumulative survival is presented in Table 2. The column "Year" contains the values 0 to 6. Year = 0 is at the start of the study, and year=6 is at the end of the study. The column "Number at risk" is how many people are still in the study. The column "Prosthesis failure" contains how many prostheses failed. The "cumulative survival" column contains how the cumulative survival data is calculated.

In the start of the study 50 patients were included. After one year, one prosthesis failed . Therefore the proportion surviving that year was 49/50 = 0.98, and the cumulative survival was 1 * 0.98 = 0.98. For year two, one more patient also had a failed prosthesis. In year two, the proportion surviving the year was 48/49 = 0.9796, and the cumulative survival was 0.98*0.9796 = 0.96. Lastly, cumulative mortality is the probability of not surviving the year. Therefore the calculation for this is 1 - cumulative survival. In year two, cumulative mortality is therefore 1-0.96 = 0.04.

A plot over the cumulative survival is shown in Figure 3. The X-axis is the number of years, and the Y-axis is the cumulative survival percentage. From the plot, it is easy to see that the drop in survival probability is consistent among all the years, except from years 3 and 4, where no subjects had a failed prosthesis. As mentioned earlier, this plot and data are all sample data for illustrative purposes only.

Year	**At risk**	**Prosthesis failure**	**Withdrawn (censored)**	**Proportion surviving**	**Cumulative survival**	**Cumulative mortality**
0	50	0	0	1	1	1-1=0
1	50	1	0	49/50=0.98	1*0.98=0.98	1-0.98=0.02
2	49	1	0	48/49=0.9796	0.98*0.9796=0.96	1-0.96=0.04
3	48	1	0	47/48=0.9792	0.96*0.9792=0.94	1-0.94=0.06
4	47	0	1	1	0.94*1=0.94	1-0.94=0.06
5	46	1	0	45/46=0.9783	0.94*0.9783=0.9196	1-0.9196=0.0804
6	45	1	0			

Table 2: Sample data for showing how cumulative survival is calculated

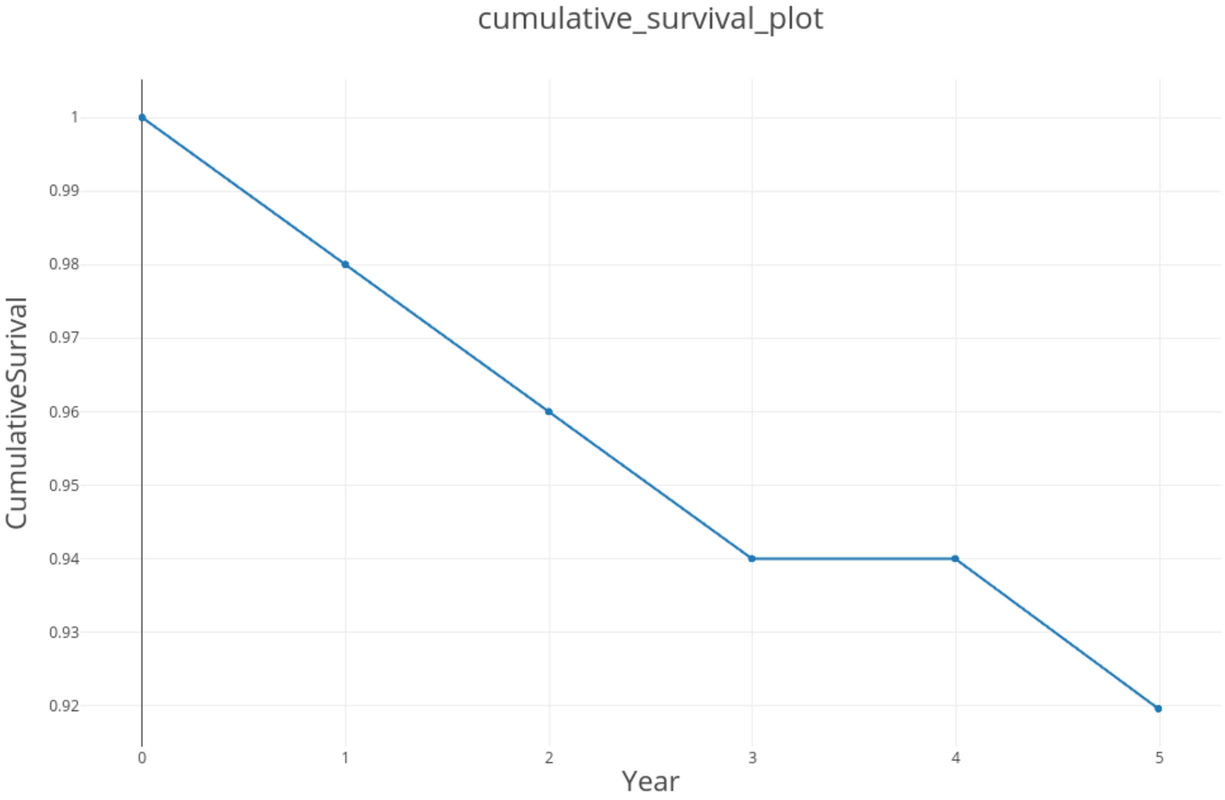


Figure 3: Cumulative survival plot which is generated using sample data

For comparing two types of prostheses, it is possible to generate plots from cumulative survival probabilities and present them in one graph.

Table 3 contains sample data of cumulative survival probabilities of two different prostheses. The table has three columns. Columns 2 and 3 contain the cumulative survival for each prosthesis, and column 1 contains the year.

One way to figure out which prosthesis is better than the other to plot the data presented in Table 3. In Figure 4, the data from Table 3 is turned into a plot. The Y-axis shows the cumulative survival probability, and the X-axis shows the year. The plot in Figure 4 makes it easy to see that the prosthesis named ”Cumulative_survival_1” is a better prosthesis since it consistently has a higher probability of survival than the prosthesis called ”Cumulative_survival_2”.

Year	Cumulative_survival_1	Cumulative_survival_2
1	1	1
2	0.98	0.95
3	0.96	0.9
4	0.94	0.83
5	0.94	0.75
6	0.9196	0.6

Table 3: Sample data over cumulative survival of two prostheses

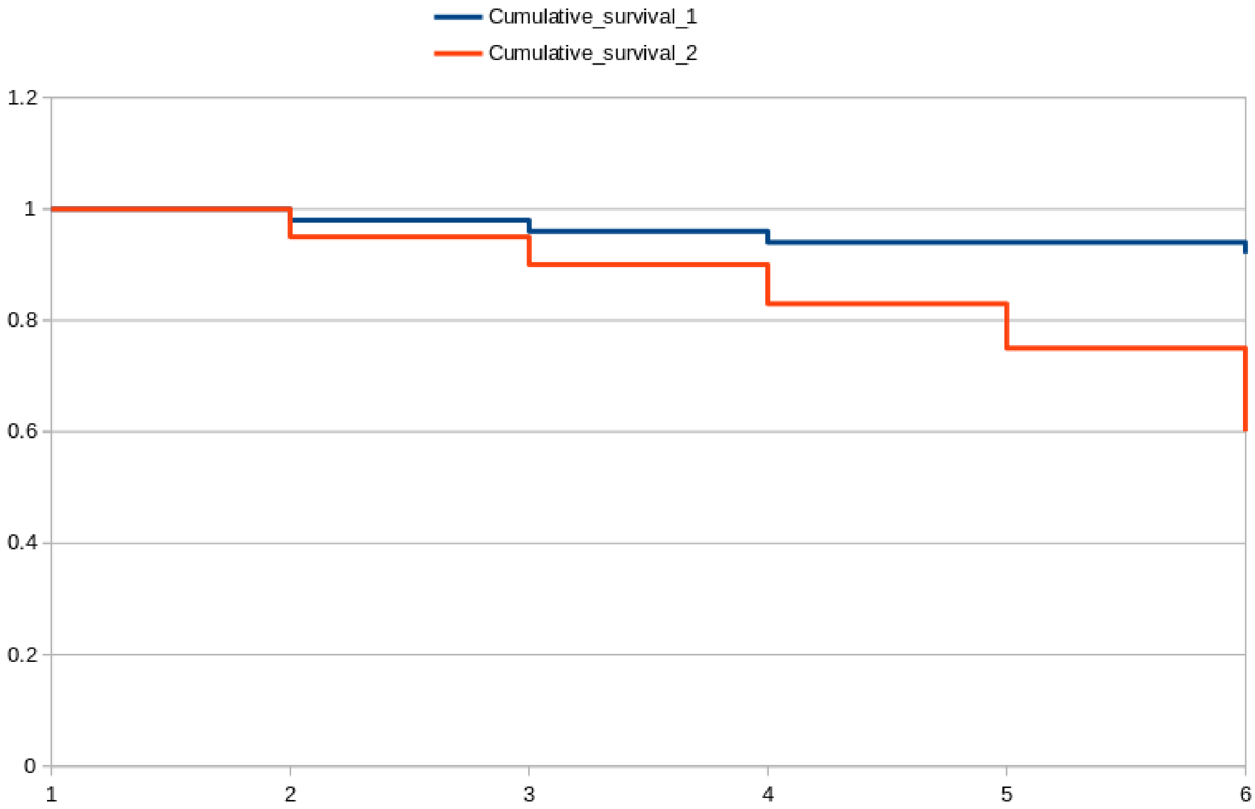


Figure 4: Cumulative survival plot of two prostheses which is generated using sample data from Table 3

6.3.3 The log-rank test

The log-rank test tests the hypothesis that there is no difference in probability between two populations in relation to an event such as survival at any time period (Bland and Altman, 2004, pp. 1073). The statistical significance (P-value) of the log-rank test score is calculated using Table 6 which contains chi-square values. The chi-square table’s degrees of freedom are calculated using the number of groups - 1 (Bland and Altman, 2004, pp. 1073). If the log-rank test value is 7 when comparing two groups, it is possible to see from Table 6 that the P-value is less than 0.01 and larger than 0.005. The end result is that the P-value is `P < 0.01`, meaning that the difference between the groups is statistically significant.

6.3.4 P-values

P-value as a measure of statistical significance is commonly used in medical articles. Normally a P-value of less or equal to 0.05 is considered significant, while a P-value larger than 0.05 is of no statistical significance (Nahm, 2017, pp. 241).

A P-value larger than 0.05 means that there is no evidence of a difference between the compared groups (Nahm, 2017, pp. 242).A P-value less or equal to 0.05 is indicating that the difference between the groups is statistically significant.

6.3.5 The Cox Proportional Hazard Model

The log-rank test can give a score between two groups' survival probabilities that tells if the groups' survival probabilities are statistically significant. But it only considers the whole groups and does not measure the effect of individual variables. The Cox proportional hazard model can give scores of individual variables and state if an individual variable positively or negatively affects survival time (Goel, Khanna, and Kishore, 2010, pp. 215).

In other words, if we want to investigate two prostheses types, the cox ph model can calculate if gender (male or female) positively or negatively affects the survival time based on hazard ratios. The cox ph model will give a hazard ratio associated with male and female in relation to the prosthesis types. For instance, if male have a hazard ratio of 1.3 and female a hazard ratio of 0.7 for the experimental prosthesis, the experimental prosthesis with gender female positively impact survival time. For male, the experimental prosthesis negatively impacts survival time.

Cox regression builds a predictive model that can test the effect of variables like men and women compared to the longevity of a prosthesis (*Cox Regression Analysis - IBM Documentation* 2021). However, since it is a predictive model, it does not display the absolute truth, like the Kaplan Meier method. Therefore it is always important to check the accuracy of the predictions. One measure that can be used for this purpose is the likelihood ratio, which is also implemented into the Web API using the Python library Lifelines (*CoxPHFitter lifelines* 2021).

The Cox proportional hazard model is based on the assumption that the two compared groups' hazard is constant in time. Consider the example of comparing males vs. females survival over their lifetime. If the males mortality rate is 1.2 (hazard ratio) compared to females, and the hazard ratio is constant over all measured survival times, then the proportionality is satisfied (Miller, 2013, pp. 2).

For measuring the accuracy of a Cox PH model, it is possible to use the concordance index. The concordance index or c index measures how well a random patient had a higher risk than the patient that experienced the event. A measure of 0.5 means that the predictions are completely random (*C-statistics: Definition, Examples, Weighting and Significance* 2021). Listed in Table 4, are the concordance values and their rating.

Score	**Explanation**
Below 0.5	Poor model
0.5	Completely random
Over 0.7	Good
Over 0.8	Strong
1	All predictions are correct

Table 4: Concordance index scores (*C-statistics: Definition, Examples, Weighting and Significance* 2021).

6.3.6 Hazard ratio

Hazard ratios are ordinarily calculated using the Cox proportional hazard model (Barraclough, Simms, and Govindan, 2011, pp. 978). Listed in Table 5, is an example of how to calculate the hazard ratio. GA means group A which is the control group, and GB means group B, which is a new experimental treatment. In Table 5, there are two time intervals, years 1 and 2. In year 1, the percentage that died in group B was 3/104 = 0.03, and in group A 4/104 = 0.04. To calculate the hazard ratio for year 1, we divide the group A (GA) percentages that died (0.03) with the percentages that died in the control group B (GB) (0.04). The hazard ratio for year 1 is therefore 0.03/0.04=0.750. We do the same for year 2, and the hazard ratio for year 2 is 0.06/0.08=0.743. Since the hazard ratios 0.750 and 0.743 are similar, the hazard ratio in this example would be approximately 0.75 (Barraclough, Simms, and Govindan, 2011, pp. 978). From this example, the hazard ratio of 0.75 tells us that the experimental treatment (GB) is better than the control group (GA).

A hazard ratio of 1 (HR = 1) means that both groups (treatments) are of the same quality. A hazard ratio below 1 (HR < 1) means that the new treatment is better than the control group. And a hazard ratio above 1 (HR > 1) means that the control group is better than the experimental treatment (Barraclough, Simms, and Govindan, 2011, pp. 978).

Year	**People (GA)**	**Dying (GA)**	**percentage dying(GA)**	**People (GB)**	**Dying (GB)**	**percentage dying(GB)**	**Hazard ratio**
1	104	4	4/104 = 0.04	104	3	3/104=0.03	0.03/0.04=0.750
2	104-4 = 100	8	8/100 = 0.08	104-3=101	6	6/101=0.06	0.06/0.08=0.743

Table 5: Hazard ratio calculation example. The example is based on (Barraclough, Simms, and Govindan, 2011, pp. 978).

df	**0.995**	**0.99**	**0.975**	**0.95**	**0.90**	**0.10**	**0.05**	**0.025**	**0.01**	**0.005**
1	—	—	0.001	0.004	0.016	2.706	3.841	5.024	6.635	7.879
2	0.010	0.020	0.051	0.103	0.211	4.605	5.991	7.378	9.210	10.597
3	0.072	0.115	0.216	0.352	0.584	6.251	7.815	9.348	11.345	12.838
4	0.207	0.297	0.484	0.711	1.064	7.779	9.488	11.143	13.277	14.860
5	0.412	0.554	0.831	1.145	1.610	9.236	11.070	12.833	15.086	16.750
6	0.676	0.872	1.237	1.635	2.204	10.645	12.592	14.449	16.812	18.548
7	0.989	1.239	1.690	2.167	2.833	12.017	14.067	16.013	18.475	20.278
8	1.344	1.646	2.180	2.733	3.490	13.362	15.507	17.535	20.090	21.955
9	1.735	2.088	2.700	3.325	4.168	14.684	16.919	19.023	21.666	23.589
10	2.156	2.558	3.247	3.940	4.865	15.987	18.307	20.483	23.209	25.188
11	2.603	3.053	3.816	4.575	5.578	17.275	19.675	21.920	24.725	26.757
12	3.074	3.571	4.404	5.226	6.304	18.549	21.026	23.337	26.217	28.300
13	3.565	4.107	5.009	5.892	7.042	19.812	22.362	24.736	27.688	29.819
14	4.075	4.660	5.629	6.571	7.790	21.064	23.685	26.119	29.141	31.319
15	4.601	5.229	6.262	7.261	8.547	22.307	24.996	27.488	30.578	32.801
16	5.142	5.812	6.908	7.962	9.312	23.542	26.296	28.845	32.000	34.267
17	5.697	6.408	7.564	8.672	10.085	24.769	27.587	30.191	33.409	35.718
18	6.265	7.015	8.231	9.390	10.865	25.989	28.869	31.526	34.805	37.156
19	6.844	7.633	8.907	10.117	11.651	27.204	30.144	32.852	36.191	38.582
20	7.434	8.260	9.591	10.851	12.443	28.412	31.410	34.170	37.566	39.997
21	8.034	8.897	10.283	11.591	13.240	29.615	32.671	35.479	38.932	41.401
22	8.643	9.542	10.982	12.338	14.041	30.813	33.924	36.781	40.289	42.796
23	9.260	10.196	11.689	13.091	14.848	32.007	35.172	38.076	41.638	44.181
24	9.886	10.856	12.401	13.848	15.659	33.196	36.415	39.364	42.980	45.559
25	10.520	11.524	13.120	14.611	16.473	34.382	37.652	40.646	44.314	46.928
26	11.160	12.198	13.844	15.379	17.292	35.563	38.885	41.923	45.642	48.290
27	11.808	12.879	14.573	16.151	18.114	36.741	40.113	43.195	46.963	49.645
28	12.461	13.565	15.308	16.928	18.939	37.916	41.337	44.461	48.278	50.993
29	13.121	14.256	16.047	17.708	19.768	39.087	42.557	45.722	49.588	52.336
30	13.787	14.953	16.791	18.493	20.599	40.256	43.773	46.979	50.892	53.672
40	20.707	22.164	24.433	26.509	29.051	51.805	55.758	59.342	63.691	66.766
50	27.991	29.707	32.357	34.764	37.689	63.167	67.505	71.420	76.154	79.490
60	35.534	37.485	40.482	43.188	46.459	74.397	79.082	83.298	88.379	91.952
70	43.275	45.442	48.758	51.739	55.329	85.527	90.531	95.023	100.425	104.215
80	51.172	53.540	57.153	60.391	64.278	96.578	101.879	106.629	112.329	116.321
90	59.196	61.754	65.647	69.126	73.291	107.565	113.145	118.136	124.116	128.299
100	67.328	70.065	74.222	77.929	82.358	118.498	124.342	129.561	135.807	140.169

Table 6: Chi-Squared probability table (*Chi-Square probabilities table* 2021)

6.3.7 Censoring

One commonly used term in survival analysis is censoring. Censoring in survival analysis is when the survival time to an event that occurs is missing (Prinja, N. Gupta, and Verma, 2010).

An example of censoring is if a clinical trial measures prosthesis longevity and a patient is dying before the end of that trial. Since a patient died, there is no way to know exactly how long the prosthesis could survive in that patient. The only information that existed was that the prosthesis lasted until the patient's time of death.

There are two types of censoring, right censoring and left censoring.

Point censoring / right censoring happens when the event does not occur or the patient is lost from the study (Prinja, N. Gupta, and Verma, 2010). Left censoring is if a patient has a risk for disease prior to the study. Left censoring is not considered a problem for clinical trials (Prinja, N. Gupta, and Verma, 2010).

”Removal of censored subjects from the data would lead to an unbiased outcome of survival time (Prinja, N. Gupta, and Verma, 2010).”

There exists no censored subjects in the experimental knee data used in this thesis. However, the original NAR data contains right-censored data. Examples of such data are patients who died and patients that moved out of the country before the study ended.

6.3.8 Why survival analysis?

Using survival analysis on knee arthroplasty can gain useful insight when comparing different prostheses. For instance, survival analysis can help answer how prosthesis A compares to prosthesis B before the first revision.

When doing survival analysis on the NAR data, comparing prostheses brands using survival analysis was of interest.

6.4 Clustering

Clustering is a part of unsupervised learning, where similar data are grouped together (Saxena et al., 2017, pp. 1). Clustering has been used for 3d object recognition, the grouping of DNA or protein sequences, handwriting recognition and automatic storage of documents like books. Here books were classified into, for instance, a class like computer-science (Saxena et al., 2017, pp. 22). Clustering has also been used in data mining for gathering knowledge from databases (Saxena et al., 2017, pp. 23).

Clustering algorithms group similar data into groups. The distance between the different groups centers should be greater than distances within the group. The algorithms do this based on a similarity measure like, for instance, Euclidean distance(Omran, Engelbrecht, and Salman, 2007, pp. 1).

6.4.1 Euclidean distance

Euclidean distance is the distance between two data points (Gartneer, 2020). In three-dimensional space, with three data points, the Euclidean distance is calculated as shown in Table 7, which is based on an example from Gartneer (2020). In Table 7, there are two data points, a and b. Data point a has the values 3,6 and 5, and data point b has the values 7,-5, and 1. The first step is to take 7-3 = 4 for the x value and do the same for the y and z values. The next step is to multiply the newly calculated values by themselves. The third step is to sum these newly multiplied calculated values (16+121+16 = 153). The last step is to take the square root of 153, which is 12,3693. The Euclidean distance between data point a and b in this example is therefore 12,3693.

Table 8 shows an extended example of Table 7. Here we calculate the Euclidean distance between the data points a and b, a and c, and a and d. The lower the Euclidean distance is, the more similarity the data points share. An Euclidean distance value of 0 means that the data points' values are a hundred percent similar, which is the case for a and d with an Euclidean distance of 0. The data point a and c have an Euclidean distance of 2, and data points a and b have an Euclidean distance of 12.3693. This means that a and c (2) are more similar than a and b (12.3693). Hence, the lower the Euclidean distance, the more similar the data points.

6.4.2 The K-Means clustering algorithm

K-Means clustering's objective is to generate a fixed amount of clusters that minimize the sum of squared Euclidean distances between objects (data points) and cluster centroids (S. Singh and Gill, 2013, pp. 2546).

data-point	**x**	**y**	**z**	**x(dist)**	**y(dist)**	**z(dist)**	**x(dist)^2**	**y(dist)^2**	**z(dist)^2**	**sum**	**squareroot (sum)**
a	3	6	5								
b	7	-5	1	7-3=4	-5-6=-11	1-5=4	4*4=16	-11*-11=121	4*4=16	16+121 +16=153	sqrt(153)= 12.3693

Table 7: Euclidean distance calculation example based on Gartneer (2020).The example shows the distance between the two components a and b.

data-point	**x**	**y**	**z**	**x(dist)**	**y(dist)**	**z(dist)**	**x(dist)^2**	**y(dist)^2**	**z(dist)^2**	**sum**	**squareroot (sum)**
a	3	6	5								
b	7	-5	1	7-3=4	-5-6=-11	1-5=4	4*4=16	-11*-11=121	4*4=16	16+121 +16=153	sqrt(153)= 12.3693
c	3	6	3	0	0	-2	0	0	4	4	2
d	3	6	5	0	0	0	0	0	0	0	0

Table 8: Euclidean distance calculation example based on Gartneer (2020).The table shows the Euclidean distance between a and b, a and c, and a and d.

The k-means clustering algorithm starts with a fixed amount of cluster centers, and each object (data point) is added to its closest cluster center. After an object is added to a cluster center, the cluster center position is updated to fit in the middle/center of all its objects. This process is repeated until it reaches convergence (S. Singh and Gill, 2013, pp. 2546).

6.4.3 DBSCAN clustering

Density Based Spatial Clustering of Applications with Noise (DBSCAN) can detect randomly shaped clusters within databases that contain noise (Suthar, Rajput, and V. k. Gupta, 2013, pp. 1775). For detecting clusters, DBSCAN usually uses Euclidean distance. An advantage of DBSCAN compared to K-Means clustering is that DBSCAN automatically detects how many clusters exist. This is not the case for K-Means that requires the users to specify the number of clusters (Suthar, Rajput, and V. k. Gupta, 2013, pp. 1777).

The DBSCAN clustering algorithm has the two parameters EPS, and the minimum required data points to form a cluster (minPts). If minPts is set to 1, only 1 data point is required to form a cluster. If minPts is set to 4, 4 data points are needed to form a cluster. The EPS parameter decides the boundary for the clusters. The data points that are within the EPS neighborhoods parameter are added to the cluster. Additionally, the data points' own EPS neighborhood is added to the cluster if it is dense. If the neighborhood of a point's EPS does not contain enough data points, it is added as noise (Suthar, Rajput, and V. k. Gupta, 2013, pp. 1777).

6.5 Logistic Regression

Logistic regression is an algorithm that has the potential to be an effective predictor. In Endo, Shibata, and Tanaka (2007), it had the highest accuracy of 85.8% of predicting breast cancer survival (Endo, Shibata, and Tanaka, 2007, pp. 12).

Logistic regression works well when the outcome variable Y (predicted value) is categorical, and the X values are either categorical or continuous (Peng, Lee, and Ingersoll, 2002, pp. 4).

A central mathematical part of logistic regression is the natural logarithm of an odds ratio (Peng, Lee, and Ingersoll, 2002, pp3). The example of calculation is given in Table 9

”An Odds Ratio (OR) is a measure of association between an exposure and an outcome. The OR represents the odds that an outcome will occur given a particular exposure, compared to the odds of the outcome occurring in the absence of that exposure.” (Szumilas, 2010, pp. 227)

Survival year	Men (M)	Female (F)	Total	Men>5	Females>5	Likelihood (M vs F)>5	ln(likelihood (M vs F)>5)
Above 5	73	15	73+15=88	73/24=3.17	15/11=1.36	3.17/1.36=2.33	LN(2.33)=0.845
Below 5	23	11	23+11=34				
Total	96	26					

Table 9: An example of how to calculate the odds ratio and the regression coefficient of logistic regression. The example is influenced by Peng, Lee, and Ingersoll (2002)

OR give a score of how well an exposed variable affects the event of interest. If the OR is equal to 1, the variable has no effect on the outcome. If the OR is higher than 1, the variable has a higher effect on the outcome. Lastly, if the OR is below 1, the variable has a lower effect on the outcome (Szumilas, 2010, pp. 227).

In Table 9 is an example influenced by Peng, Lee, and Ingersoll (2002) of how to calculate the natural logarithm of an odds ratio. Here the natural logarithm of the odds ratio of men having a prosthesis surviving above five years compared to females are calculated. In the example, 73 men had a prosthesis that survived above five years and 23 men below five years. For females, there were 15 above five years and 11 below. For men, there were `73/24 = 3.17` times more likely for a prosthesis to survive above five years than below. The equivalent for females was 1.36. When comparing men surviving above five years relative to females, the probability is `3.17/1.36 = 2.33`. This means that men are 2.33 times more likely to survive above five years than females, and this number is also called the odds ratio. Lastly, to get the logistic regression coefficient for the gender predictor, there is a need to take the natural logarithm of the odds ratio of 2.33, which is in this example 0.845. The value of 0.845 is the regression coefficient β for a logistic regression model that can be used to model survival year above or below five years in relation to gender.

6.6 Making Data Analysis More Accessible

There exist many methods and programming languages for making data analysis available on the Internet. Our team chose to separate the programming code for the front-end team and the back-end team using the client-server model shown in Figure 5. For making the back-end APIs, we chose to use the FastAPI web framework for Python.

6.6.1 The client-server model and Web API

In this thesis, all the data mining tasks are run in a Web API that returns data analysis results as JSON.

A Web API is a type of API that uses a communication network. A Web API most often uses HTTP requests for communication and typically returns JSON or XML as output data (*What is an API?* 2021). This thesis Web API runs on the server and is, therefore, part of the server process in the client-server model.

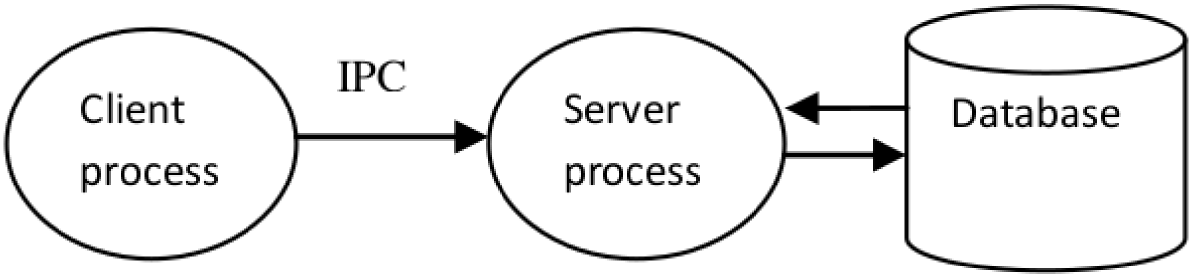


Figure 5: Client-server architecture (Sulyman, 2014, pp. 3)

According to Sulyman (2014), the client-server model works by using the client to request information while the server provides a service. Normally the server handles the data processing and sends the results to the clients. The server and the clients communicate with web protocols such as the Hypertext Transfer Protocol(HTTP), where clients send HTTP requests to the server, and the server acts on these requests (Kopecký, Fremantle, and Boakes, 2014, pp. 5).

The GET method has the variables inside the URI, and the POST method has the variables in the body. The GET method represents form information in a URL where only ASCII characters are accepted. The POST method accepts binary and ASCII characters and allows larger variable sizes than GET methods. The POST method also is more suitable than the GET method if a large amount of information should be retrieved from the server (*Difference Between GET and POST Method in HTML (with Comparison Chart and Examples)* 2021).

The mobile-first or API-first approach is an approach where websites are built on top of web APIs. In this approach, the team first builds a bunch of APIs and then creates a website as a client for the APIs (Kopecký, Fremantle, and Boakes, 2014, pp. 6).

In the development of the arthroplasty prototype, we chose the API-first approach. Using this approach, the back-end team created APIs first using the HTTP protocols GET and POST.

There are several advantages to building the APIs first for our project. The main advantage is that the back-end (server) team and the front end (client) team can have completely separate code. This is possible because there is no shared dependence among the back-end and front-end team code. Because of that separation, the front-end team has 100 percent freedom in choosing technologies. The front-end team can choose to create the client website in HTML and JavaScript or choose to create the front-end in a JavaScript framework like React. This is possible because all the data analysis jobs, like machine-learning, survival analysis, etc., are accessible through the back-end team APIs. All these APIs can be made available on the front-end using GET and POST requests inside various programming languages, such as, for instance, JavaScript.

Another advantage of using APIs is that it is easy to add additional functionality to the system. Should such additional functionality be needed, it is possible to create a new GET or POST request inside the back-end project folder where HTTP requests are located. Similarly, to use the newly added HTTP request on the client seems to be easy and efficient task to implement.

The architecture of the API is shown in Figure 6. In simple terms, the server works in the way that when the server receives POST or GET requests, it handles those requests in the FastAPI framework. When the FastAPI framework receives a specific GET request, the received GET request runs the code assigned to that GET request, and the same is the case for other HTTP requests.

An example is the GET request:

"http://localhost:8000/statistics/info/?feature_name=SURVYRS".

This GET request gives statistics about a feature or column name that lies in the arthroplasty database. When the FastAPI framework receives a specific GET request, it runs the code assigned to that GET request. All GET requests returns results that are sent back to the server. The result can be easily retrieved by the client using a front-end language like JavaScript.

Imagine that the Web API receives a GET request of the URL:

"http://localhost:8000/statistics/info/?feature_name=SURVYRS".

The first thing the Web API does is that it first reads all the row data for SURVYRS in the arthroplasty database, translates it into statistics, and lastly sends the statistical results back to the URL. The results which are sent back are in a JSON format.

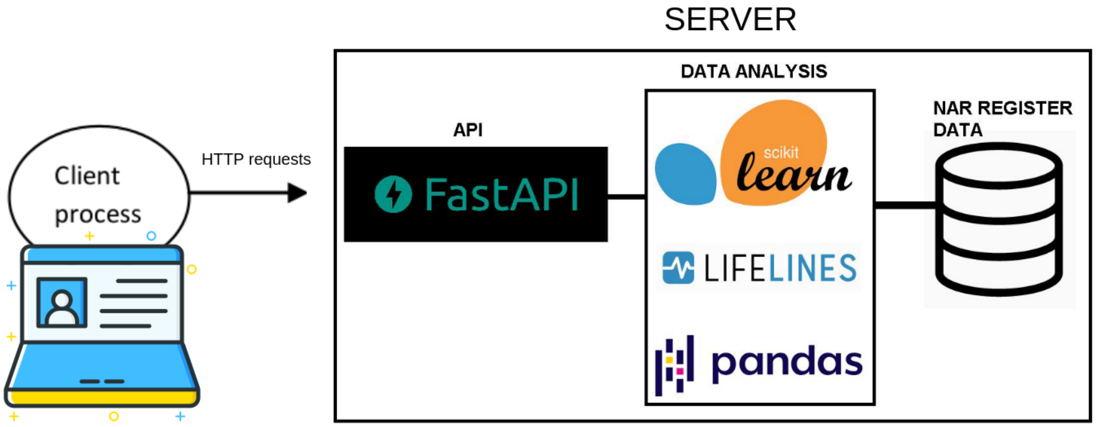


Figure 6: Client-server architecture for the master thesis prototype

6.7 The Dynamic Systems Development Method (DSDM)

The research within this thesis requires a development method that was suitable for small teams and could work in a limited timeframe. The Dynamic Systems Development Method (DSDM) fits these criteria. This method was mostly used by the back-end team.

DSDM focuses on satisfying business requirements within a dedicated short timeframe (Beynon-Davies et al., 1999, p. 211-212).

The DSDM method has nine principles. The first principle is "active user involvement"(Beynon-Davies et al., 1999, p. 213). Here it states that it is crucial that the users are actively involved in the development. The second principle is "teams must be empowered to make decisions"(Beynon-Davies et al., 1999, p. 213). Here it is important that both developers and users can make decisions in the project, where the user makes the requirements for the application, and the developers make the technical decisions. The third principle is "the focus is on frequent delivery of products"(Beynon-Davies et al., 1999, p. 213). This principle stresses the importance of holding an agreed timeline for deliveries. The fourth principle is "fitness for business purpose is the essential criterion for acceptance of deliverables"(Beynon-Davies et al., 1999, p. 213). This principle stresses that satisfying the business requirements within the timeframe is more important than satisfying the technical quality of the system (Beynon-Davies et al., 1999, p. 213). The fifth principle is "iterative and incremental development is necessary to converge on an accurate business solution"(Beynon-Davies et al., 1999, p. 214). This principle stresses the importance of delivering several partial solutions were all receive user feedback. Here, the user feedback in each iteration is used to improve the system (Beynon-Davies et al., 1999, p. 214). The sixth principle is, "all changes during development are reversible"(Beynon-Davies et al., 1999, p. 214). Meaning, the developer can go back and use previous versions of the system as needed. The seventh principle is "requirements are baselined at a high level"(Beynon-Davies et al., 1999, p. 214). Meaning, requirements, purpose, and scope should be frozen and be decided on a high level without going into much detail (Beynon-Davies et al., 1999, p. 214). The eight principle states that "testing is integrated throughout the life cycle" (Beynon-Davies et al., 1999, p. 214). Meaning, the testing of the system should be done incrementally to make sure that it meets business requirements and technical requirements (Beynon-Davies et al., 1999, p. 214). The nine principle states that "a collaborative and co-operative approach between all stakeholders is essential"(Beynon-Davies et al., 1999, p. 214). This principle states that it is important for the stakeholder to cooperate, and that cooperation is extra important if low-level requirements are not fixed (Beynon-Davies et al., 1999, p. 214).

Listed in Table 10 are the DSDM principles and how they have been used.

Principle	**Description**
1. Active user involvement	The arthroplasty register has influenced the requirements.
2. Teams must be empowered to make decisions	The team of programmers has made the technical decisions, while the arthroplasty register has influenced the requirements.
3. The focus is on frequent delivery of products	Different data mining methods have been made at different times throughout the project and shared with the front-end developers.
4. Fitness for business purpose is the essential criterion for acceptance of delivery	Implementing methods that could be of interest to the arthroplasty register has been a top priority. In addition to the requirements, all relevant literature was reviewed to identify suitable methods
5. Iterative and incremental development is necessary to converge on an accurate business solution	We have followed all feedback we could receive under this year‘s special circumstances.
6. All changes during the development are reversible	The team has made various changes throughout the project.
7. Requirements are baselined on a high level	The business requirements focused on detecting underperforming prostheses.
8. Testing is integrated throughout the life cycle	Testing of the system has been done throughout the development.
9. A collaborative approach between all stakeholders	The arthroplasty register has been involved in influencing the requirements. There has also been close collaboration within the group (front-end and back-end).

Table 10: The DSDM principles from (Beynon-Davies et al., 1999, p. 213-214), as applied to the research in this thesis.

7 Technologies

The project's development requires technologies that are suitable for data mining and in this case that can be utilized to make procedures available on the Internet. This Chapter, mentions the main programming languages used in the development. In addition there were a few tools used such as GitHub (*GitHub: Build like the best teams on the planet* 2021) and GitHub Desktop (*GitHub Desktop: Simple collaboration from your desktop* 2021) for source control.

7.1 Python 3

Early in the project, the back-end team decided to use Python 3 to build the back-end API. We chose Python 3 because we had previous experience with the language and many machine learning libraries are available in the language. An advantage of Python 3 is that it is a popular programming language. According to a survey among 24000 data professionals in 2018, Python was the most used and recommended language for new data scientists (Hayes, 2021).

Since the Python 3 standard library does not offer machine learning, survival analysis, and web API functionality out of the box, extra Python libraries were included to support this functionality. These libraries are listed below.

7.2 Scikit-learn

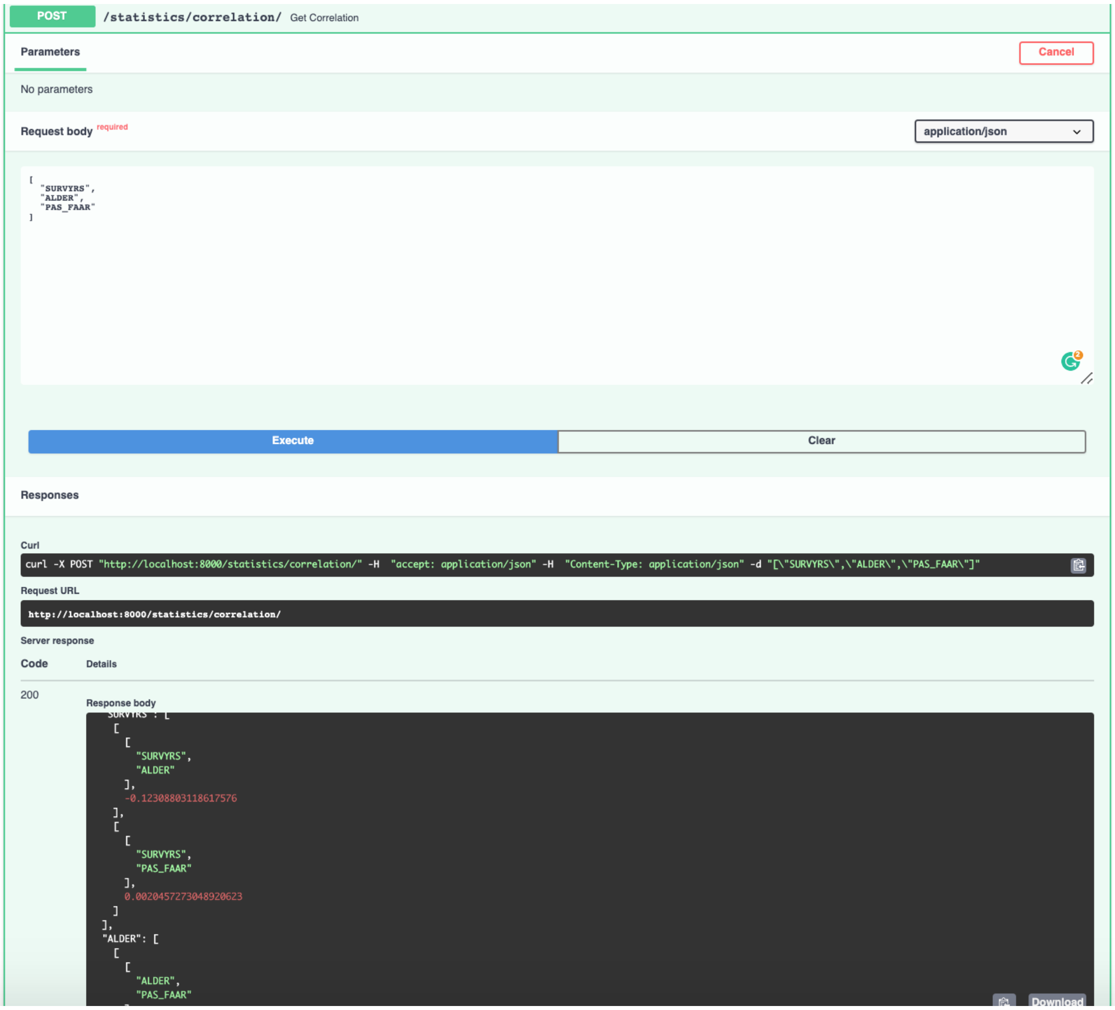
Scikit-learn is an Open Source Python module for machine learning available for both academic and commercial applications (Pedregosa et al., 2012, pp. 2826). Some of the machine learning methods available in Scikit are classification, regression, clustering, preprocessing, and more (scikit, 2021a). Various big corporations are using Scikit learn for multiple tasks. For instance, DataRobot is using it for predictive analytics, J.P.Morgan uses it for classification and predictive analytics, and Spotify is using it for music recommendations (scikit, 2021b).

7.3 Lifelines

The Python library Lifelines enables the use of survival analysis in the Python programming language (Davidson-Pilon, 2019, pp. 1). Lifelines was included because it offers survival analysis methods, which are not part of Scikit learn.

7.4 FastAPI Web Framework

There was a need to include a web framework for making the data mining methods available on the Internet. The back-end team chose the web framework FastAPI. We chose FastAPI because this web framework has the right amount of functionality for the project. Additionally, FastAPI is a web framework for making APIs (*FastAPI* 2021), which makes it perfect for the client-server architecture. Another advantage is that it has automatic interactive documentation (*FastAPI* 2021). An example of this type of automatic documentation can be seen in Figure 7.



```
/statistics/correlation/
```

```
[
  "SURVYRS",
  "ALDER",
  "PAS_FAAR"
]
```

```
curl -X POST "http://localhost:8000/statistics/correlation/" -H  "accept: application/json" -H  "Content-Type: application/json" -d "[\"SURVYRS\",\"ALDER\",\"PAS_FAAR\"]"
```

```
http://localhost:8000/statistics/correlation/
```

200

```
  "SURVYRS": [
    [
      [
        "SURVYRS",
        "ALDER"
      ],
      -0.12308803118617576
    ],
    [
      [
        "SURVYRS",
        "PAS_FAAR"
      ],
      0.0020457273048920623
    ]
  ],
  "ALDER": [
    [
      [
        "ALDER",
        "PAS_FAAR"
```

Figure 7: FastAPI documenatation example for retrieving correlation values. In this example, we retrieve correlations for the variables: SURVYRS, ALDER and PAS_FAAR

7.5 Flask

Flask is a Python web framework that has built-in Jinja templating support (*jinja documentation* 2021) (*Flask Documentation* 2021). Flask has been used in this thesis for building a minimal front-end to show off the APIs capabilities.

7.6 Postman

Postman (2021) has been used in this thesis for creating and doing testing on the APIs. The advantage of Postman is that it supports testing of HTTP requests which our API uses.

7.7 Miscellaneous Technologies

7.7.1 GitHub

Over 65 million developers use GitHub (*GitHub: Build like the best teams on the planet* 2021). GitHub has built-in access control, allowing only certain people to access a code repository, which makes the code more secure. Additionally, GitHub makes it easier to collaborate on code development.

In this project, GitHub has been used to store the API code and the Flask code. Once the code is stored on GitHub, it is easy for others in the group to retrieve the latest code, as all who subscribe are notified when updates occur.

7.7.2 Pandas

Pandas is an Open Source Python library for data analysis and manipulation (*Pandas - Python Data Analysis Library* 2021).

Pandas have been widely used in the project to read and manipulate data. It has been used for turning text data into numbers, for exploring the data by displaying statistics, selecting and deleting columns before doing data mining, and more.

7.7.3 Rpy2 package

Certain plots in this thesis are generated using the R programming language (*The R Project for Statistical Computing* 2021) in combination with Python. In other words, we got access to the R library within Python code thanks to the rpy2 package (*rpy2 - R in Python* 2021), which enables the use of R code within Python code.

Like the Python programming language, R also contains various packages, which extend the functionality of the programming language.

The R package and ggplot2 (*ggplot2* 2021) has been tested for generating a few plots. One of them can be seen in Appendix 16.

7.7.4 Django Web framework

At the beginning of the project, the back-end team tested the web framework Django (*Getting started with Django* 2021). Django is an Open Source web framework that includes a lot of functionality out of the box. It has built-in user authentication, security against SQL injection, a template system, and much more (*Getting started with Django* 2021).

We found that even though the framework was good,it required more code than we thought was necessary for creating APIs. This led to development taking longer than expected. Because of this, we later switched to the web framework FastAPI (*FastAPI* 2021), which requires less code, but also has less functionality included.

8 Data

8.1 NSD

Norwegian Center for Research Data (NSD) approved the project prior to working with the experimental knee data. The approval from NSD is shown in Appendix B 16.

8.2 Variables

The experimental dataset used in this project contains 344 columns and 1000 rows. Each row corresponds to one patient, and each row includes various information about what has happened to that patient at a specific point in time. A row also contains demographic information about patients, like age (ALDER), gender (PAS_KJONN), as well as current health at the time of surgery described here by a ASA physical status classification value (P_ASA).

One patient can be included in the dataset multiple times. For example, this can happen if a patient had surgery in 2010, but the prosthesis failed in 2015. In such cases, the patient would be registered as a revision surgery in 2015. That means that this particular patient would be registered twice in the dataset: once in row for 2010 with variables called primary surgery variables (P variables), and second time in 2015 registered with variables of revision surgery (R variables). Even though a patient can be included multiple times, the dataset does not contain a patient identifier due to the privacy regulation (the registry is not using patient national ID numbers). The dataset could be larger, depending on the purpose of the study. But in this case, it was not needed to include more patients, since the main objective was to develop the prototype.

Primary surgery variables (P variables) contain information about a primary surgery on a patient. The information includes which prosthesis components were used, how large the surgery was (P_AKT_OP_11_SPES), and the product brand of names of prostheses, etc. R variables contains only revision surgery information.

8.3 Statistics

This section contains various statistics extracted from the knee arthroplasty dataset that consisted of 1000 records and 344 variables.

The dataset is quite evenly distributed with primary surgeries (first surgery) and revision surgeries: that means 51% primary surgeries and 49% revision surgeries. Regarding revision surgeries, the dataset only contains patients that have only had one revision surgery.

The dataset is also quite evenly distributed regarding different types of prostheses. In Figure 8, those are shown, ranging in values between 154 and 176.

The gender of the patient in the dataset is indicated with the column name PAS_KJONN. PAS_KJONN and contains 508 Mann (men) and 492 Kvinne (women), which equals 50.8% males and 49.2% females.

The age column is called ALDER (age), and this column has a mean value of 54. The value distribution of the age column is shown in Figure 9. There it is possible to see that the age in the data ranges from 31 to 78 years old. It is also clear that there are fewer patients in the lower and higher age groups. Additionally, the genders are quite evenly distributed between age groups.

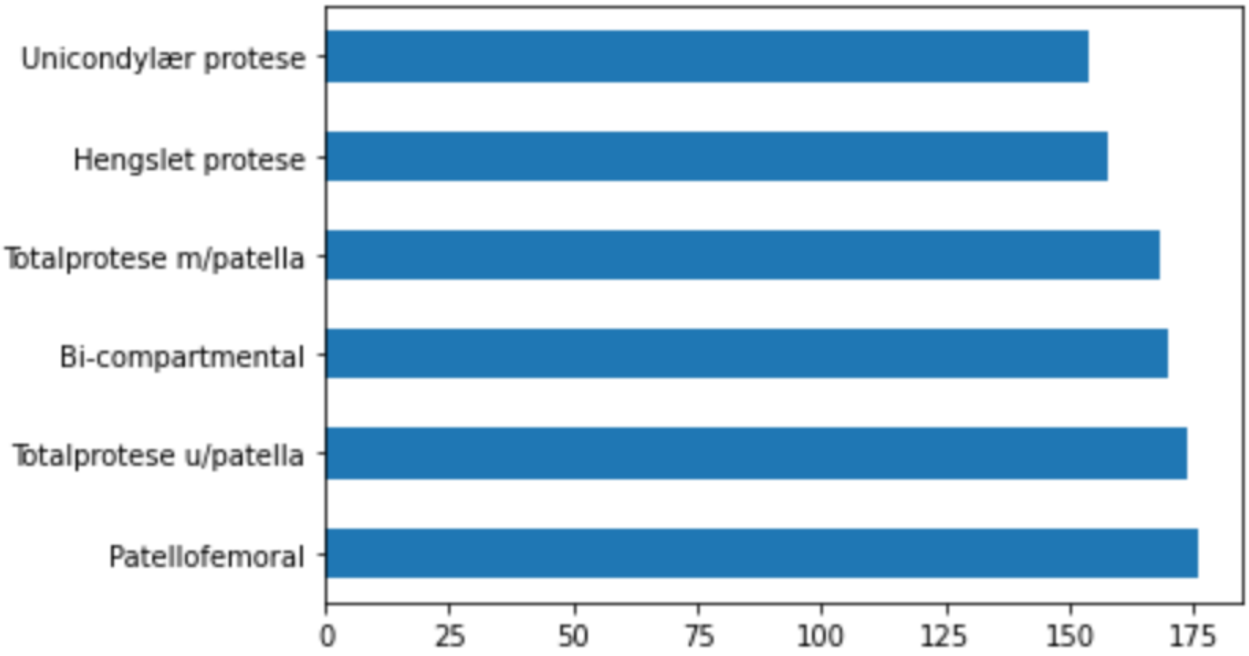


Figure 8: The value distribution of the different types of prostheses. The Y-axis indicates the type, and the X-axis indicates the number of prostheses

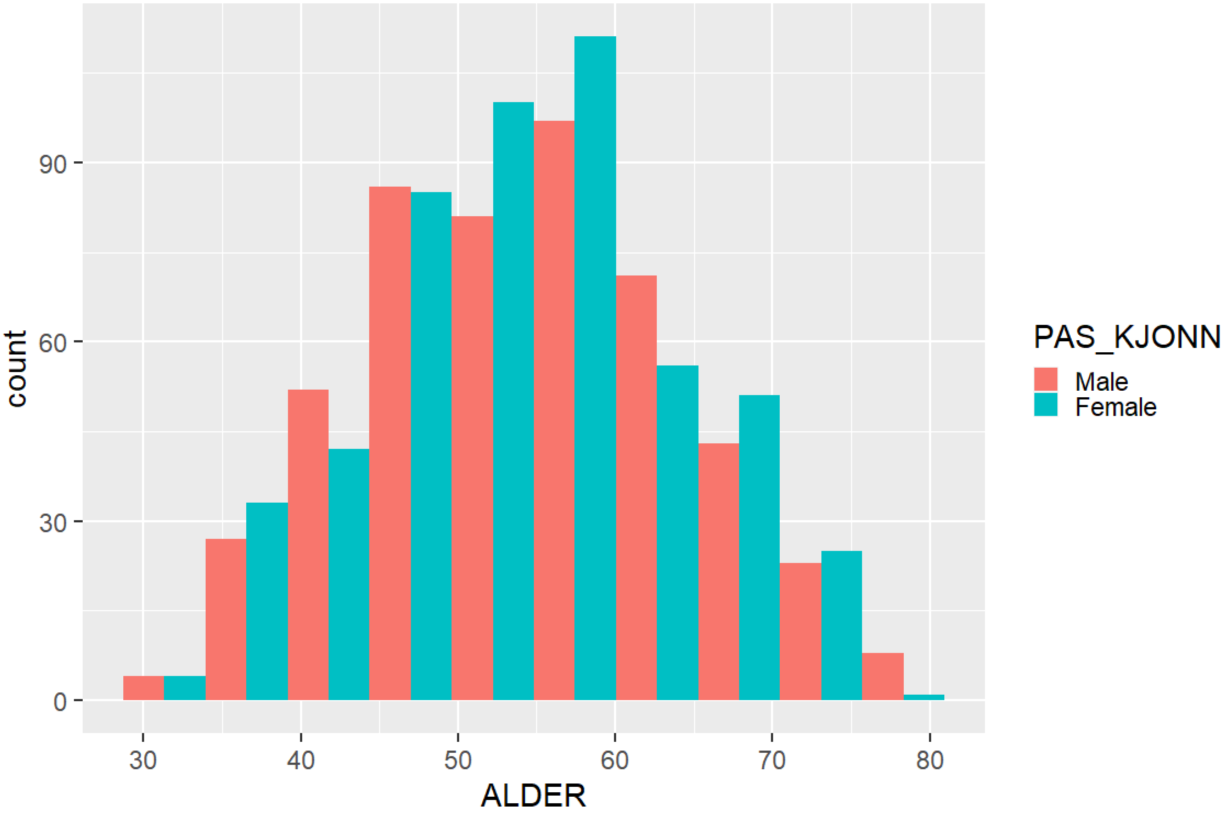


Figure 9: The value distribution of age (ALDER column). The X-axis indicates the age, and the X-axis indicates the number of values

The P_ASA column indicates the patients' health condition before a primary surgery. The value distribution of P_ASA is shown in Figure 10. It shows that the most common conditions are missing (Mangler) and symptomatic disease (Symptomatisk sykdom), and the least common are healthy (Frisk) and dying or near death (Moribund). However, overall the conditions are quite evenly distributed.

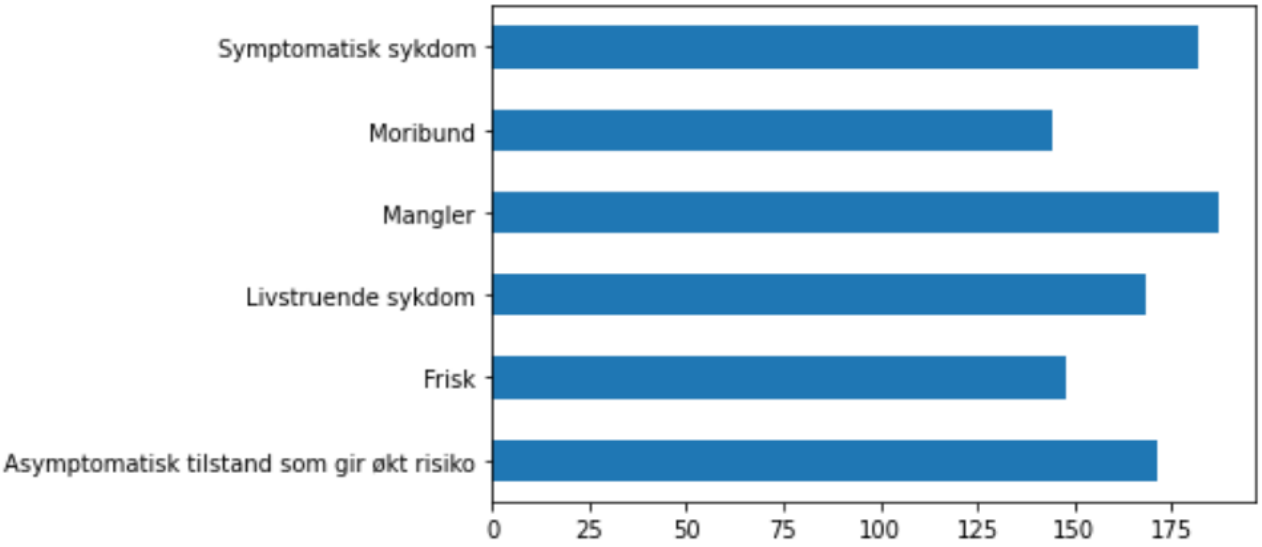


Figure 10: The value distribution of the P_ASA column. The Y-axis indicates the health condition, and the X-axis indicates the frequency

The FYLKE column indicates which county the patient lives in. Figure 11 shows the value distribution between the different counties. It is possible to see that the most registered counties are Sør-Trøndelag, Sogn og Fjordane and Troms and the least registered counties are Rogaland, Oslo, and Hordaland. The average value of patients registered per county is 52.6.

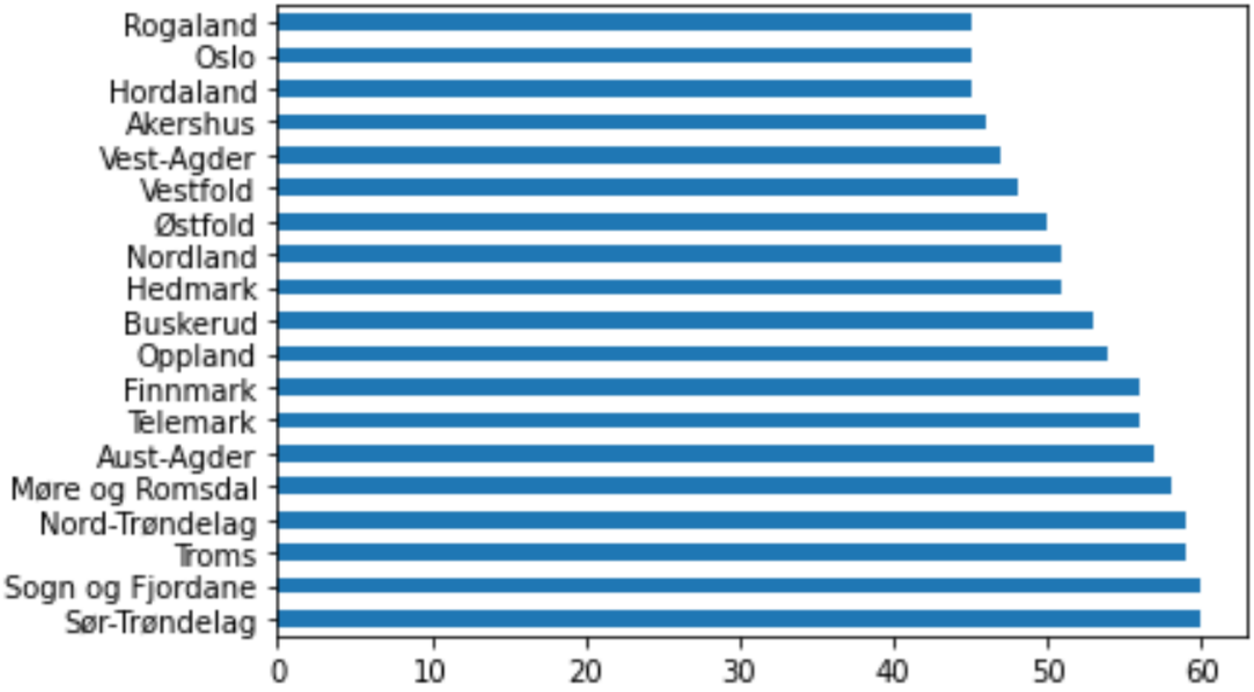


Figure 11: The value distribution of the FYLKE column. The Y-axis indicates the county, and the X-axis indicates the number of the frequency of patients that underwent surgery

For further investigation, the column SURVYRS is combined with `PAS_KJONN` in Figure 12. The column SURVYRS indicates the number of years from primary surgery until the end of the period or revision. From Figure 12, where SURVYRS is combined with `PAS_KJONN`, it is easy to see that both females (Kvinne) and males (Mann) are quite close in terms of survival on most of the survival years. Females are somewhat overrepresented in survival years 7 and 8, but males are surviving longer in survival years 6 and 9.

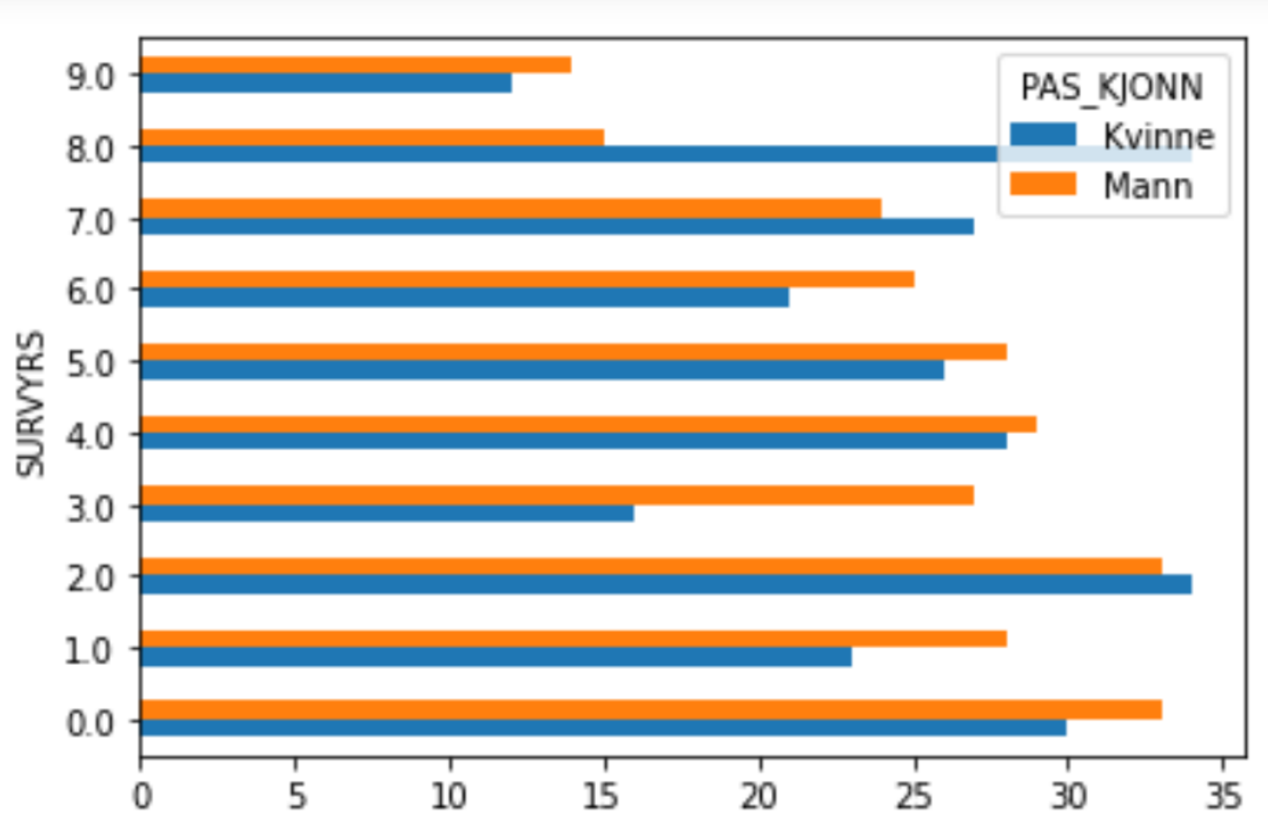


Figure 12: The value distribution of the PAS_KJONN column combined with the SURVYRS column. The Y-axis indicates the survival years, and the X-axis indicates the frequency of female(Kvinne) and male(Mann) patients

9 Identifying user needs and establishing requirements

User needs were identified at the arthroplasty registers offices on the 2nd of October 2020 when the team met with the arthroplasty register. The meeting was attended by two biomedical researchers, a statistician, and a professor of orthopedic surgery as well as the team of back- and front-end developers.

At the meeting, each team member started with a short presentation that was well received. After the presentation, the team members at the register explained that it was important for them to detect underperforming prostheses early. They used Cox PH analysis and Kaplan Meier analysis to detect underperformers, which is considered the standard.

It was also understood how important it was for the register to detect underperforming prostheses.

Requirements say what a system should do, and they are established based on what the user needs (Preece, Rogers, and Sharp, 2002, pp. 204). The requirements are divided between functional requirements that state what the system should do in practice and non-functional requirements, which are the system and development constraints (Preece, Rogers, and Sharp, 2002, pp. 205).

Based on the identified user needs, which are methods to detect underperforming prostheses, we have formulated the following requirements:

Functional Requirements

- Present statistics and results for clinicians and researchers.
- Do predictions of outcomes of TKR.
- Allow the user to perform classification such as Logistic Regression.
- Let the user be able to generate survival curves.
- Allow the user to detect risk factors for TKR analysis.
- Allow the user to perform cluster analysis.

Non-functional requirements

- Let the user choose which input data and parameters to train the model on (flexibility).
- Provide documentation for tasks available in the system.
- The choice of the backend technologies should not limit the choice of technologies that the front-end team chooses (interoperability).
- The testability of the system should be high (testability).

10 Development iterations

This project has been carried out as a part of back-end development with another master student T.Hufthammer, 2021. In this collaboration we focused on the development of the Web API for different data mining procedures. As the back-end team, we have also collaborated with two other master students Stolt-Nielsen, 2021 and Solheim, 2021 who worked on the front-end development. We all had weekly meetings throughout the project.

Ahead of each iteration, we had a small meeting to plan the activities for the upcoming iteration. These meetings were also held after each iteration as well to summarize the process.

We developed the system using the DSDM approach, but we used Trello boards as a supplementary tool to visually workflow and organize tasks. Trello boards allow for a KanBan-styled development workflow, which we used to organize the back-end development. Figure 13, shows tasks that were in progress, done or planned, within the back-end team.

One of the problems we encountered was that we initially received only hip data and no knee data. However, we were informed that this data is similar in structure, so we started developing the Web API. We programmed a Web API that was flexible and as adaptable as possible to accommodate the knee data. One of the strengths of the Web API is its flexibility.

10.1 First Iteration

We started the first iteration in November 2020 upon receiving hip data. The knee data was delivered late in April 2021, which was much later than expected. Therefore, we had no choice but to work with the data that was available at the current time.

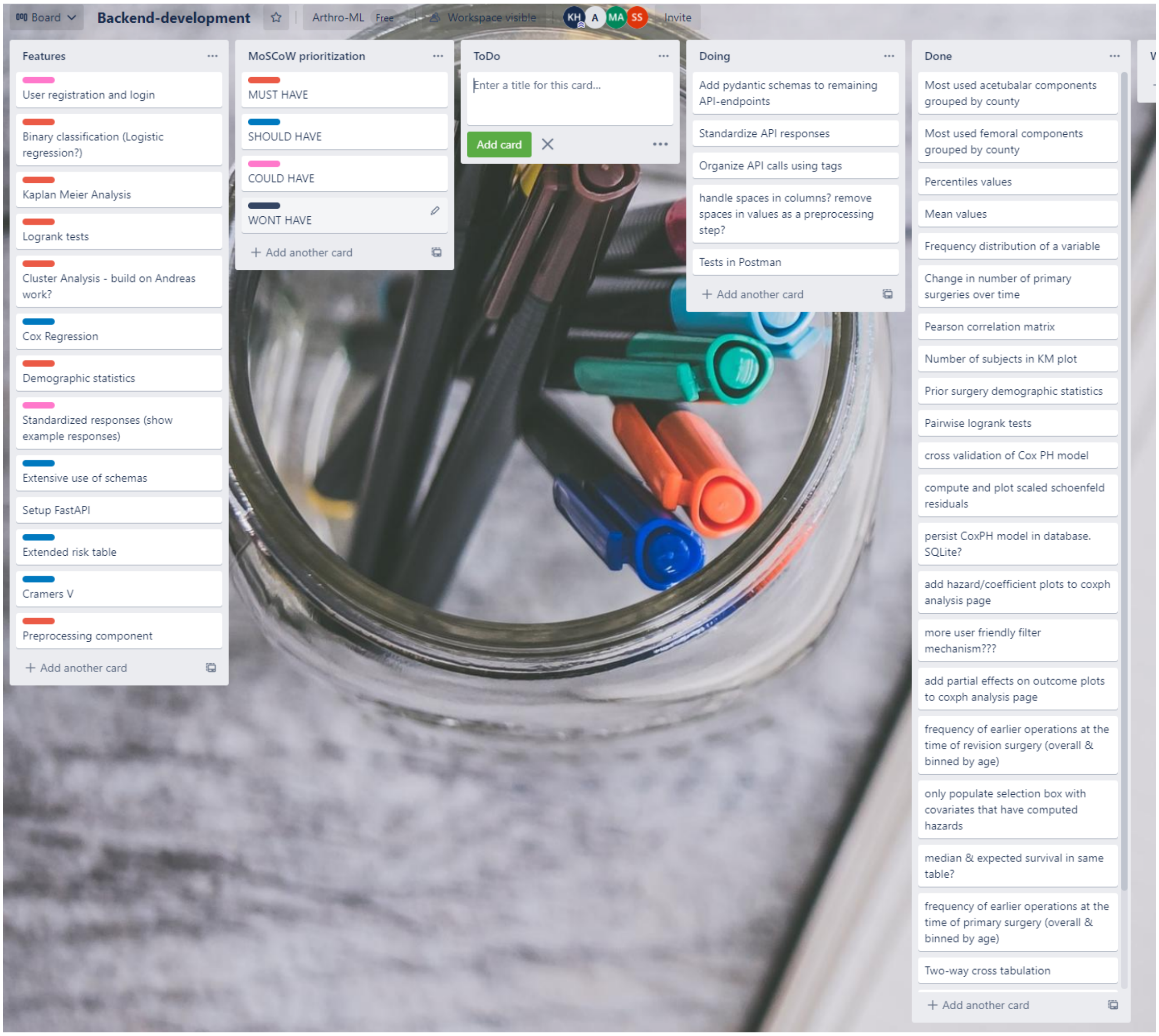
In the beginning, we explored the hip dataset to understand the structure of the dataset and find values that were important for analysis. We spent some time generating statistics and plotting visualizations in Jupyter Notebook which helped us get more acquainted with the data. We consulted the literature, master theses of former group members and looked into various procedures for data mining. Lastly, we had two meetings with two other front-end team members to discuss what type of information was interesting to implement in the API. We agreed that various statistics based on gender, age, and survival year of prostheses information were interesting. We also instructed the front-end team in how to use the API to extract data. The front-end team has helped us to understand that we could also benefit from a HCI perspective within back-end development. We understood that we had to regularly get together and share intermediary results for them to be informed and for us to get their feedback. Back-end developers are not usually expected to think about the final user but the functionality. However, front-ends are very important for users that are not programmers and need help to get into data mining in a more user friendly way.

After consulting with Solheim, 2021 halfway through the iteration, we began to implement methods on the API to retrieve information regarding prostheses use in the counties of Norway. Solheim, 2021 was interested in building an interactive map of Norway to display the most frequently used prostheses in the counties. In close collaboration with Solheim, 2021, we implemented several API endpoints to export such data.

Additionally, various API methods were implemented such as methods for retrieving the average survival of implants as shown in Appendix C 16. A method for retrieving the percentage and number of missing values are shown in Appendix F 16.

List of methods that were implemented:

- List of missing values (frequency counts)
- Average survival of duration of implants
- Reasons for revision



Features

- 🟪 User registration and login
- 🟥 Binary classification (Logistic regression?)
- 🟥 Kaplan Meier Analysis
- 🟥 Logrank tests
- 🟥 Cluster Analysis - build on Andreas work?
- 🟦 Cox Regression
- 🟥 Demographic statistics
- 🟪 Standardized responses (show example responses)
- 🟦 Extensive use of schemas
- Setup FastAPI
- 🟦 Extended risk table
- 🟦 Cramers V
- 🟥 Preprocessing component
- + Add another card

MoSCoW prioritization

- 🟥 MUST HAVE
- 🟦 SHOULD HAVE
- 🟪 COULD HAVE
- ⬛ WONT HAVE
- + Add another card

ToDo

- Enter a title for this card...
- Add card ✕ ...

Doing

- Add pydantic schemas to remaining API-endpoints
- Standardize API responses
- Organize API calls using tags
- handle spaces in columns? remove spaces in values as a preprocessing step?
- Tests in Postman
- + Add another card

Done

- Most used acetubalar components grouped by county
- Most used femoral components grouped by county
- Percentiles values
- Mean values
- Frequency distribution of a variable
- Change in number of primary surgeries over time
- Pearson correlation matrix
- Number of subjects in KM plot
- Prior surgery demographic statistics
- Pairwise logrank tests
- cross validation of Cox PH model
- compute and plot scaled schoenfeld residuals
- persist CoxPH model in database. SQLite?
- add hazard/coefficient plots to coxph analysis page
- more user friendly filter mechanism???
- add partial effects on outcome plots to coxph analysis page
- frequency of earlier operations at the time of revision surgery (overall & binned by age)
- only populate selection box with covariates that have computed hazards
- median & expected survival in same table?
- frequency of earlier operations at the time of primary surgery (overall & binned by age)
- Two-way cross tabulation
- + Add another card

Figure 13: Trello board for the back-end team

- Reasons for revision by different age groups
- Five most used Patella components
- Five most used Patella components by county
- Five most used Tibia components
- Five most used Tibia components by county

10.2 Second Iteration

The second iteration was initiated a week after finishing the first iteration. The first few days was spent fixing mistakes and issues that were identified in-between the first and second iteration from tests that we conducted in Postman Postman, 2021. Although all methods was working correctly in most cases, it turned out that some of the API endpoints failed in certain scenarios. We manually programmed a Kaplan Meier model during this iteration to better understand how the model actually works. After we had gained a better understanding, we used the survival library Lifelines to implement this model into the API. The advantage of the Kaplan Meier model in Lifelines compared to the manually programmed model is that Lifelines implementation also generates P values. The p-values from the Kaplan-Meier method is obtained from logrank tests. Regardless, this was a very useful experience to understand details of the methods and the demands of fully implementing the method.

During this iteration, we also incorporated a Logistic Regression component into the API using Scikit-learn. Integrating Logistic Regression into the API helped us gain experience with Scikit-learn which is widely used by industries, researchers and scholars.

In addition to this, we also created an API endpoint shown in Appendix D 16, which was given to the front-end developer Solheim, 2021. He could retrieve information and use it to visualize it on a map of Norway. This was the first thing that was implemented in our back and front end collaboration.

Midway through the second iteration, the back-end team discussed whether it would be a good idea to make a front-end to showcase the methods of the API. We figured that the 'raw data' obtained from the analyses on the API did not give us anything in terms of knowledge. In addition, having some kind of concrete presentation (front end) of our methods would help to gain feedback from potential users. Therefore, we decided to develop a minor front end application that display the result from the methods such that others could benefit from the analyses.

The latter part of the iteration was spent on the setting up a basic front end application in Flask. The first method that was integrated into the front end was the Logistic Regression component developed earlier in the iteration. Afterward, we held a meeting to plan out the next iteration and discuss potential improvements to the existing methods.

10.3 Third Iteration

During this iteration, we first implemented Cox PH analysis into the API. Afterward we implemented K-Means clustering, contingency table and Weibull analysis. We have also shared the results with front-end developers to inform them about the progress, and to welcome their comments, questions and feedback. They needed information in order to plan their own work and better understand what they needed to structure and show to the final user. This is a technical part of development that is hard to understand for the final user, and therefore it is important to create a proper user interface. The front-end team came up with some suggestions regarding the front-end solutions. Those included interface Stolt-Nielsen, 2021 and visualization Solheim, 2021. This allowed us in the back-end team to focus on the methods and leave the user interface design to the front-end.

10.4 Fourth Iteration

The test knee dataset was delivered during the fourth iteration by Dr. Peter Ellison in late April 2021. After that I could explore the knee prostheses data and adapt it to the API that was developed for the hip data.

The knee data could not be used with the API directly because some of the data types differed from the hip data and some of the knee data columns had the wrong data type. For instance numbers were coded as text. This meant that some API methods were incompatible with the knee data. Therefore there was a need to iterate through variables and transform the wrongly coded ones to a more appropriate data type. Another problem arose from mismatch in column names between hip and knee data, and this caused some API methods to not work for the knee data. To fix this problem, we removed "hardcoded" hip column names from the API that did not exist in the knee data. After that, most of the API methods worked fine for both hip and knee data.

The delivery of the knee data coincided with the development of the initial front-end application that we built with Flask. Therefore I focused my efforts towards exploring the knee data and adapting the API to fit the knee data instead. Thus, the other back-end developer T.Hufthammer, 2021 took lead in front-end development which I contributed to in the later stages. This helped us stay efficient and avoid stagnation in development.

After gaining access to knee data, Solheim, 2021 received statistical information from the API containing gender, age distribution and implant types for both hip and knee data. The API method for this is shown in Appendix E 16. Additionally, we helped Stolt-Nielsen, 2021 from the front-end team figuring out which knee column names might be interesting to include in their HCI prototype.

DBSCAN was also implemented into the API. See Appendix G 16 for the API endpoint.

The purpose of the minimal front-end was to receive proper user feedback and show off the capabilities of the API and its various methods.During this iteration, we also had evaluations of the front end.

During this iteration, we successfully managed to integrate the knee data to function properly with the API. The end result was an API that works with both knee and hip data.

11 Artifact

11.1 Introduction

According to Design Science, artifacts are artificial man-made things (Dresch, Lacerda, and Josè, 2015, pp. 106). There exists different types of artifacts according to Design Science. The artifact that is created is what is referred to as an instantiation. That is the implementation of an artifact in an environment (Dresch, Lacerda, and Josè, 2015, pp. 110).

In this thesis, the artifact is the web-based API that is built in cooperation with another student. The web-based API artifact consists of methods available through GET and POST requests, making it possible to run different data analysis methods online. These methods consists of clustering, various statistics, and survival analysis.

The design of the API can be seen in Figure 6. The web-based API also has a web-based auto-generated documentation interface where it is possible to test the built POST and GET requests. A screenshot of a part of the web-based documentation interface can be seen in Figure 7.

The web API and a minimal frontend

The data mining methods presented in this thesis are made accessible to end-users through a Web API. However, even though the web API can perform data mining methods, the results from the API are always displayed as text and numbers in JSON format. Therefore, the results are not very visually pleasing. For example, methods such as Kaplan Meier usually presents the results as a chart. The Web API results for Kaplan Meier displays the chart information in a text format which is not ideal.

Because of this, the web API is not an ideal solution for receiving user feedback. Therefore a minimal frontend was developed to display the results from the Web API in a more pleasing manner. The Python framework Flask was used which enabled user friendly presentation of the results. All chart result information from web API are transformed into real charts. Therefore the results are more visually pleasing, and also better for receiving user feedback.

11.2 Usecase and Data Mining Tasks

The methods implemented within the prototype are focused on detecting underperforming prostheses. Kaplan Meier and Cox PH are typical methods used for this purpose. However, other methods can also be interesting for a statistician or a prostheses surgeon. Therefore, the Web API does also contain statistical methods and the ability to perform standard machine learning tasks like logistic regression and k-means clustering.

11.3 Applications of Scikit and Lifelines

The methods within the Web API are implemented using various Python libraries. The primary libraries used for performing data analysis are Scikit-learn and Lifelines. For performing statistics, Pandas was used.

11.4 API development using FastAPI

The Web API made using FastAPI performs all the data analysis tasks. It returns JSON for each API call. An image of the Web API that calculates statistical information is shown in Appendix E 16. And an image of the Web API that can perform information search for missing values is shown in Appendix F 16.

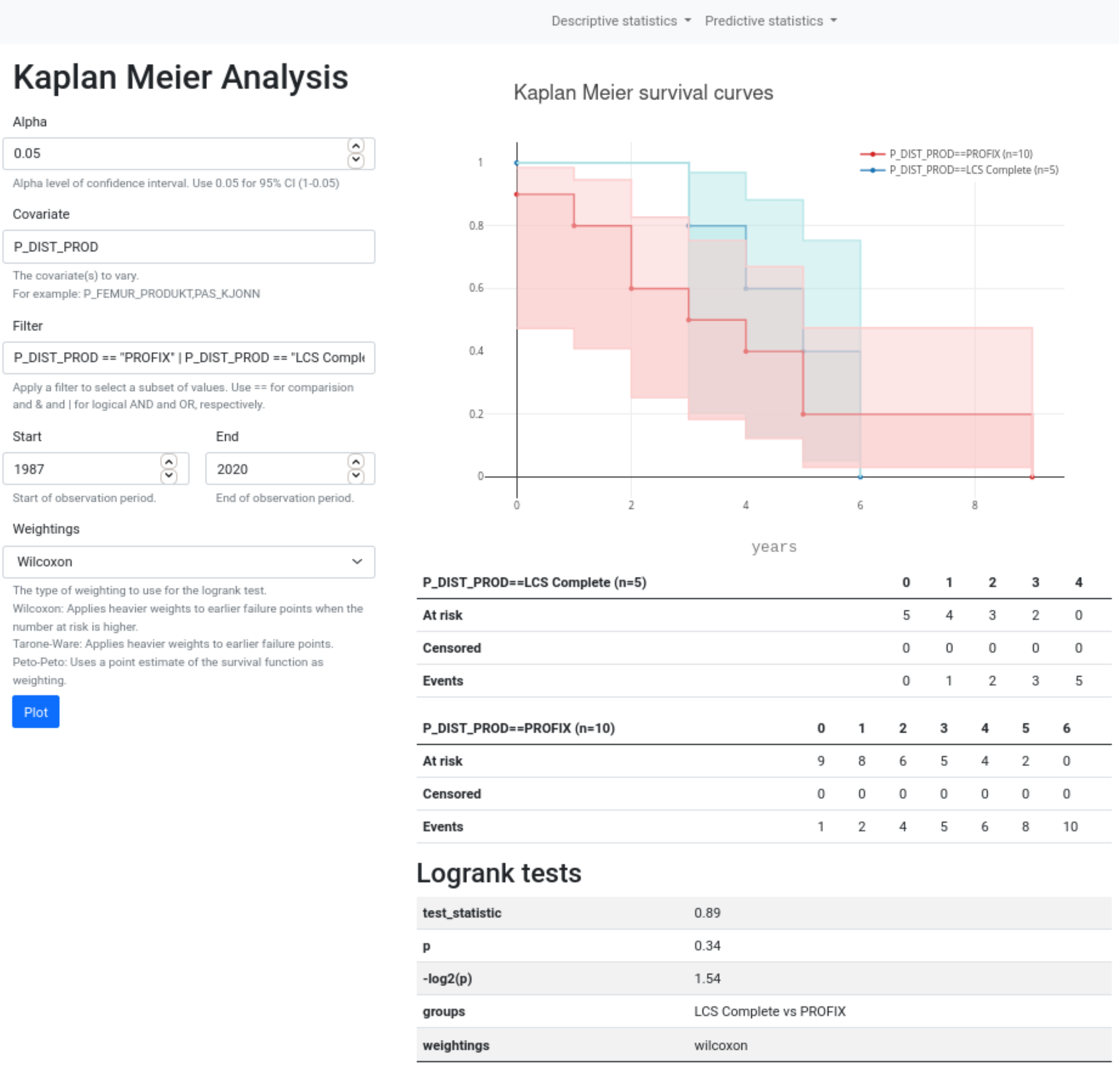
11.5 User Interface and Interactivity

This section shows the Flask prototype's visual aspect and the data analysis methods it can perform. Since the results were generated using an experimental dataset, the results are just illustrative. The Flask prototype uses the Web API to perform all the data analysis calculations. Together, the Flask prototype and the Web API make a functional prototype.

11.5.1 Kaplan Meier

Kaplan Meier is useful for comparing two prostheses and can be run in the Flask prototype.

The results from the Kaplan Meier analysis are shown in Figure 14. In this example, Profix prostheses are compared to LCS Complete. The survival curves suggest that Profix is expected to have higher survival than LCS Complete. This interpretation comes from the observation that the survival curve of Profix is above the survival curve of LCS Complete. The report from the artifact also generates P values, which is, in this case, 0.34, meaning that this finding is not statistically significant. The blue and red areas in the Kaplan Meier curve are the confidence intervals. These indicate the reliability of the estimate. A broad confidence interval indicates unreliable estimates. Conversely, a narrow confidence interval indicates a reliable estimate. The Sharp red and blue lines are the actual Kaplan Meier Curves.


Descriptive statistics ▾
Predictive statistics ▾
Kaplan Meier survival curves
P_DIST_PROD==PROFIX (n=10)
P_DIST_PROD==LCS Complete (n=5)
1
0.8
0.6
0.4
0.2
0
0
2
4
6
8
years
Plot

Kaplan Meier Analysis

Alpha

0.05

Alpha level of confidence interval. Use 0.05 for 95% CI (1-0.05)

Covariate

P_DIST_PROD

The covariate(s) to vary.
For example: P_FEMUR_PRODUKT,PAS_KJONN

Filter

P_DIST_PROD == "PROFIX" | P_DIST_PROD == "LCS Comple

Apply a filter to select a subset of values. Use == for comparision and & and | for logical AND and OR, respectively.

Start	End
1987	2020
Start of observation period.	End of observation period.

Weightings

Wilcoxon

The type of weighting to use for the logrank test.
Wilcoxon: Applies heavier weights to earlier failure points when the number at risk is higher.
Tarone-Ware: Applies heavier weights to earlier failure points.
Peto-Peto: Uses a point estimate of the survival function as weighting.

P_DIST_PROD==LCS Complete (n=5)	0	1	2	3	4
At risk	5	4	3	2	0
Censored	0	0	0	0	0
Events	0	1	2	3	5

P_DIST_PROD==PROFIX (n=10)	0	1	2	3	4	5	6
At risk	9	8	6	5	4	2	0
Censored	0	0	0	0	0	0	0
Events	1	2	4	5	6	8	10

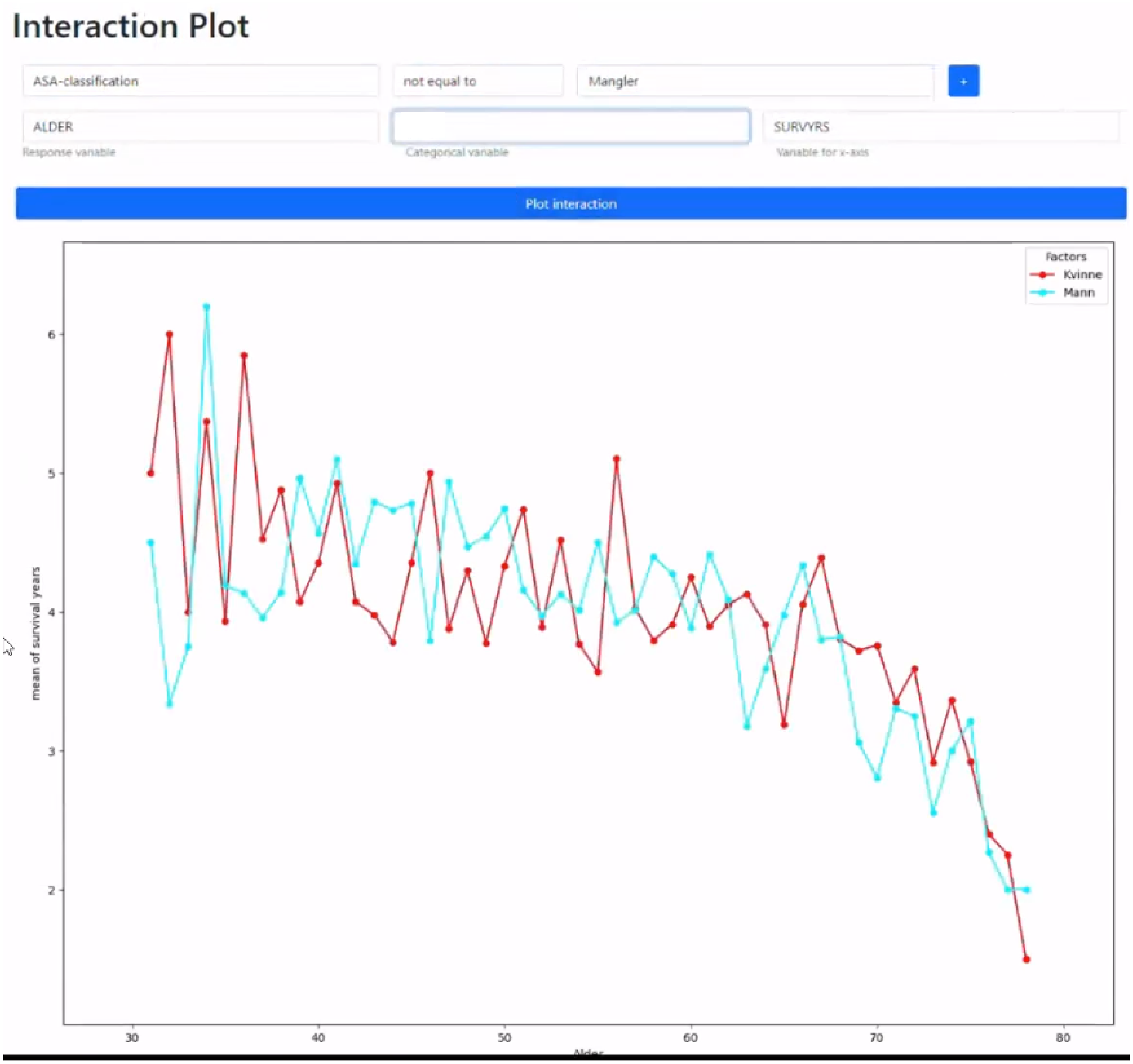
Logrank tests

test_statistic	0.89
p	0.34
-log2(p)	1.54
groups	LCS Complete vs PROFIX
weightings	wilcoxon

Figure 14: Kaplan Meier analysis generated from the Artifact

11.5.2 Interaction Plot

We developed a method on the Web API that produces a plot that can be used to identify interaction effects, i.e., an effect that coincides with the effect of the primary exposure variable. Figure 15 shows an interaction plot that measure the mean response in survival duration (Y-axis) against the age of the patient (X-axis) during primary surgery. We consider the gender of the patients as a possible interaction effect. The blue and red traces show the effect of gender ’male’ and ’female’, respectively.



Interaction Plot

ASA-classification | not equal to | Mangler | +

ALDER — Response variable

Categorical variable

SURVYRS — Variable for x-axis

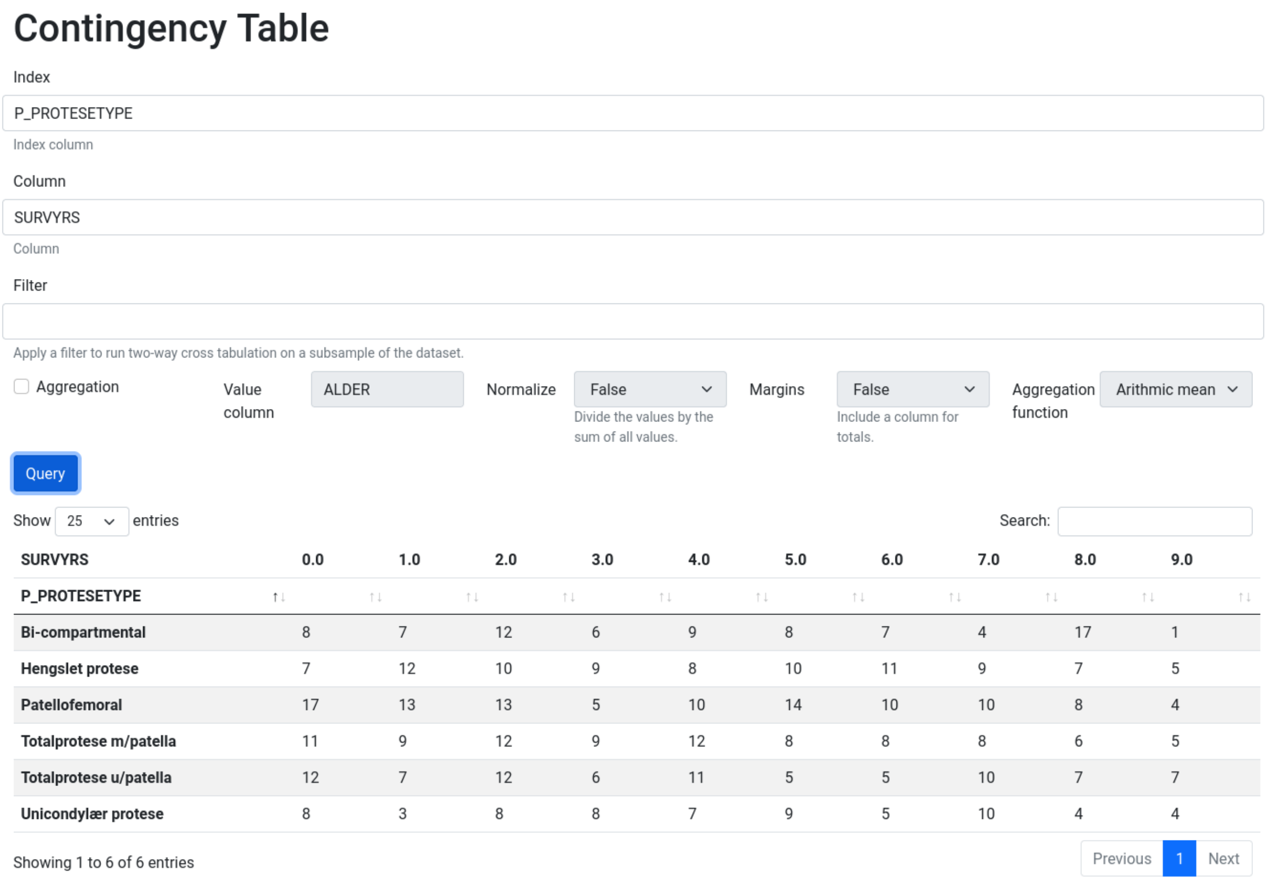
Plot interaction

Figure 15: Interaction plot

11.5.3 Contingency table

Contingency table shows the value of combining two columns in the dataset. But it also has additional functionality, like showing the min,max or average age of the values of the two combined columns. It is useful for gaining additional knowledge of the data.

In the table, the column SURVYRS are combined with P_PROTESETYPE. Each unique value in the column P_PROTESETYPE are on the Y-axis, and each unique value in SURVYRS are on the X-axis. The values for each combination are shown in the rows. If aggregation are turned on with ”arithmetic mean” and ALDER, the rows will display the average age.



Contingency Table

Index

P_PROTESETYPE

Index column

Column

SURVYRS

Column

Filter

Apply a filter to run two-way cross tabulation on a subsample of the dataset.

□ Aggregation

Value column: ALDER

Normalize: False
Divide the values by the sum of all values.

Margins: False
Include a column for totals.

Aggregation function: Arithmic mean

Show 25 entries

Search:

SURVYRS	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0
P_PROTESETYPE										
Bi-compartmental	8	7	12	6	9	8	7	4	17	1
Hengslet protese	7	12	10	9	8	10	11	9	7	5
Patellofemoral	17	13	13	5	10	14	10	10	8	4
Totalprotese m/patella	11	9	12	9	12	8	8	8	6	5
Totalprotese u/patella	12	7	12	6	11	5	5	10	7	7
Unicondylær protese	8	3	8	8	7	9	5	10	4	4

Showing 1 to 6 of 6 entries

Figure 16: Contingency table

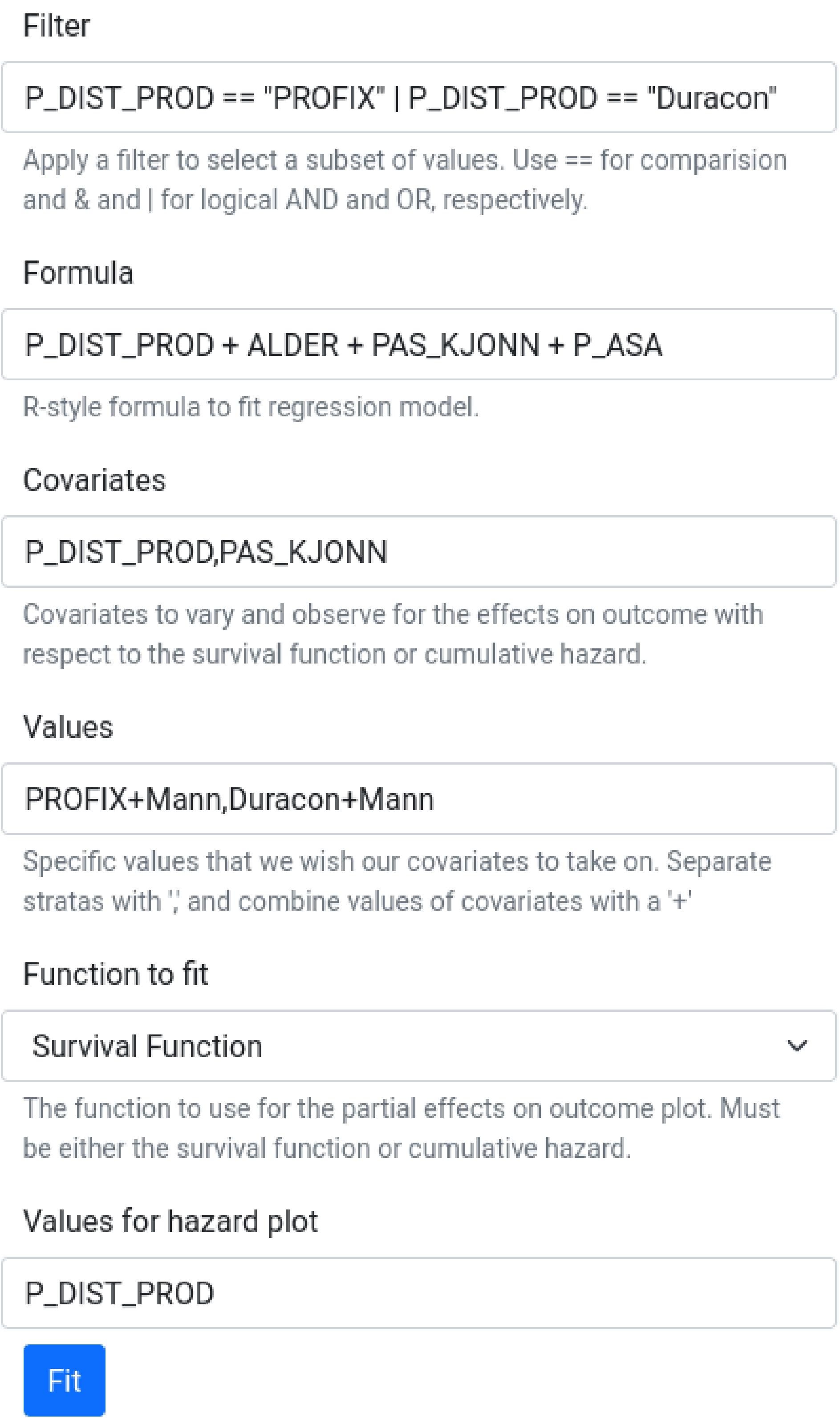
11.5.4 Cox PH regression

Examples of Cox PH regression are shown in Figures 17, 18 and 20. Figure 17 shows the most important input parameters for the Cox models. The predictor variables that are fitted to the model are P_DIST_PROD + ALDER + PAS_KJONN + P_ASA. However, from P_DIST_PROD only the values "PROFIX" and "Duracon" are included. Here we want measure the influence of gender on the primary exposure variables PROFIX and Duracon (prosthesis type).

As mentioned earlier, the concordance index is a measure of the accuracy of a Cox PH model. The concordance index for the fitted model is 0.71 (see Figure 18) which can be considered a 'good fit' (see Table 4). Based on the figure, it seems like the Duracon prosthesis is associated with a worse survival outcome (greater hazard) than the PROFIX prosthesis.

Figure displays the estimated regression coefficients, hazard ratios, 95% CIs, p-values, among other things. Here, the column *coef* corresponds to regression coefficients and *exp(coef)* corresponds to hazard ratios. We exponentiate the regression coefficients to obtain hazard ratios which are simpler to interpret. The table includes CIs which indicates the reliability of the estimates. CIs are available for both regression coefficients and hazard ratios. The p-values tells you whether effect of the hazard ratios are significant or not. A p-value higher than 0.05 indicates that the effect is non-significant. In this example, none of the estimate hazard ratios turned out significant although *P_ASA[Frisk]* came close with a p-value of 0.052. The hazard ratio of this factor was 1.155 which corresponds to a 944% higher risk. A 9 fold increase. However, the confidence band for this particular factor is extremely wide (90.768-0.982) which signifies that the estimate is not reliable. This should act as a reminder that confidence intervals should always be taken into consideration when interpreting hazard ratios.

The Cox PH procedure also has an additional section that users can use to analyse the results more thoroughly. In this section, the user may inspect hazard ratios, p-values, confidence intervals, predict longevity of prostheses, and visually assess the proportional hazard assumption. The analysis section contains much of the same information as the ’training’ section because it was developed at a later time. In future iterations, it would be wise to restrict the ’training’ section to only include the most essential details such as model summary and the proportional hazard assumption test.



Filter

P_DIST_PROD == "PROFIX" | P_DIST_PROD == "Duracon"

Apply a filter to select a subset of values. Use == for comparision and & and | for logical AND and OR, respectively.

Formula

P_DIST_PROD + ALDER + PAS_KJONN + P_ASA

R-style formula to fit regression model.

Covariates

P_DIST_PROD,PAS_KJONN

Covariates to vary and observe for the effects on outcome with respect to the survival function or cumulative hazard.

Values

PROFIX+Mann,Duracon+Mann

Specific values that we wish our covariates to take on. Separate stratas with ',' and combine values of covariates with a '+'

Function to fit

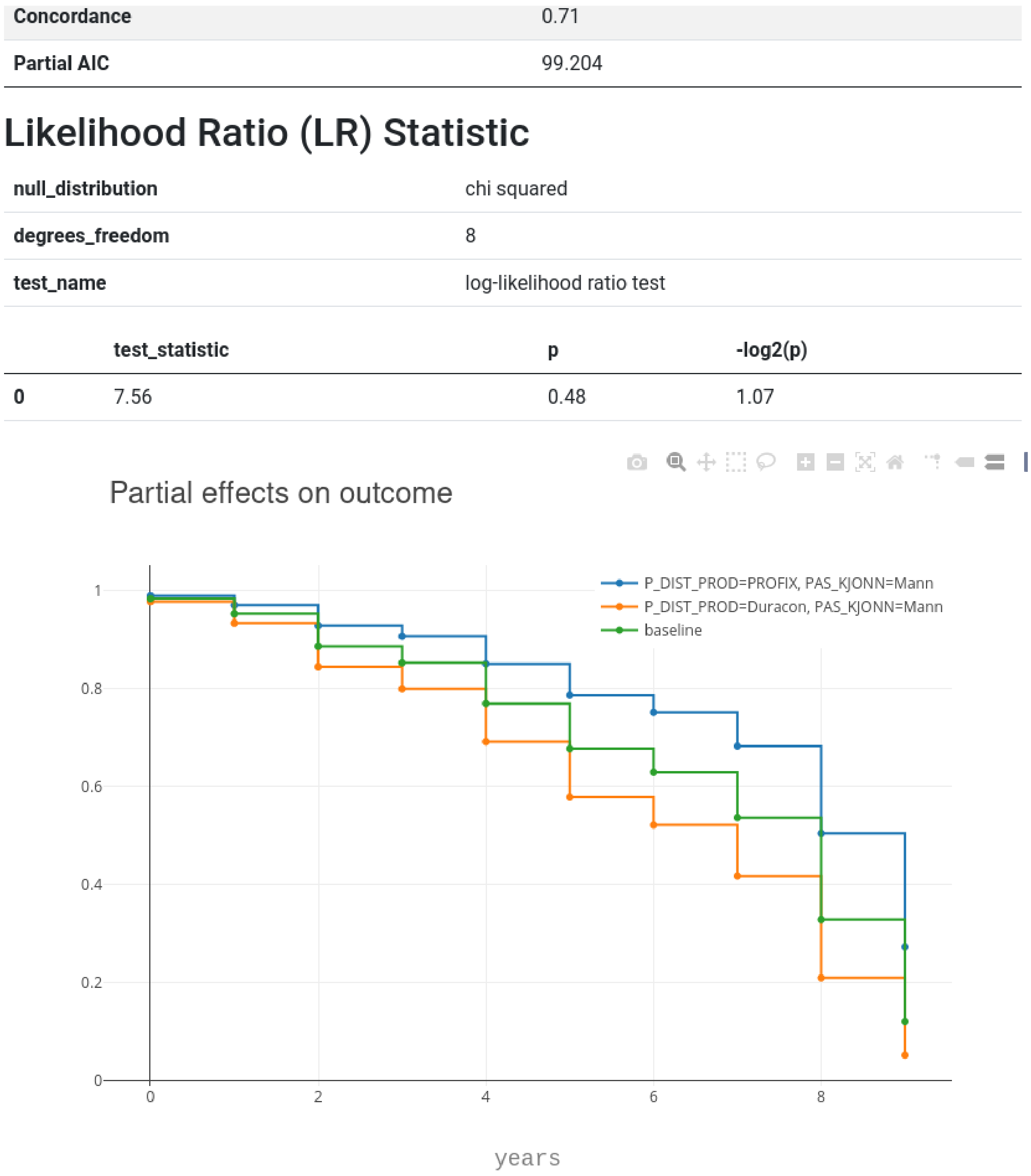
Survival Function

The function to use for the partial effects on outcome plot. Must be either the survival function or cumulative hazard.

Values for hazard plot

P_DIST_PROD

Figure 17: Cox PH report page 1



Concordance	0.71
Partial AIC	99.204

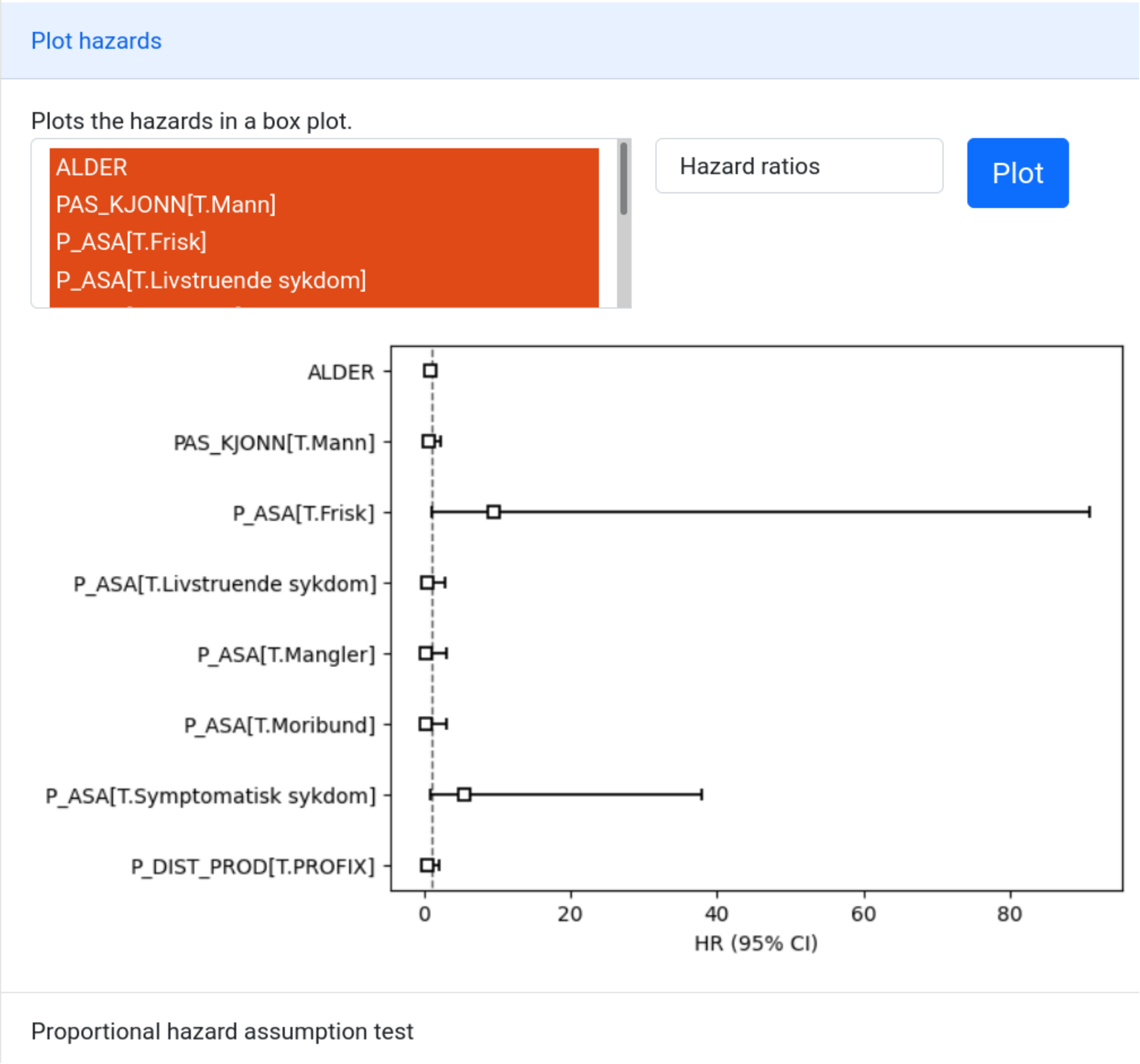
Likelihood Ratio (LR) Statistic

null_distribution	chi squared
degrees_freedom	8
test_name	log-likelihood ratio test

	test_statistic	p	-log2(p)
0	7.56	0.48	1.07

Partial effects on outcome
P_DIST_PROD=PROFIX, PAS_KJONN=Mann
P_DIST_PROD=Duracon, PAS_KJONN=Mann
baseline
1
0.8
0.6
0.4
0.2
0
0
2
4
6
8
years

Figure 18: Cox PH report page 2



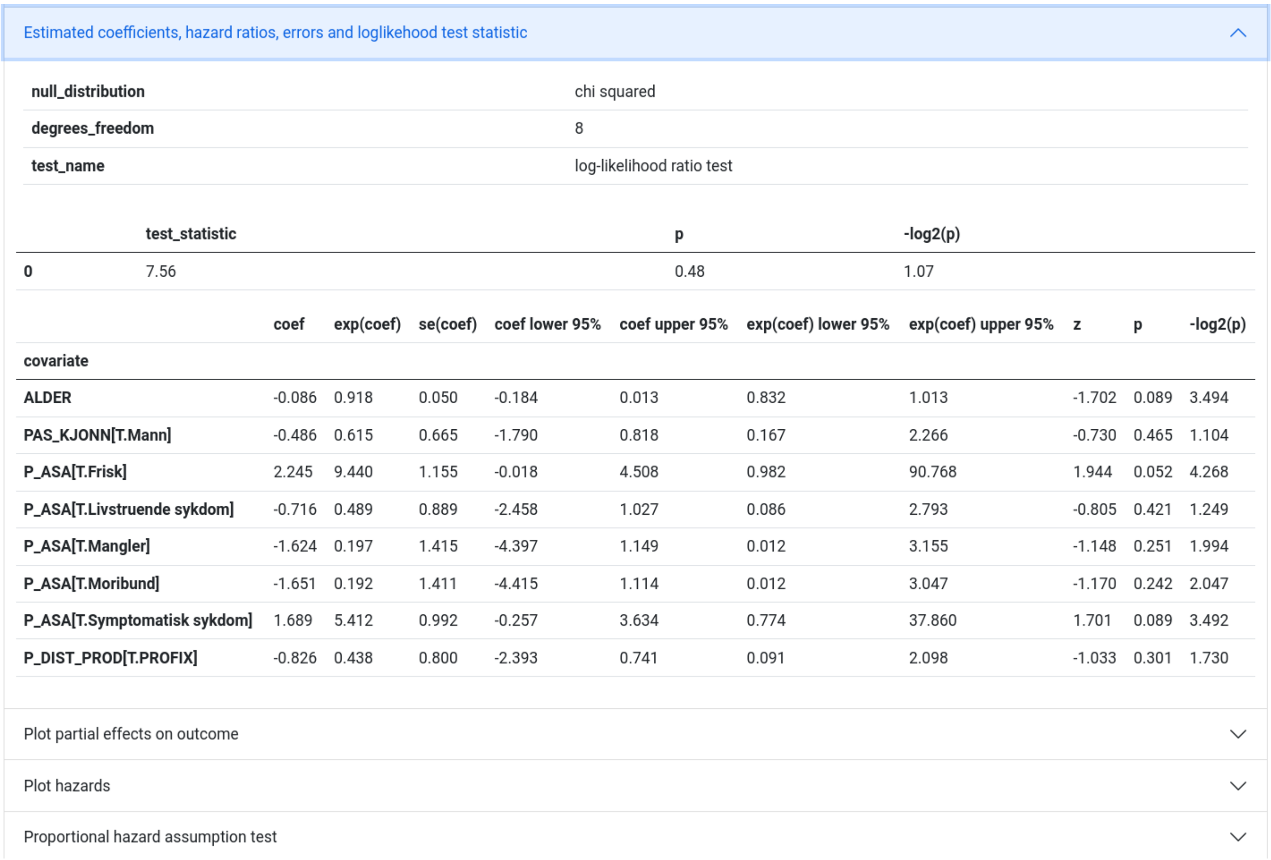
Plot hazards

Plots the hazards in a box plot.

ALDER
PAS_KJONN[T.Mann]
P_ASA[T.Frisk]
P_ASA[T.Livstruende sykdom]

Proportional hazard assumption test

Figure 19: Cox PH report page 3



Estimated coefficients, hazard ratios, errors and loglikehood test statistic

null_distribution	chi squared
degrees_freedom	8
test_name	log-likelihood ratio test

	test_statistic	**p**	**-log2(p)**
0	7.56	0.48	1.07

	coef	**exp(coef)**	**se(coef)**	**coef lower 95%**	**coef upper 95%**	**exp(coef) lower 95%**	**exp(coef) upper 95%**	**z**	**p**	**-log2(p)**
covariate										
ALDER	-0.086	0.918	0.050	-0.184	0.013	0.832	1.013	-1.702	0.089	3.494
PAS_KJONN[T.Mann]	-0.486	0.615	0.665	-1.790	0.818	0.167	2.266	-0.730	0.465	1.104
P_ASA[T.Frisk]	2.245	9.440	1.155	-0.018	4.508	0.982	90.768	1.944	0.052	4.268
P_ASA[T.Livstruende sykdom]	-0.716	0.489	0.889	-2.458	1.027	0.086	2.793	-0.805	0.421	1.249
P_ASA[T.Mangler]	-1.624	0.197	1.415	-4.397	1.149	0.012	3.155	-1.148	0.251	1.994
P_ASA[T.Moribund]	-1.651	0.192	1.411	-4.415	1.114	0.012	3.047	-1.170	0.242	2.047
P_ASA[T.Symptomatisk sykdom]	1.689	5.412	0.992	-0.257	3.634	0.774	37.860	1.701	0.089	3.492
P_DIST_PROD[T.PROFIX]	-0.826	0.438	0.800	-2.393	0.741	0.091	2.098	-1.033	0.301	1.730

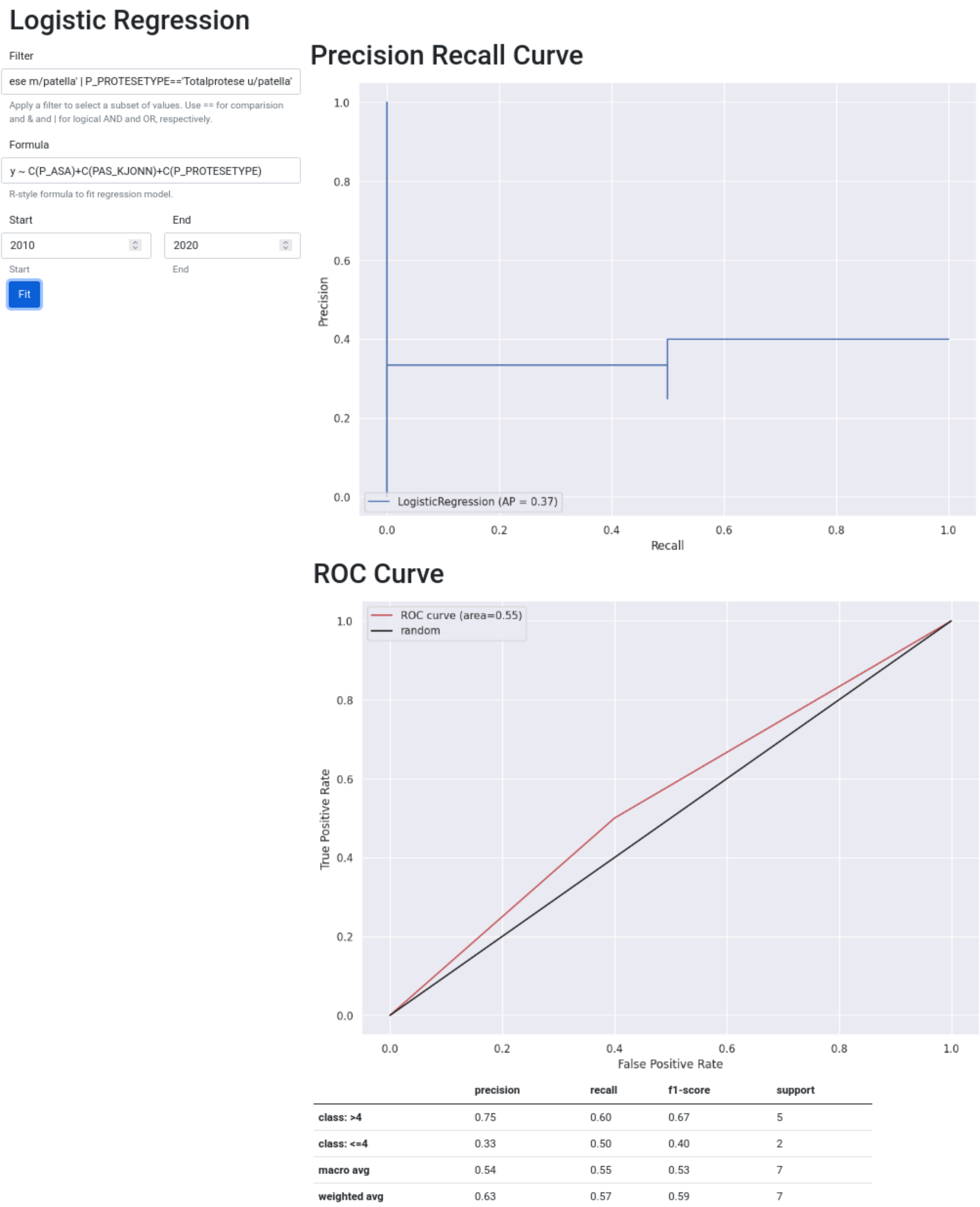
Plot partial effects on outcome
Plot hazards
Proportional hazard assumption test

Figure 20: Cox PH report page 4

11.5.5 Logistic Regression

The Logistic Regression procedure that we implemented is shown in Figure 21. The "Formula" specifies the predictors or explanatory variables to include in the model. Here, the 'y' variable corresponds to the response variable (dependent variable) that we want to predict. The variables following the *tilde* are the independent variables used to predict the response variable. The classification report shown on the right displays accuracy metrics for each target class. We 'hardcoded' the target classes as survival duration above or below 4 years. The filter input further narrows down the data for analysis to only include data that has the prosthesis types "Totalprotese m/patella" and "Totalprotese u/patella". We also provide the possibility to limit the time period to include prostheses from. This is useful if we want to fit the model to data from a specific period.

As shown in Figure 21, we also report a ROC and precision-recall curve.



Logistic Regression

Filter

ese m/patella' | P_PROTESETYPE=='Totalprotese u/patella'

Apply a filter to select a subset of values. Use == for comparision and & and | for logical AND and OR, respectively.

Formula

y ~ C(P_ASA)+C(PAS_KJONN)+C(P_PROTESETYPE)

R-style formula to fit regression model.

Start	End
2010	2020
Start	End

Fit

Precision Recall Curve

ROC Curve

	precision	recall	f1-score	support
class: >4	0.75	0.60	0.67	5
class: <=4	0.33	0.50	0.40	2
macro avg	0.54	0.55	0.53	7
weighted avg	0.63	0.57	0.59	7

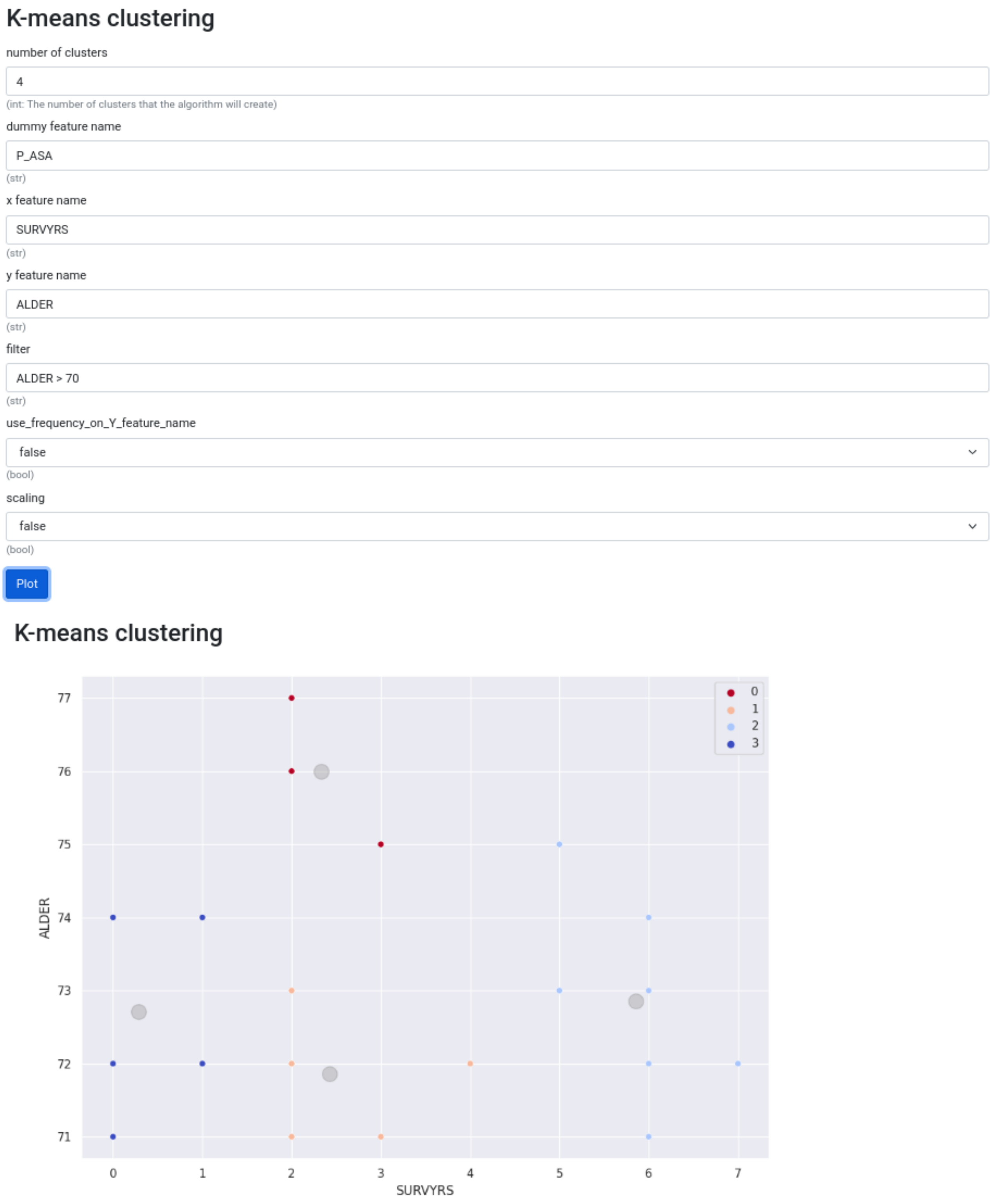
Figure 21: Logistic Regression

11.5.6 K-means clustering

We also developed an experimental procedure for K-means clustering. Although the procedure does not produce very interesting results using the sample data, it might serve as a good starting point for further development. It would be interesting to see what kind of results can be obtained on more realistic arthroplasty data. The procedure has been tested and validated on external datasets.

Examples of K-means clustering are shown in Figures 22, 23 and 24. In Figure 22 are the input parameters and the K-means clustering plot. The most important parameters are the variables it creates clusters from: "x feature name" and "y feature name". Other important parameters are the number of clusters it should generate and if scaling should be turned on or off. The parameter "use_frequency_on_Y_feature_name" combines the survival year of prostheses and the "y_feature_name" column in a way that it counts the number of values in "y_feature_name" for each survival year. Filter lets the user do analysis on a subset of the data.

The gray dots in K-means clustering are the cluster centers, and the different clusters are presented with the different colors. A K-means report with more precise information regarding the clusters is shown in Figure 23. The last plot that is generated from the report is shown in Figure 24. This plot displays the original unfiltered and unclustered data of column "y feature name", "x feature name" and the "dummy feature name". This plot can be useful for comparing the clustered data with the original data.



K-means clustering

number of clusters

4

(int: The number of clusters that the algorithm will create)

dummy feature name

P_ASA

(str)

x feature name

SURVYRS

(str)

y feature name

ALDER

(str)

filter

ALDER > 70

(str)

use_frequency_on_Y_feature_name

false

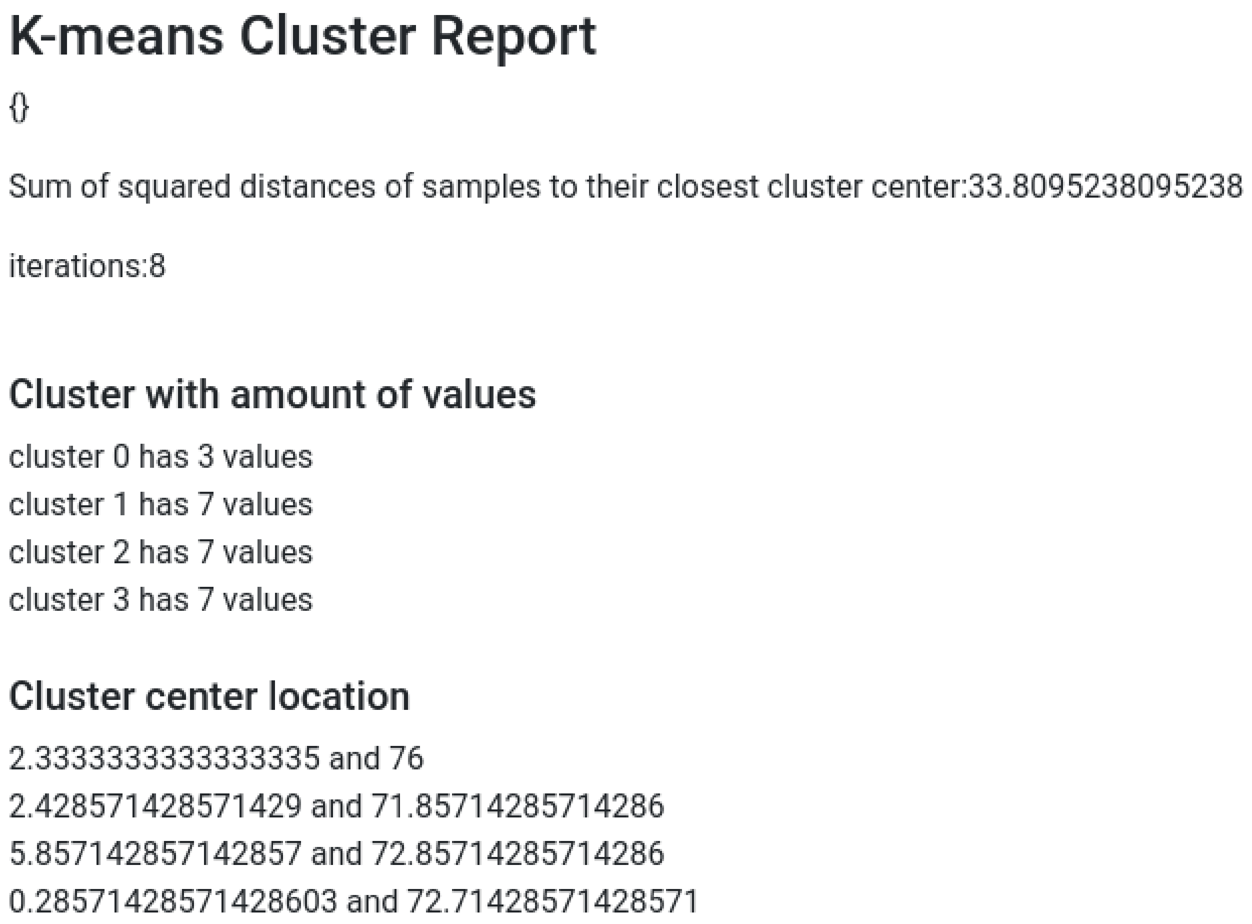
(bool)

scaling

false

(bool)

Figure 22: K-means report page 1



K-means Cluster Report

{}

Sum of squared distances of samples to their closest cluster center:33.8095238095238

iterations:8

Cluster with amount of values

cluster 0 has 3 values
cluster 1 has 7 values
cluster 2 has 7 values
cluster 3 has 7 values

Cluster center location

2.3333333333333335 and 76
2.428571428571429 and 71.85714285714286
5.857142857142857 and 72.85714285714286
0.28571428571428603 and 72.71428571428571

Figure 23: K-means report page 2

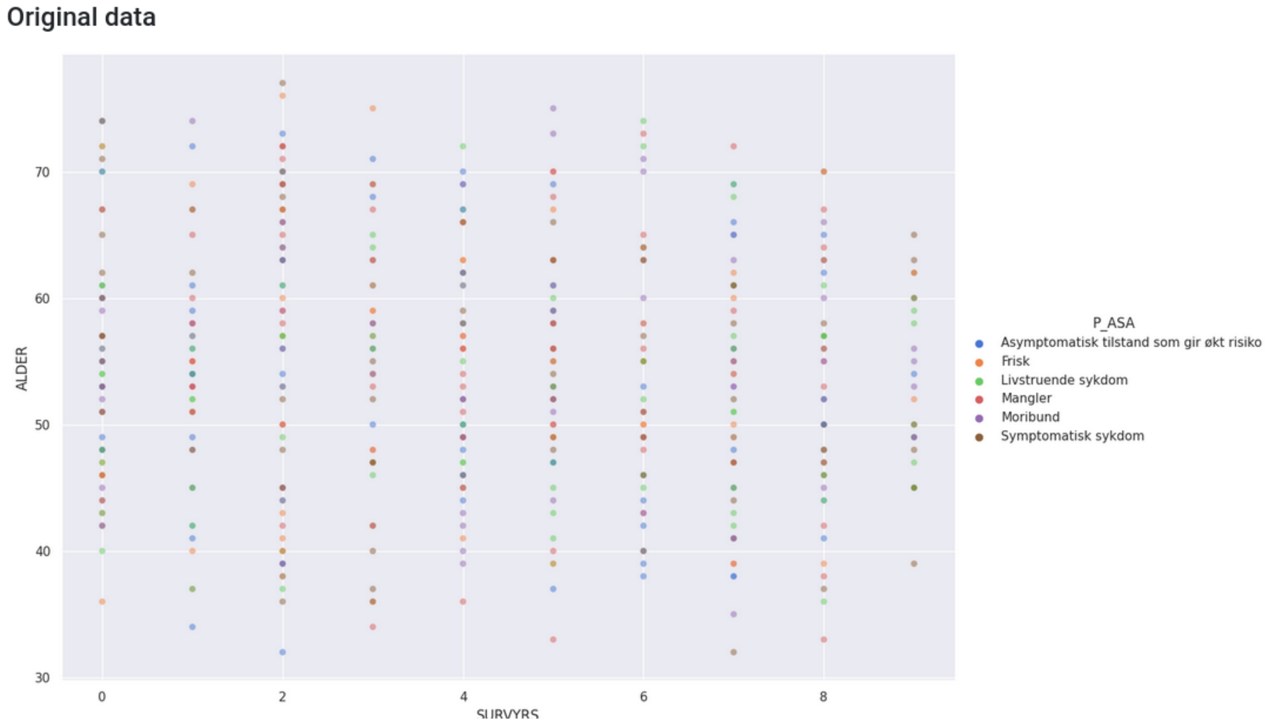


Figure 24: K-means report page 3

11.6 Fulfilling the purpose of the Artifact

At the meeting with the NAR, I was encouraged to attempt and replicate results from Gjøtesen (2013). Gjøtesen (2013) uses Kaplan Meier and Cox PH regression to assess the performance of some prostheses shown to have increased rates of aseptic tibia loosening. Furthermore, we were informed that both of these methods are the standard in registry based research.

This led to the conclusion that these methods much used by the NAR and that it is important to include them in the Artifact.

We were also informed that these methods are used annually by the NAR to monitor and detect underperforming prostheses. NAR receives updates regularly of patients that are undergoing treatment. Therefore they could, in theory, have looked for underperforming prostheses more often. The primary goal for the NAR is to detect underperforming prostheses as early as possible. As such, the registry could potentially benefit from performing Kaplan Meier and Cox PH on a regular basis rather than once a year. In turn, this could allow them to detect underperformers sooner. Bad prostheses should be removed from the Norwegian market which would increase the safety of patients and maintain the quality of the available prostheses.

How can we empower the NAR to run those analyses more often? Obviously, the solution is to simplify the process of running data analysis for detecting underperforming prostheses. What we have done is making various data analysis methods easily available through a user-friendly web-based artifact. Using the web-based Artifact, allows the user to perform data analysis such as Kaplan Meier and Cox PH analysis independently and as often as they need it. In addition to this, it is also possible with little extra programming work to connect the Artifact directly to the NARs database that always has updated patient data.

Compared to traditional statistical programs, the Artifact also takes care of a lot of data preprocessing automatically. For example, categorical variables are automatically handled, and numerical features can be scaled according to the choice of the user. transform textual columns to numbers, remove missing values when the method requires it.

All of the above are time savers when doing data analysis that also help to improve its completeness and accuracy.

In summary, the Artifact can save time and simplify the process of doing analysis on the NAR registry data in several ways:

1. It can easily perform various data analysis methods.
2. It can be directly connected to the NARs database with updated patient data with little extra programming.
3. The Artifact takes care of a lot of data preprocessing automatically.
4. The Artifact is Web based, and can therefore be accessible everywhere.

12 Evaluation

The functional prototype consisting of a front-end in Flask and a back-end in FastAPI has been evaluated. In the evaluation, the back-end team showed different videos of the functional prototype. The videos demonstrated how to run different types of data analysis in the prototype. During each video, we also received comments and questions from the evaluators.

After the videos, all participants were asked to complete a System Usability Scale (SUS) questionnaire to evaluate the functional prototype. The SUS questionnaire consists of ten questions that range from strongly disagree to strongly agree. After the SUS questionnaire is answered, the answers are transformed into a score between 0 and 100. A score of 68 indicates that the system is above average, and below 68 indicates it is below average in terms of system usability. By looking at the score, it is possible to see how usable the system is (*System Usability Scale (SUS)* 2013).

Regarding the content of the back-end and front-end, a questionnaire was issued to assess how evaluators agreed with the content of the final system. A Likert scale was used to acknowledge the level of agreement (disagreeable on the left and agreeable to the right of the scale). The questions from the questionnaire are shown in Section 12.4.

This section includes the most important feedback and the results from all the SUS questionnaires.

12.1 Group 1

Evaluators in Group 1 consisted of two biomedical engineers.

The presented models were interaction plot, K-means clustering, logistic regression, Kaplan Meier, and Weibull plot. The most important comments from this interview are listed below

One expert commented that it would be an advantage to directly connect the prototype to an SQL database instead of reading the data from a file. Additionally, he commented that the prototype is really good and that it has built-in survival analysis methods that are actually used at the Norwegian Arthroplasty Register. Compared to statistical software, the prototype is ”slicker” in the process of performing different methods. Additionally, he commented that the NAR register only uses Kaplan Meier and Cox PH analysis and that this analysis is only performed once each year on the NAR register.

One of the experts commented that ”it’s very nice what you have done”. But he also said that it would also be nice to implement Principal Component Analysis into the prototype.

Afterward, the experts completed a SUS questionnaire for the back-end team. The SUS scores from the two experts were 87.5 and 92.5, which averages 90.

12.2 Group 2

Evaluators consisted of multiple IT experts with varying work experience (from two to six years experience).

The models that were presented included interaction plot, logistic regression, Kaplan Meier, and Weibull plot (*KDD process* 2021). The most important comments from this interview are listed below.

One HCI expert noticed that the interface mixed between Norwegian and English and pointed out that it is important to have a consistent language. The reason for this inconsistency was that the variables in the dataset are in Norwegian, and the prototype is in English.

Furthermore, one expert suggested we include a summary describing the dataset, such as the distribution of values within variables, the portion of missing values, and other descriptions of the data.

One of the HCI experts noticed that a label was missing from the y-axis of a survival plot. The expert argued that although expert users may be aware of what the axis means, some users might experience it negatively.

Another HCI expert commented that the colors in the charts lacked consistency and suggested that we maintain the same color scheme for all graphs in the system. She further argued that the user might find the system confusing and unintuitive when colors are used inconsistently.

One IT expert commented that the prototype looked like the microdata system, which is an open system for data, where it is possible to choose certain variables and run machine learning methods like logistic regression.

Another IT expert commented that the strength of the prototype was that it was very interactive and offered users the possibilities to select the variables to run the analysis on.

An IT expert also commented that it would be nice to test the system in real life.

Afterward, the IT experts completed a SUS questionnaire for the back-end team. The SUS scores from the experts were 90, 80, and 80, which averages 83.

12.3 SUS scores

The SUS scores from groups 1 and 2 for the back-end team are shown in Figure 27. The average SUS score for all particiants are 86, which is considered excellent.

The prototype was well received by positive comments and excellent SUS scores, but there is still room for improvement. Some evaluators pointed out a few minor usability issues. Others had suggestions for additional functionality. The functionalities that was requested were PCA analysis and a summary statistics page describing the dataset. Additionally, it was mentioned that the system would be better if it was directly connected to the SQL database that stored the data instead of us reading the data from a file.

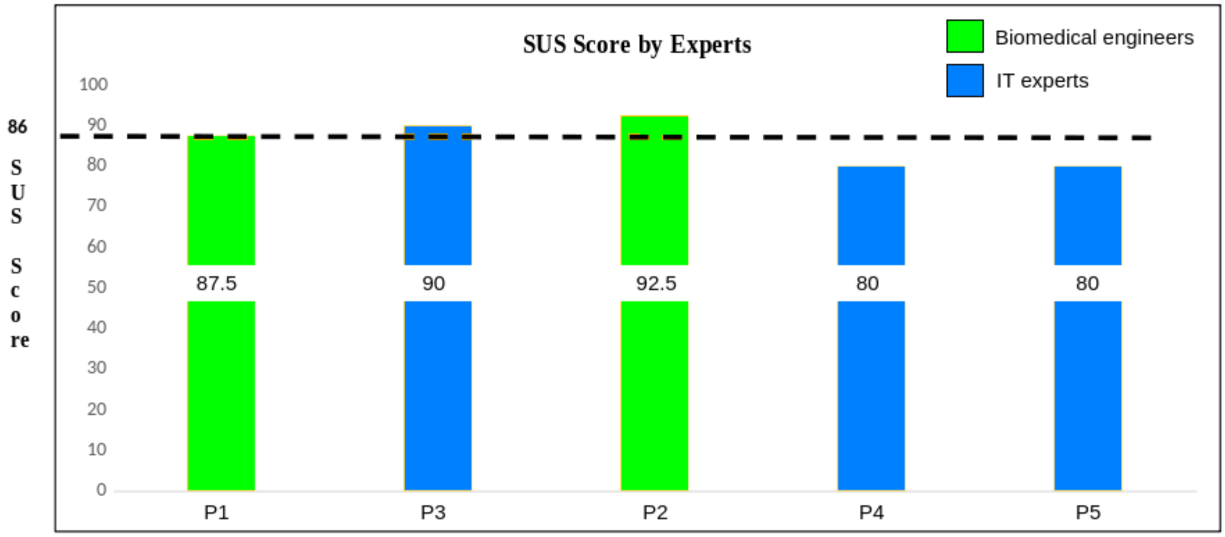
The SUS questionaire with answers are shown in Appendix G 16.

12.4 Content evaluation table

An evaluation questionnaire was developed to evaluate the content of the back- and front-end system combined. Since the users are expected to work with the whole system, the content was seen from the user-perspective. In parts it addressed the data mining methods and how to present their results, and also it assesses components of the visualization as well as the UI. The questionnaire was based on ten components to make it as concise as possible. It also offered a possibility to offer comments and suggestions.

A Likert-scale starting from 'Totally Disagree' to 'Totally Agree' was used to quantify the value for each component. Listed below are the questions in the questionnaire:

1. Choice of Data Mining (DM) tasks
2. Need to add additional tasks
3. Welcoming starting page with something like demographics
4. Save all DM sessions
5. Choice of visualization
6. Level of interactivity in visualization



Average SUS score = 86

Figure 25: SUS scores

7. HCI outlay is satisfying
8. Need to add additional HCI functionality
9. HCI interface is well suited for experts
10. HCI interface has potential to meet patient needs

The questionnaire were answered by ten evaluators. The most relevant questions and answers regarding the back-end choice of data mining tasks are presented in this section.

Answers regarding the question ”choice of data mining tasks” can be seen in Figure x. the majority of 80% were agreeable.

The answers regarding the question ”Need to add additional tasks” are presented in Figure x. The answers were quite evenly distributed, 50% who agree with the question and 30% disagreeing. Those who wanted additional tasks were mostly biomedical experts.

There were also questions regarding the HCI aspects. In response to the question ”HCI is well suited for experts”, 100% agreed. In the next question ”HCI interface has potential to meet patient needs”, 80% were agreeable, the rest were neutral. These questions were important since the back-end tasks will be accessible via a HCI layer. Therefore it was useful to test how the potential user felt about this aspect of development.

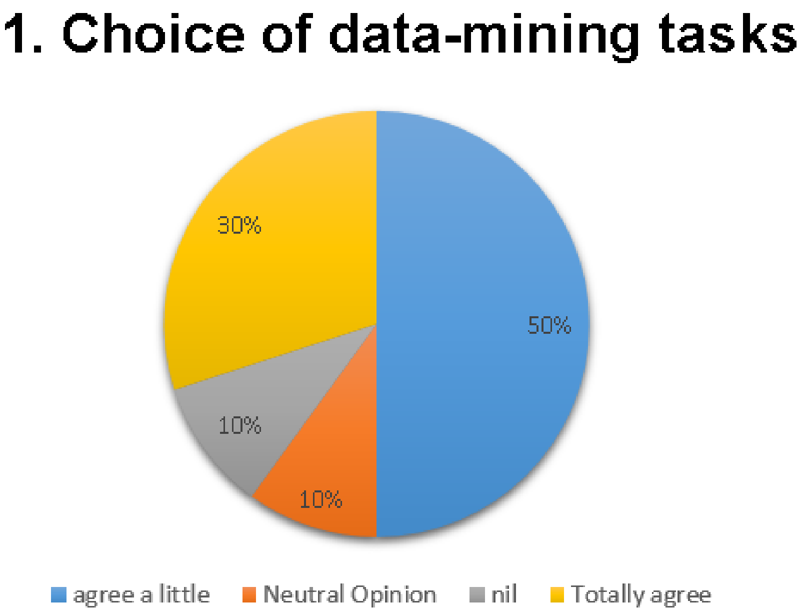


Figure 26: Question 1

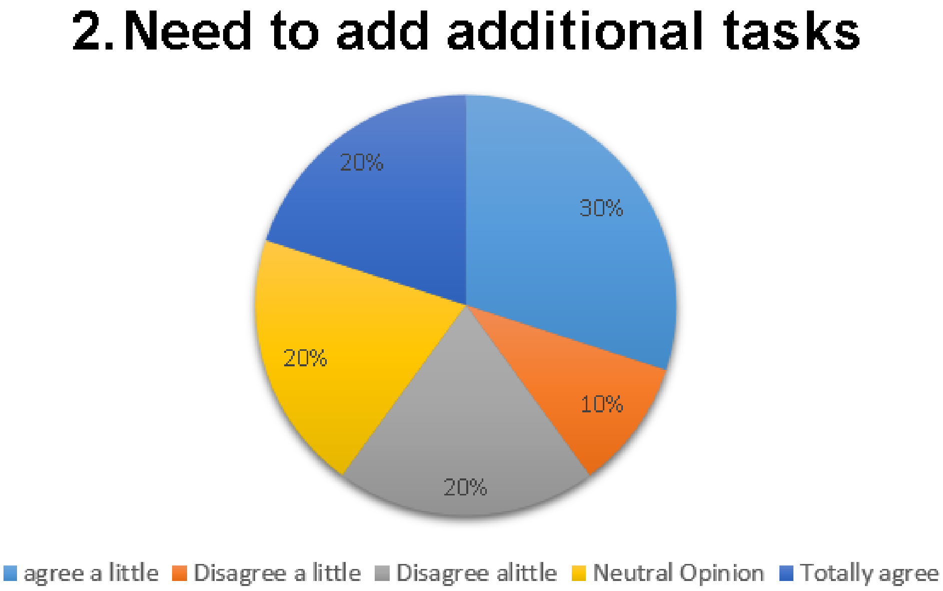


Figure 27: Question 2

13 Discussion

This section answers the research questions, and discusses methodologies and methods used, the back-end development and limitations.

Answering research questions

The three research questions will be discussed here.

RQ1: Is it possible to develop a minimal web-based functional prototype that utilizes data-mining to assist medical professionals making decisions regarding total knee arthroplasty?

We have successfully managed to create a fully functioning prototype to help physicians answer questions regarding underperforming prostheses in total knee arthroplasty. The implementation included back-end procedures for a number of data analytical methods. A simple interface was also developed to carry out work, commonly seen in data-mining. That meant choosing data from a certain period of time, specific problems and patient selections which all helps to demonstrate the feasibility of the prototype. This kind of work gives the user a greater flexibility of exploring data and more independently than what is seen when working with the registry data. The user could minimise dependence on the statistician after working with the prototype for a while.

The development of the artifact is described in Chapter 10.

The evaluation in Chapter 12 has suggested that the choice of implemented methods and how they are presented were appreciated. Some suggestions were made to provide additional functionality.

RQ2: Which data-mining algorithms have the highest potential for predicting underperforming total knee prostheses?

As expected by the staff from the NAR registry, good results were obtained using Kaplan Meier and COX PH analysis. The prototype has however enabled interactivity with data selection, which is not commonly seen in traditional statistical software.
The user might benefit from performing several data mining sessions, in which special patient selections were made and some implants were compared for a particular period of time, or with respect to risk and other factors collected in the registry. It could be an advantage to start with performing well known procedures, and then gradually move to additional features. In Section 6.3.1 and 6.3.5 of Chapter 6.3 we describe the Kaplan Meier and Cox PH method, respectively.

Regarding the evaluation feedback, there are differences. Further details from the evaluation can be found in Chapter 12.

RQ3: Is survival analysis, as the gold standard, the best predictor of under-performing total knee prostheses?

Survival analysis is certainly the most recognized and accepted. Due to the capacity to inspect the time period, it has been adopted as a golden standard. The duration of prostheses is mainly expressed in terms of time, and used for comparing different prostheses and their performance. The traditional approach to detecting underperforming prostesis are explain in Section 6.3.1 and 6.3.5. This fact could be used when considering other data analytical methods. If they can perform as good or better as the golden standard, then they should be considered for future analysis. As this research has shown, it is possible to implement many methods, and further assess them. There are also more clinical questions connected to underperforming prostheses for which additional methods could provide better and more accurate results. Having the prototype in place, it was made possible to take such direction in the future development. In Section 14.2 of Chapter

14, more ideas are given about adding more functionality to the functional prototype.

Study limitations

One of the limitations in the study is the lack of real data. The provided experimental data for analysis consists of a dataset of 1000 records of fictive patients. The registry data contains data such as patient IDs, additional medical data such as real previous surgeries, real health issues, real county or region. If more data were available it could be expected to obtain realistic outcomes and with that meaningful clinical interpretation.

Even though the data could be more realistic, the project's primary purpose was still to build a data mining system for arthroplasty and present its methods. This task has still been achieved using experimental data. However, if the data was real, we could have focused more on the analysis of the results, and we might have found something of interest.

Another study limitation is that we have had limited contact with the staff at arthroplasty registry. They have indeed guided us in implementing the Kaplan Meier and Cox PH analyses methods, and we have talked to them a few times. However, ideally, closer collaboration with experts might have given us better and more user-friendly artifacts.

Another study limitation is that we could not have user-testing of the artifacts because of the Corona situation. It made such user-testing impossible since we could not meet people such as physicians in their real environment. Therefore we had to settle with video recordings of the artifacts being presented over Zoom, which does not give the full proper feeling of using the artifact in real life.

14 Conclusion and future work

14.1 Conclusion

Creating something relevant to the user is obviously the most important thing when creating a system for solving problems in real life.

Making something relevant for the user is mentioned in different words among Design Science, KDD, and the DSDM method. In Design Science, this part is mentioned in the guideline ”Problem relevance.” In KDD, it is emphasized in ”Domain understanding and KDD goals.” Lastly, in the DSDM method the principle 4 says ”Fitness for business purpose is the essential criterion for acceptance of delivery.”.

The NAR emphasized the importance of detecting underperforming prostheses. Analysis for this purpose are mainly Kaplan Meier and Cox PH analysis. Based on this information, we have implemented methods to detect underperforming prostheses using Kaplan Meier and Cox PH analysis in a fully functional prototype. Based on the feedback from the evaluation, it is highly likely that we have managed to create something that the user actually would like to use. Therefore, the project managed to meet many of the user requirements.

By implementing the most important methods which users actually rely on in real life, Kaplan Meier and Cox PH, we could find solutions based on data-mining to identify underperforming prostheses. This in turn could help understand the potential of the system, and open possibilities for including additional methods. Some of them are already implemented and left for users to explore. Sometimes users are unaware of all the methods they could apply, but having a functional prototype could give them an opportunity to test other methods than those they considered to be gold standard. Hopefully novel methods will gain appeal, and could contribute to clinical research.

14.2 Future work

First and foremost, for the future work, the group should include the visualizations of Solheim, 2021 and the design of Stolt-Nielsen, 2021 into a functional prototype. This could be beneficial and appealing to different user groups. The visualizations of Solheim, 2021, would also improve the insight into the underlying data and functioning of methods. Thus the future direction is to include more user centered development and with that also further evaluation and new development iterations.

More methods could be implemented, now that we have a functioning prototype. From the evaluation, we were advised to implement a summary statistics page, PCA analysis and improve a few minor usability issues.

On a personal note regarding the back-end, my preference would be to implement R functionality within the API. The R language seems to be more mature than Python with regards to machine learning and multivariate statistics. Additionally, the R language also seems to offer more simplistic solutions to implementing machine learning and statistics than Python. In addition, one of the experts informed us that the registry is quite particular about the presentation of visualizations in their reports and that these visualizations are made in R. One reason for this might be that Python is a language built to do almost anything, while R, on the other hand, is more specialized towards statistical analysis. However, it is possible to combine these programming languages in a back-end system using the Python library rpy2. Due to time constraints of this project, this has become future work.

If our system is further developed and integrated into the NAR register, it could potentially lead to better clinical outcomes for patients.

15 Definitions

The following terms are used in this thesis:

A **knee prosthesis** is an artificial component that mimics the function of the knee (Shiel, 2019).

A **Web API** is a type of Application Programming Interface (API) that exposes a set of publicly available methods over a communication network. (*What is an API?* 2021).

References

[1] Taiwo Ayodele. "Introduction to Machine Learning". In: *New Advances in Machine Learning* (Feb. 2010). DOI: 10.5772/9394.

[2] Mona Badawy et al. "Hospital volume and the risk of revision in Oxford unicompartmental knee arthroplasty in the Nordic countries -an observational study of 14, 496 cases". In: *BMC Musculoskeletal Disorders* 18.1 (Sept. 2017), pp. 1–8. DOI: 10.1186/s12891-017-1750-7. URL: https://doi.org/10.1186/s12891-017-1750-7.

[3] Helen Barraclough, Lorinda Simms, and Ramaswamy Govindan. "Biostatistics Primer What a Clinician Ought to Know: Hazard Ratios". In: *Journal of thoracic oncology : official publication of the International Association for the Study of Lung Cancer* 6 (June 2011), pp. 978–82. DOI: 10.1097/JTO.0b013e31821b10ab.

[4] P Beynon-Davies et al. "Rapid application development (RAD): an empirical review". In: *European Journal of Information Systems* 8.3 (1999), pp. 211–223. DOI: 10.1057/palgrave.ejis.3000325.

[5] J Martin Bland and Douglas G Altman. "The logrank test". In: *BMJ* 328.7447 (2004), p. 1073. ISSN: 0959-8138. DOI: 10.1136/bmj.328.7447.1073. eprint: https://www.bmj.com/content/328/7447/1073.full.pdf. URL: https://www.bmj.com/content/328/7447/1073.

[6] Louis Boyce et al. "The outcomes of total knee arthroplasty in morbidly obese patients: a systematic review of the literature". In: *Archives of Orthopaedic and Trauma Surgery* 139.4 (Feb. 2019), pp. 553–560. DOI: 10.1007/s00402-019-03127-5. URL: https://doi.org/10.1007/s00402-019-03127-5.

[7] Thomas E. Brown, Benjamin L. Harper, and Kristian Bjorgul. "Comparison of Cemented and Uncemented Fixation in Total Knee Arthroplasty". In: *Orthopedics* 36.5 (Jan. 2013), pp. 380–387. DOI: 10.3928/01477447-20130426-10.

[8] *C-statistics: Definition, Examples, Weighting and Significance*. 2021. URL: https://www.statisticshowto.com/c-statistic/ (visited on 06/09/2021).

[9] *Chi-Square probabilities table*. 2021. URL: https://people.richland.edu/james/lecture/m170/tbl-chi.html (visited on 04/08/2021).

[10] *Cox Regression Analysis - IBM Documentation*. 2021. URL: https://www.ibm.com/docs/en/spss-statistics/SaaS?topic=statistics-cox-regression-analysis (visited on 06/09/2021).

[11] *CoxPHFitter lifelines*. 2021. URL: https://lifelines.readthedocs.io/en/latest/fitters/regression/CoxPHFitter.html (visited on 06/09/2021).

[12] Cameron Davidson-Pilon. "lifelines: survival analysis in Python". In: *Journal of Open Source Software* 4 (Aug. 2019), p. 1317. DOI: 10.21105/joss.01317.

[13] *Difference Between GET and POST Method in HTML (with Comparison Chart and Examples)*. URL: https://techdifferences.com/difference-between-get-and-post-method-in-html.html (visited on 03/28/2021).

[14] Aline Dresch, Daniel Pacheco Lacerda, and Antonio Valle Antunes Jose. *Design science research: a method for science and technology advancement*. Springer, 2015.

[15] Aline Dresch, Daniel Pacheco Lacerda, and Antonio Valle Antunes Josè. *Design science research: a method for science and technology advancement*. Springer, 2015.

[16] Arihito Endo, Takeo Shibata, and Hiroshi Tanaka. "Comparison of Seven Algorithms to Predict Breast Cancer Survival". In: *Comparison of Seven Algorithms to Predict Breast Cancer Survival* (Nov. 2007), pp. 11–16. URL: https://www.researchgate.net/publication/255591934_Comparison_of_Seven_Algorithms_to_Predict_Breast_Cancer_Survival.

[17] Ørjan Ertkjern. "Postmarket Surveillance of Orthopaedic Implants using Web-technologies". Master Thesis. The University of Bergen, June 2015.

[18] Jonathan T Evans et al. "How long does a knee replacement last? A systematic review and meta-analysis of case series and national registry reports with more than 15 years of follow-up". In: *The Lancet* 393.10172 (2019), pp. 655–663. DOI: 10.1016/s0140-6736(18)32531-5.

[19] *FAR Finnish Arthroplasty Register*. URL: https://www.thl.fi/far/#index (visited on 03/28/2021).

[20] *FastAPI*. 2021. URL: https://fastapi.tiangolo.com/ (visited on 04/06/2021).

[21] *Flask Documentation*. 2021. URL: https://flask.palletsprojects.com/en/2.0.x/ (visited on 06/09/2021).

[22] Anders El-Galaly et al. "Can Machine-learning Algorithms Predict Early Revision TKA in the Danish Knee Arthroplasty Registry?" In: *Clinical Orthopaedics and Related Research* (June 2020). DOI: 10.1097/CORR.0000000000001343.

[23] Chance E. Gartneer. *How to Find Euclidean Distance*. Dec. 2020. URL: https://sciencing.com/euclidean-distance-7829754.html (visited on 04/13/2021).

[24] *Getting started with Django*. 2021. URL: https://www.djangoproject.com/start/ (visited on 05/23/2021).

[25] *ggplot2*. 2021. URL: https://ggplot2.tidyverse.org/ (visited on 05/16/2021).

[26] *GitHub Desktop: Simple collaboration from your desktop*. 2021. URL: https://desktop.github.com/ (visited on 05/30/2021).

[27] *GitHub: Build like the best teams on the planet*. 2021. URL: https://github.com/team (visited on 05/30/2021).

[28] Gjøtesen. *Survival rates and causes of revision in cemented primary total knee replacement*. 2013. DOI: 10.1302/0301-620X.95B5.30271.

[29] Manish Goel, Pardeep Khanna, and Jugal Kishore. "Understanding survival analysis: Kaplan-Meier estimate". In: *International journal of Ayurveda research* 1 (Oct. 2010), pp. 274–8. DOI: 10.4103/0974-7788.76794.

[30] Karimollah Hajian-Tilaki. "Receiver Operating Characteristic (ROC) Curve Analysis for Medical Diagnostic Test Evaluation". In: *Caspian journal of internal medicine* 4 (Sept. 2013), pp. 627–635.

[31] Bob Hayes. *Programming Languages Most Used and Recommended by Data Scientists*. 2021. URL: http://businessoverbroadway.com/2019/01/13/programming-languages-most-used-and-recommended-by-data-scientists/ (visited on 03/18/2021).

[32] Healio. *What is arthroplasty?* May 2012. URL: https://www.healio.com/orthopedics/news/online/%7B2cc2d4a7-903b-4ff5-8009-cfdf91de5470%7D/what-is-arthroplasty.

[33] Alan Hefner et al. *(PDF) Design Science in Information Systems Research*. Mar. 2004. URL: https://www.researchgate.net/publication/201168946_Design_Science_in_Information_Systems_Research.

[34] Kimberly Holland. *Understanding Cartilage, Joints, and the Aging Process*. May 2018. URL: https://www.healthline.com/health/osteoarthritis/understanding-aging-and-joints.

[35] Andreas Iden. "Data Mining Approach to Modelling of Outcomes in Total Knee Arthroplasty". Master thesis. The University of Bergen, June 2020.

[36] Kitty Jager et al. "The analysis of survival data: The Kaplan-Meier method". In: *Kidney international* 74 (Aug. 2008), pp. 560–5. DOI: 10.1038/ki.2008.217.

[37] *jinja documentation*. 2021. URL: https://jinja.palletsprojects.com/en/3.0.x/ (visited on 06/09/2021).

[38] *kaplan+meier - Search Results - PubMed*. 2021. URL: https://pubmed.ncbi.nlm.nih.gov/?term=kaplan%2Bmeier (visited on 04/08/2021).

[39] *KDD process*. 2021. URL: https://lifelines.readthedocs.io/en/latest/fitters/univariate/WeibullFitter.html.

[40] *KDD Process in Data Mining*. 2021. URL: https://www.javatpoint.com/kdd-process-in-data-mining (visited on 06/09/2021).

[41] *Knee replacement*. Dec. 2017. URL: https://www.mayoclinic.org/tests-procedures/knee-replacement/about/pac-20385276.

[42] Jacek Kopecký, Paul Fremantle, and Rich Boakes. "A history and future of Web APIs". In: *it - Information Technology* 56 (Jan. 2014). DOI: 10.1515/itit-2013-1035.

[43] Sunil Kumar and Ilyoung Chong. "Correlation Analysis to Identify the Effective Data in Machine Learning: Prediction of Depressive Disorder and Emotion States". In: *International Journal of Environmental Research and Public Health* 15.12 (2018), p. 2907. DOI: 10.3390/ijerph15122907.

[44] Tesfaye H Leta et al. *Failure of aseptic revision total knee arthroplasties.* Feb. 2015. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4366664/.

[45] Joshua R. Lewis et al. "A Predictive Model for Knee Joint Replacement in Older Women". In: *PLoS ONE* 8.12 (Nov. 2013). DOI: 10.1371/journal.pone.0083665.

[46] Giulia Lorenzoni et al. "Comparison of Machine Learning Techniques for Prediction of Hospitalization in Heart Failure Patients". In: *Journal of Clinical Medicine* 8.9 (2019), p. 1298. DOI: 10.3390/jcm8091298.

[47] Oded Maimon and Lior Rokach. *Data Mining and Knowledge Discovery Handbook.* Springer US, 2005.

[48] Jane Miller. *Writing about Hazards Analysis for Biomedical and Public Health Audiences.* Jan. 2013. DOI: 10.13140/RG.2.2.35357.10723.

[49] Francis Sahngun Nahm. "What the P values really tell us". eng. In: *The Korean journal of pain* 30.4 (Oct. 2017). PMC5665734[pmcid], pp. 241–242. ISSN: 2005-9159. DOI: 10.3344/kjp.2017.30.4.241. URL: https://doi.org/10.3344/kjp.2017.30.4.241.

[50] *norwegian arthroplasty register.* URL: http://nrlweb.ihelse.net/eng/.

[51] *Norwegian national advisory unit on arthroplasty and hip fractures: annual report.* 2019. URL: http://nrlweb.ihelse.net/eng/Rapporter/Report2019_english.pdf.

[52] *Numpy.* 2021. URL: https://numpy.org/ (visited on 05/13/2021).

[53] Mahamed Omran, Andries Engelbrecht, and Ayed Salman. "An overview of clustering methods". In: *Intell. Data Anal.* 11 (Nov. 2007), pp. 583–605. DOI: 10.3233/IDA-2007-11602.

[54] *Pandas.* 2021. URL: https://pandas.pydata.org/ (visited on 05/13/2021).

[55] *Pandas - Python Data Analysis Library.* 2021. URL: https://pandas.pydata.org/ (visited on 05/24/2021).

[56] Alma B Pedersen and Anne Mari Fenstad. "NORDIC ARTHROPLASTY REGISTER ASSOCIATION (NARA) REPORT". In: (2015), pp. 1–32. URL: http://nrlweb.ihelse.net/NARA_2015_ORIG_ny.pdf.

[57] Fabian Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (Jan. 2012).

[58] Mikko Peltola, Antti Malmivaara, and Mika Paavola. "Learning Curve for New Technology?" In: *The Journal of Bone & Joint Surgery* 95.23 (Dec. 2013), pp. 2097–2103. DOI: 10.2106/jbjs.l.01296. URL: https://doi.org/10.2106/jbjs.l.01296.

[59] Joanne Peng, Kuk Lee, and Gary Ingersoll. "An Introduction to Logistic Regression Analysis and Reporting". In: *Journal of Educational Research - J EDUC RES* 96 (Sept. 2002), pp. 3–14. DOI: 10.1080/00220670209598786.

[60] Postman. *Postman.* https://www.postman.com/. 2021.

[61] Jenny Preece, Yvonne Rogers, and Helen Sharp. *Interaction design: beyond human-computer interaction.* J. Wiley & Sons, 2002.

[62] Shankar Prinja, Nidhi Gupta, and Ramesh Verma. "Censoring in clinical trials: review of survival analysis techniques". In: *Indian journal of community medicine : official publication of Indian Association of Preventive and Social Medicine* (Apr. 2010). URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2940174/.

[63] Otto Robertsson et al. "Knee arthroplasty in Denmark, Norway and Sweden". In: *Acta Orthopaedica* 81.1 (2010), pp. 82–89. DOI: 10.3109/17453671003685442.

[64] *rpy2 - R in Python.* 2021. URL: https://rpy2.github.io/ (visited on 05/15/2021).

[65] Amit Saxena et al. "A Review of Clustering Techniques and Developments". In: *Neurocomputing* 267 (July 2017). DOI: 10.1016/j.neucom.2017.06.053.

[66] scikit. *scikit learn index*. 2021. URL: https://scikit-learn.org/stable/index.html (visited on 03/18/2021).

[67] scikit. *Who is using scikit learn?* 2021. URL: https://scikit-learn.org/stable/testimonials/testimonials.html (visited on 03/18/2021).

[68] Thorsten M Seyler et al. "Sports Activity after Total Hip and Knee Arthroplasty". In: *Sports Medicine* 36.7 (2006), pp. 571–583. DOI: 10.2165/00007256-200636070-00003. URL: https://doi.org/10.2165/00007256-200636070-00003.

[69] William C. Shiel. *Total Knee Replacement Recovery, Surgery Risks Exercises*. Nov. 2019. URL: https://www.medicinenet.com/total_knee_replacement/article.htm.

[70] Ritesh Singh and Keshab Mukhopadhyay. "Survival analysis in clinical trials: Basics and must know areas". In: *Perspectives in clinical research* (Oct. 2011). URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3227332/.

[71] Sudhir Singh and Nasib Singh Gill. *Analysis and Study of K-Means Clustering Algorithm*. July 2013. URL: https://www.ijert.org/analysis-and-study-of-k-means-clustering-algorithm.

[72] Pinky Sodhi, Naman Awasthi, and Vishal Sharma. "Introduction to Machine Learning and Its Basic Application in Python". In: *SSRN Electronic Journal* (2019), pp. 1354–1375. DOI: 10.2139/ssrn.3323796.

[73] Arle Farsund Solheim. "Arthroplasty Data Visualization". Master thesis. The University of Bergen, June 2021.

[74] Sunniva Stolt-Nielsen. "Design Driven Development of a Web-Enabled System for Data Mining in Arthroplasty Registry". Master thesis. The University of Bergen, June 2021.

[75] Shakirat Sulyman. "Client-Server Model". In: *IOSR Journal of Computer Engineering* 16 (Jan. 2014), pp. 57–71. DOI: 10.9790/0661-16195771.

[76] Nidhi Suthar, Indr jeet Rajput, and Vinit kumar Gupta. *A Technical Survey on DBSCAN Clustering Algorithm*. May 2013. URL: https://www.ijser.org/researchpaper/A-Technical-Survey-on-DBSCAN-Clustering-Algorithm.pdf (visited on 04/15/2021).

[77] *System Usability Scale (SUS)*. Sept. 2013. URL: https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html.

[78] Magdalena Szumilas. "Explaining Odds Ratio". In: *Journal of the Canadian Academy of Child and Adolescent Psychiatry = Journal de l'Académie canadienne de psychiatrie de l'enfant et de l'adolescent* 19 (Aug. 2010), pp. 227–229. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2938757/.

[79] Knut T.Hufthammer. "Data Mining For Outcome Analysis In Hip Arthroplasty". Master thesis. The University of Bergen, June 2021.

[80] *The Norwegian Arthroplasty Register*. 2021. URL: http://nrlweb.ihelse.net/eng/default_gml.htm (visited on 05/31/2021).

[81] *The R Project for Statistical Computing*. 2021. URL: https://www.r-project.org/ (visited on 05/15/2021).

[82] *Total Knee Replacement*. URL: https://orthoinfo.aaos.org/en/treatment/total-knee-replacement/.

[83] Natalja Voznuka et al. "Report Generation and Data Mining in the Domain of Thoracic Surgery". In: *Journal of Medical Systems* 28.5 (2004), pp. 497–509. DOI: 10.1023/b:joms.0000041176.58311.29.

[84] *What is an API?* 2021. URL: https://www.redhat.com/en/topics/api/what-are-application-programming-interfaces (visited on 06/09/2021).

16 Appendix

Appendix A: Knee implants

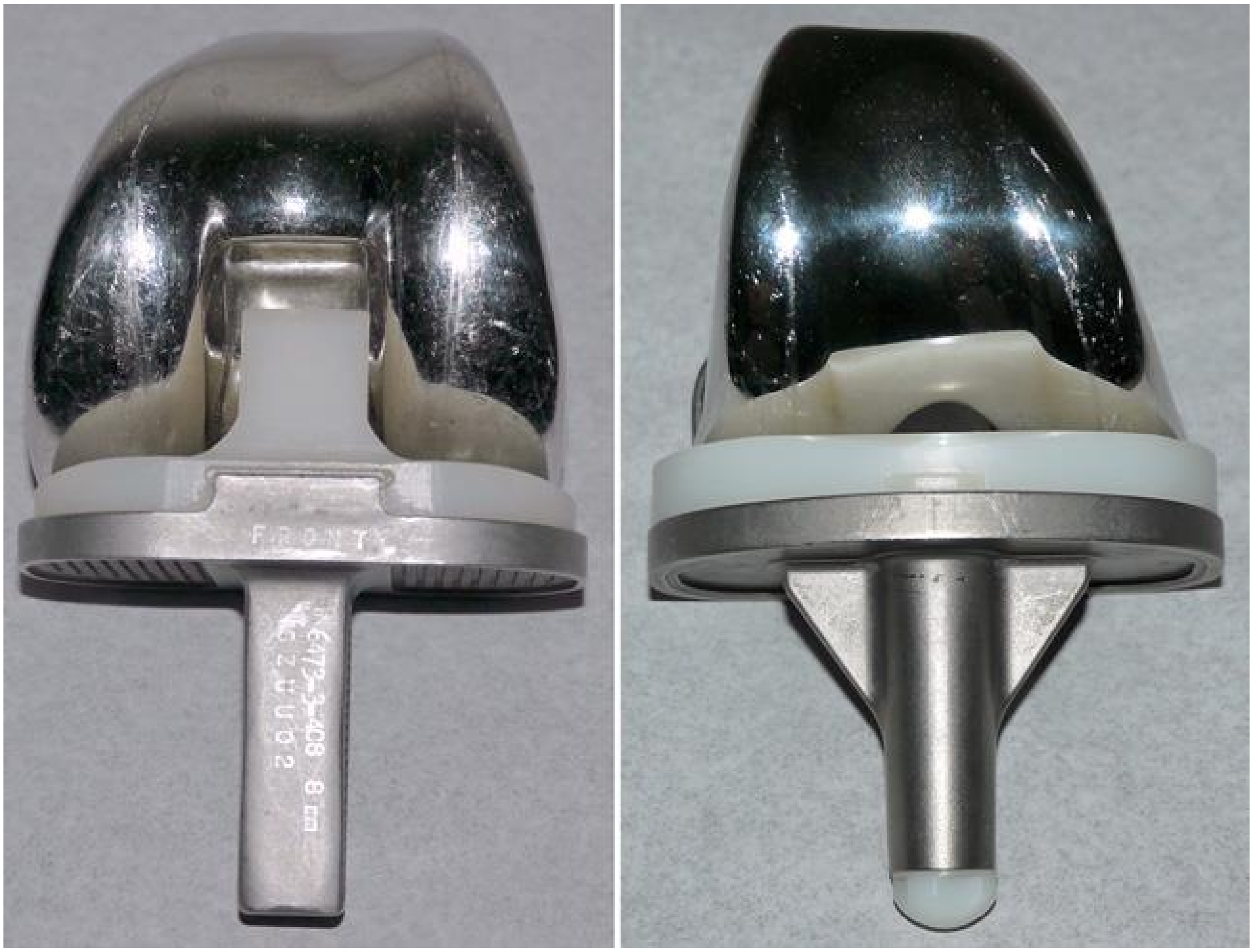


Figure 28: Knee implants (Shiel, 2019)

Appendix B: NSD Approval

NORSK SENTER FOR FORSKNINGSDATA

NSD sin vurdering

Prosjekttittel

Maskinlæring hos norsk register for kneproteser

Referansenummer

902870

Registrert

18.10.2020 av Sølve Ånneland

Behandlingsansvarlig institusjon

Universitetet i Bergen / Det samfunnsvitenskapelige fakultet / Institutt for informasjons- og medievitenskap

Prosjektansvarlig (vitenskapelig ansatt/veileder eller stipendiat)

Ankica Babic, Ankica.Babic@uib.no, tlf: 4755589139

Type prosjekt

Studentprosjekt, masterstudium

Prosjektperiode

05.11.2020 - 15.06.2021

Status

29.10.2020 - Vurdert

Vurdering (1)

29.10.2020 - Vurdert

Det er vår vurdering at behandlingen av personopplysninger i prosjektet vil være i samsvar med personvernlovgivningen så fremt den gjennomføres i tråd med det som er dokumentert i meldeskjemaet med vedlegg den 29.10.2020, samt i meldingsdialogen mellom innmelder og NSD. Behandlingen kan starte.

DEL PROSJEKTET MED PROSJEKTANSVARLIG
Det er obligatorisk for studenter å dele meldeskjemaet med prosjektansvarlig (veileder). Det gjøres ved å trykke på “Del prosjekt” i meldeskjemaet.

MELD VESENTLIGE ENDRINGER

Dersom det skjer vesentlige endringer i behandlingen av personopplysninger, kan det være nødvendig å melde dette til NSD ved å oppdatere meldeskjemaet. Før du melder inn en endring, oppfordrer vi deg til å lese om hvilke type endringer det er nødvendig å melde:

https://nsd.no/personvernombud/meld_prosjekt/meld_endringer.html

Du må vente på svar fra NSD før endringen gjennomføres.

TYPE OPPLYSNINGER OG VARIGHET

Prosjektet vil behandle alminnelige kategorier av personopplysninger frem til 15.06.2021.

LOVLIG GRUNNLAG

Prosjektet vil innhente samtykke fra de registrerte til behandlingen av personopplysninger. Vår vurdering er at prosjektet legger opp til et samtykke i samsvar med kravene i art. 4 og 7, ved at det er en frivillig, spesifikk, informert og utvetydig bekreftelse som kan dokumenteres, og som den registrerte kan trekke tilbake. Lovlig grunnlag for behandlingen vil dermed være den registrertes samtykke, jf. personvernforordningen art. 6 nr. 1 bokstav a.

PERSONVERNPRINSIPPER

NSD vurderer at den planlagte behandlingen av personopplysninger vil følge prinsippene i personvernforordningen om:

- lovlighet, rettferdighet og åpenhet (art. 5.1 a), ved at de registrerte får tilfredsstillende informasjon om og samtykker til behandlingen
- formålsbegrensning (art. 5.1 b), ved at personopplysninger samles inn for spesifikke, uttrykkelig angitte og berettigede formål, og ikke behandles til nye, uforenlige formål
- dataminimering (art. 5.1 c), ved at det kun behandles opplysninger som er adekvate, relevante og nødvendige for formålet med prosjektet
- lagringsbegrensning (art. 5.1 e), ved at personopplysningene ikke lagres lengre enn nødvendig for å oppfylle formålet

DE REGISTRERTES RETTIGHETER

Så lenge de registrerte kan identifiseres i datamaterialet vil de ha følgende rettigheter: åpenhet (art. 12), informasjon (art. 13), innsyn (art. 15), retting (art. 16), sletting (art. 17), begrensning (art. 18), underretning (art. 19), dataportabilitet (art. 20).

NSD vurderer at informasjonen om behandlingen som de registrerte vil motta oppfyller lovens krav til form og innhold, jf. art. 12.1 og art. 13.

Vi minner om at hvis en registrert tar kontakt om sine rettigheter, har behandlingsansvarlig institusjon plikt til å svare innen en måned.

FØLG DIN INSTITUSJONS RETNINGSLINJER

NSD legger til grunn at behandlingen oppfyller kravene i personvernforordningen om riktighet (art. 5.1 d), integritet og konfidensialitet (art. 5.1. f) og sikkerhet (art. 32).

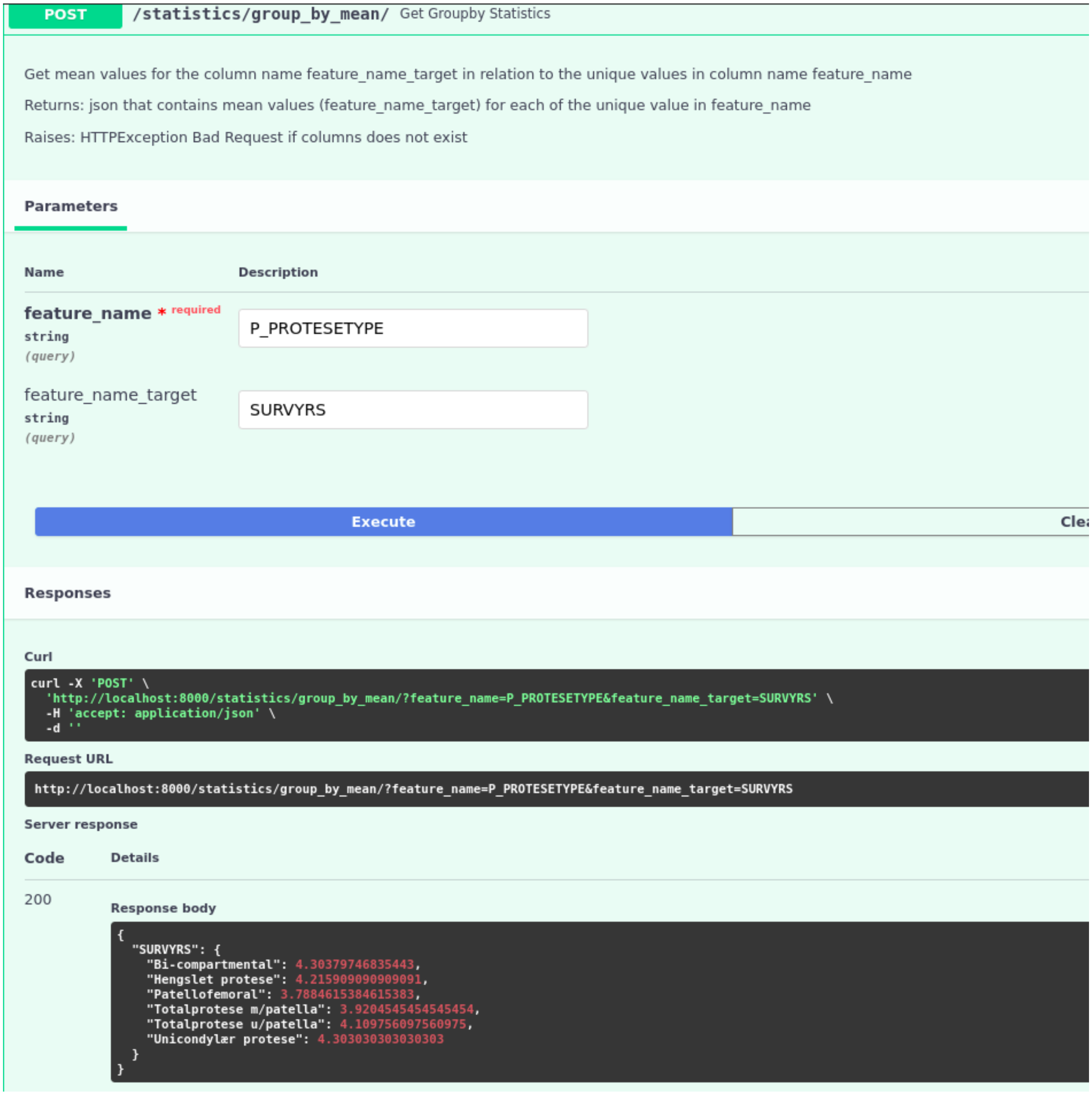
For å forsikre dere om at kravene oppfylles, må dere følge interne retningslinjer og/eller rådføre dere med behandlingsansvarlig institusjon.

OPPFØLGING AV PROSJEKTET
NSD vil følge opp ved planlagt avslutning for å avklare om behandlingen av personopplysningene er avsluttet.

Lykke til med prosjektet!

Tlf. Personverntjenester: 55 58 21 17 (tast 1)

Appendix C: API:average implant survival



/statistics/group_by_mean/ Get Groupby Statistics

Get mean values for the column name feature_name_target in relation to the unique values in column name feature_name

Returns: json that contains mean values (feature_name_target) for each of the unique value in feature_name

Raises: HTTPException Bad Request if columns does not exist

Name	Description
feature_name * required string *(query)*	P_PROTESETYPE
feature_name_target string *(query)*	SURVYRS

Curl

```
curl -X 'POST' \
  'http://localhost:8000/statistics/group_by_mean/?feature_name=P_PROTESETYPE&feature_name_target=SURVYRS' \
  -H 'accept: application/json' \
  -d ''
```

Request URL

```
http://localhost:8000/statistics/group_by_mean/?feature_name=P_PROTESETYPE&feature_name_target=SURVYRS
```

Server response

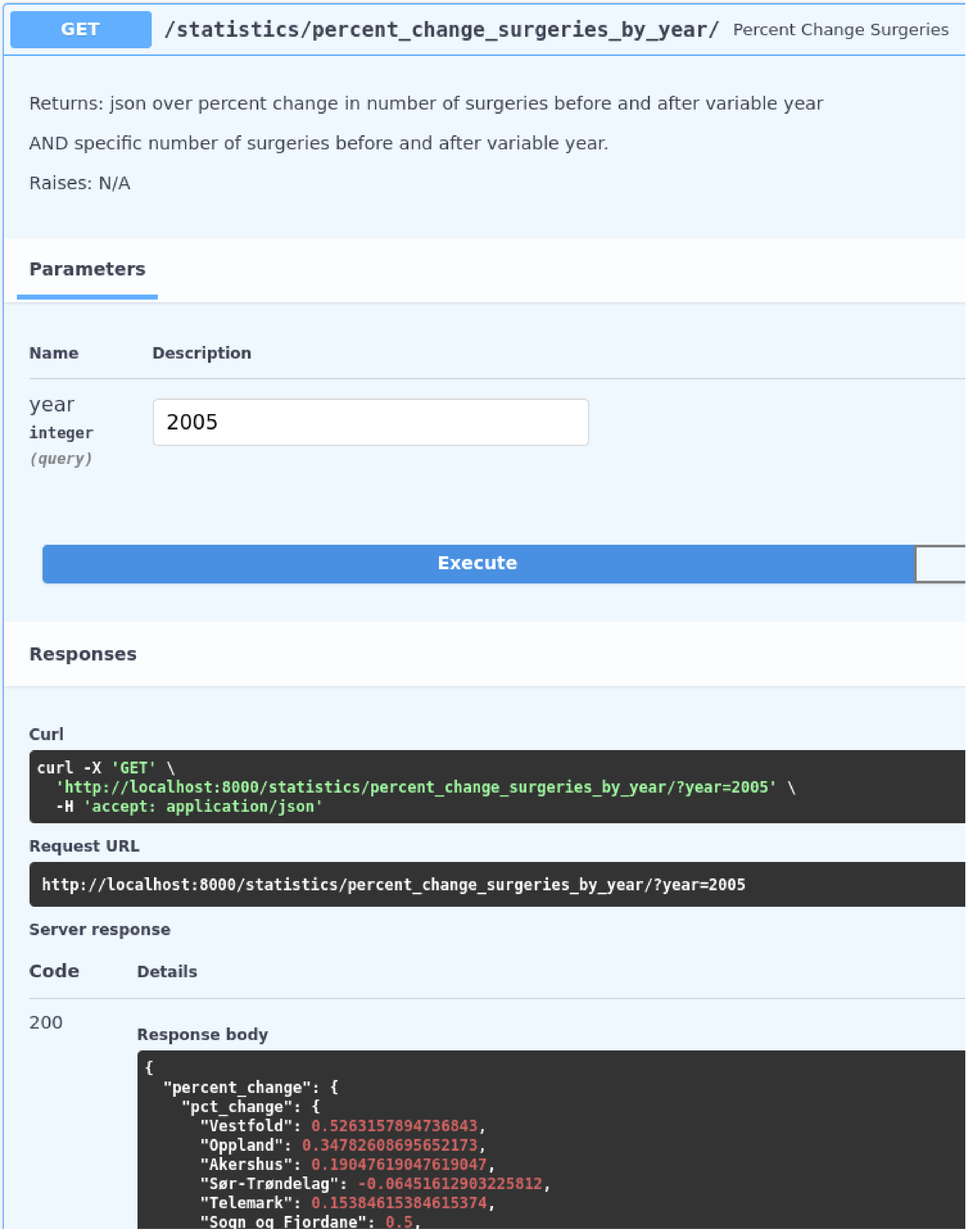
Code Details

200 Response body

```
{
  "SURVYRS": {
    "Bi-compartmental": 4.30379746835443,
    "Hengslet protese": 4.215909090909091,
    "Patellofemoral": 3.7884615384615383,
    "Totalprotese m/patella": 3.9204545454545454,
    "Totalprotese u/patella": 4.109756097560975,
    "Unicondylær protese": 4.303030303030303
  }
}
```

Figure 29: API endpoint to retrieve average implant survival duration

Appendix D: API:map information



/statistics/percent_change_surgeries_by_year/ Percent Change Surgeries

Returns: json over percent change in number of surgeries before and after variable year

AND specific number of surgeries before and after variable year.

Raises: N/A

Name	Description
year integer *(query)*	2005

Curl

```
curl -X 'GET' \
  'http://localhost:8000/statistics/percent_change_surgeries_by_year/?year=2005' \
  -H 'accept: application/json'
```

Request URL

```
http://localhost:8000/statistics/percent_change_surgeries_by_year/?year=2005
```

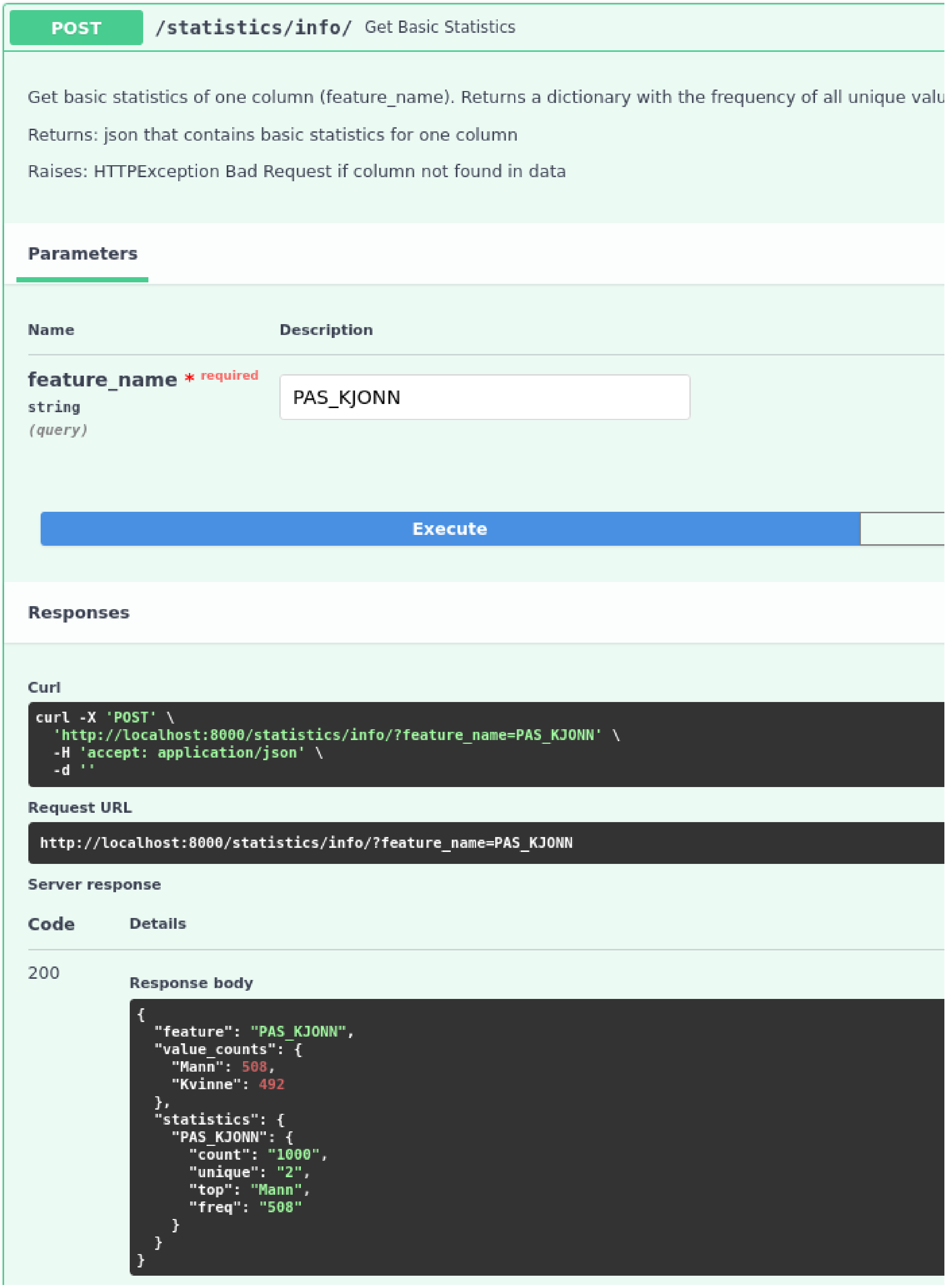
Server response

Code	Details
200	Response body

```
{
  "percent_change": {
    "pct_change": {
      "Vestfold": 0.5263157894736843,
      "Oppland": 0.34782608695652173,
      "Akershus": 0.19047619047619047,
      "Sør-Trøndelag": -0.06451612903225812,
      "Telemark": 0.15384615384615374,
      "Sogn og Fjordane": 0.5,
```

Figure 30: API endpoint to retrieve information that can be plotted on a map. The figure shows percentage change in the number of surgeries before compared to after a user specified year

Appendix E: API:statistical information



/statistics/info/ Get Basic Statistics

Get basic statistics of one column (feature_name). Returns a dictionary with the frequency of all unique valu

Returns: json that contains basic statistics for one column

Raises: HTTPException Bad Request if column not found in data

Parameters

Name	Description
feature_name * required string *(query)*	PAS_KJONN

Responses

Curl

```
curl -X 'POST' \
  'http://localhost:8000/statistics/info/?feature_name=PAS_KJONN' \
  -H 'accept: application/json' \
  -d ''
```

Request URL

```
http://localhost:8000/statistics/info/?feature_name=PAS_KJONN
```

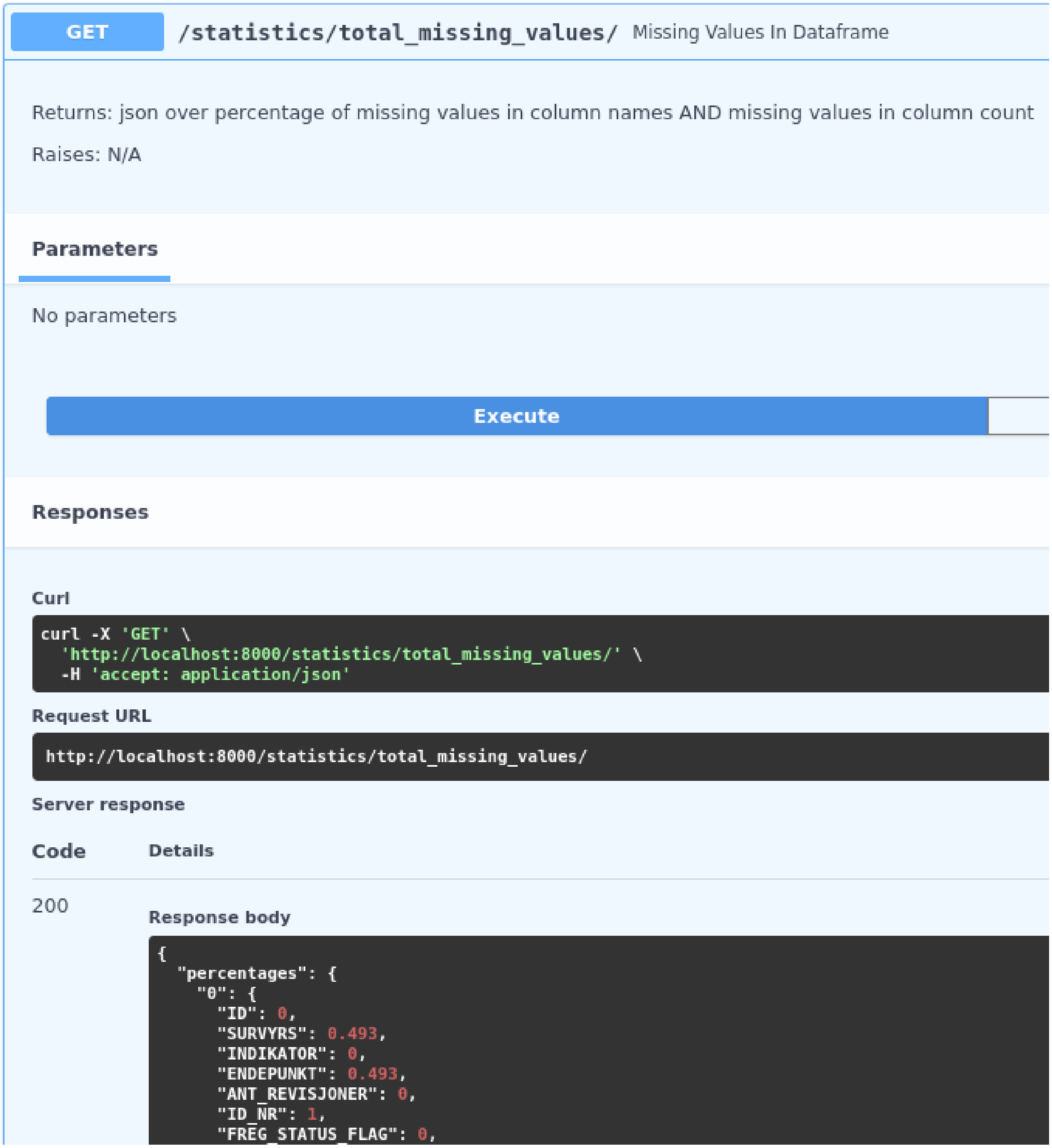
Server response

Code	Details
200	**Response body**

```
{
  "feature": "PAS_KJONN",
  "value_counts": {
    "Mann": 508,
    "Kvinne": 492
  },
  "statistics": {
    "PAS_KJONN": {
      "count": "1000",
      "unique": "2",
      "top": "Mann",
      "freq": "508"
    }
  }
}
```

Figure 31: API endpoint to retrieve statistical information

Appendix F: API:missing values



/statistics/total_missing_values/ Missing Values In Dataframe

Returns: json over percentage of missing values in column names AND missing values in column count

Raises: N/A

No parameters

Curl

```
curl -X 'GET' \
  'http://localhost:8000/statistics/total_missing_values/' \
  -H 'accept: application/json'
```

Request URL

```
http://localhost:8000/statistics/total_missing_values/
```

Server response

Code Details

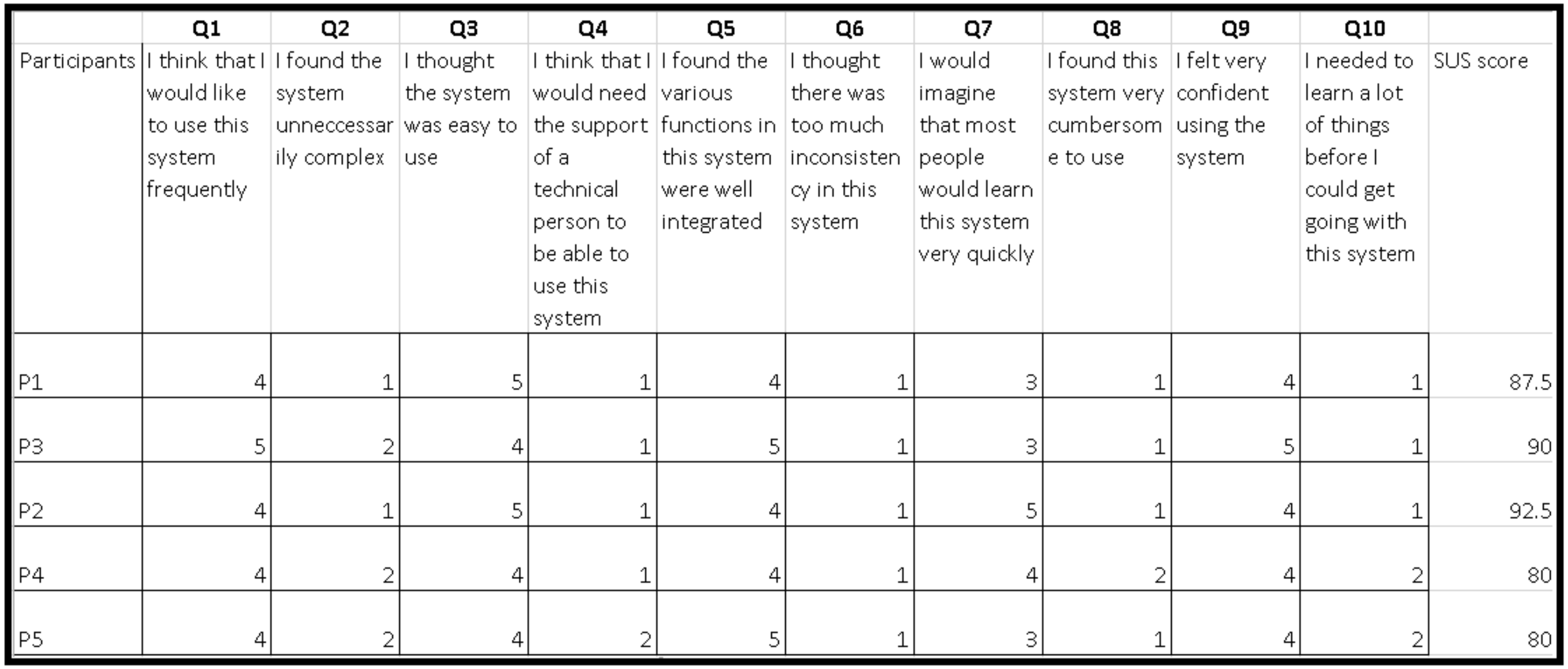
200

Response body

```
{
  "percentages": {
    "0": {
      "ID": 0,
      "SURVYRS": 0.493,
      "INDIKATOR": 0,
      "ENDEPUNKT": 0.493,
      "ANT_REVISJONER": 0,
      "ID_NR": 1,
      "FREG_STATUS_FLAG": 0,
```

Figure 32: API endpoint to retrieve the percentage and number of missing values

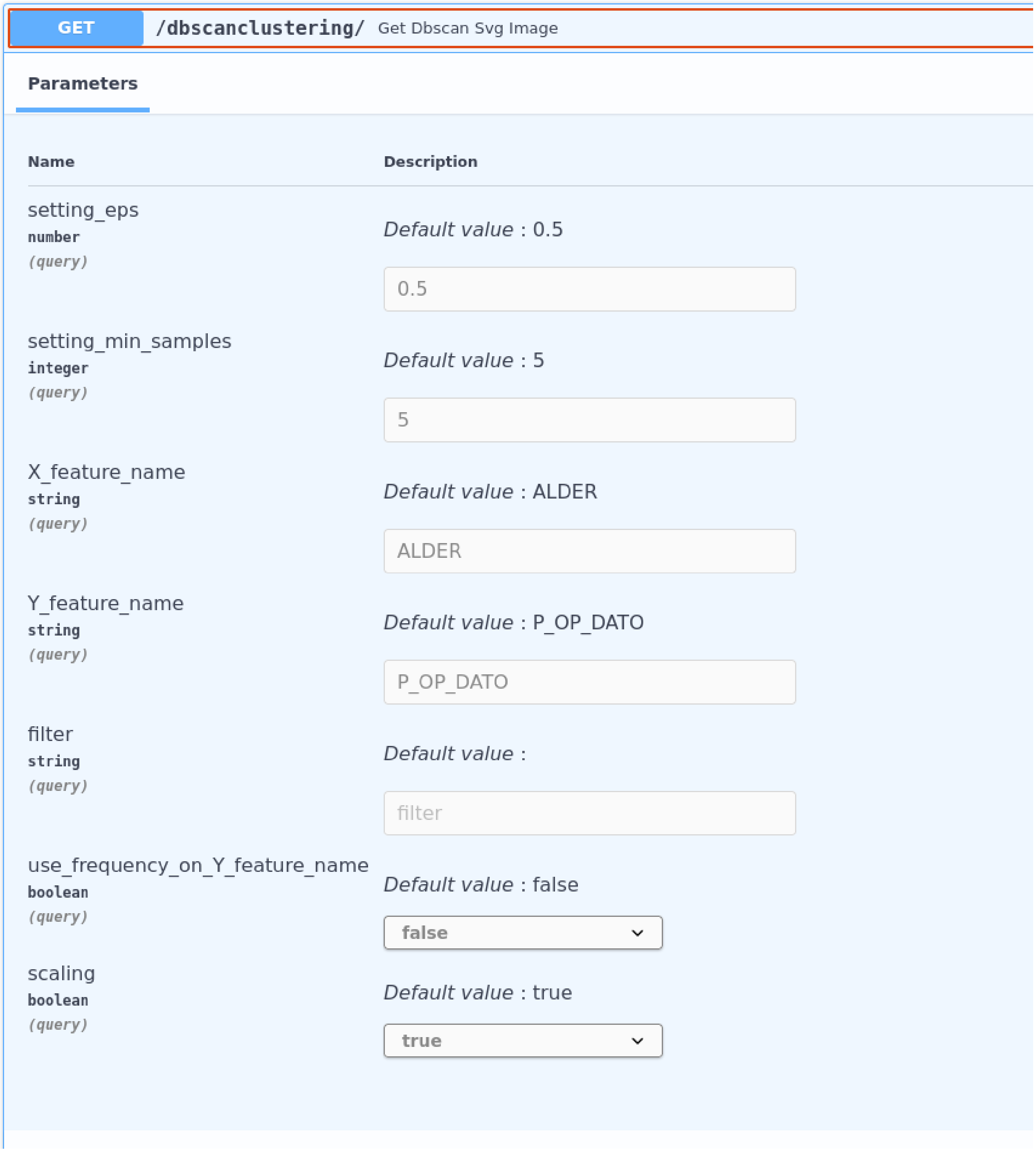
Appendix G: SUS questionnaire



	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	
Participants	I think that I would like to use this system frequently	I found the system unneccessarily complex	I thought the system was easy to use	I think that I would need the support of a technical person to be able to use this system	I found the various functions in this system were well integrated	I thought there was too much inconsistency in this system	I would imagine that most people would learn this system very quickly	I found this system very cumbersome to use	I felt very confident using the system	I needed to learn a lot of things before I could get going with this system	SUS score
P1	4	1	5	1	4	1	3	1	4	1	87.5
P3	5	2	4	1	5	1	3	1	5	1	90
P2	4	1	5	1	4	1	5	1	4	1	92.5
P4	4	2	4	1	4	1	4	2	4	2	80
P5	4	2	4	2	5	1	3	1	4	2	80

Figure 33: SUS questionnaire with answers

Appendix H: API: DBSCAN



GET /dbscanclustering/ Get Dbscan Svg Image

Parameters

Name	Description
setting_eps number *(query)*	*Default value* : 0.5 0.5
setting_min_samples integer *(query)*	*Default value* : 5 5
X_feature_name string *(query)*	*Default value* : ALDER ALDER
Y_feature_name string *(query)*	*Default value* : P_OP_DATO P_OP_DATO
filter string *(query)*	*Default value* : filter
use_frequency_on_Y_feature_name boolean *(query)*	*Default value* : false false
scaling boolean *(query)*	*Default value* : true true

Figure 34: API endpoint for DBSCAN clustering

Appendix I: Rpy2 plots

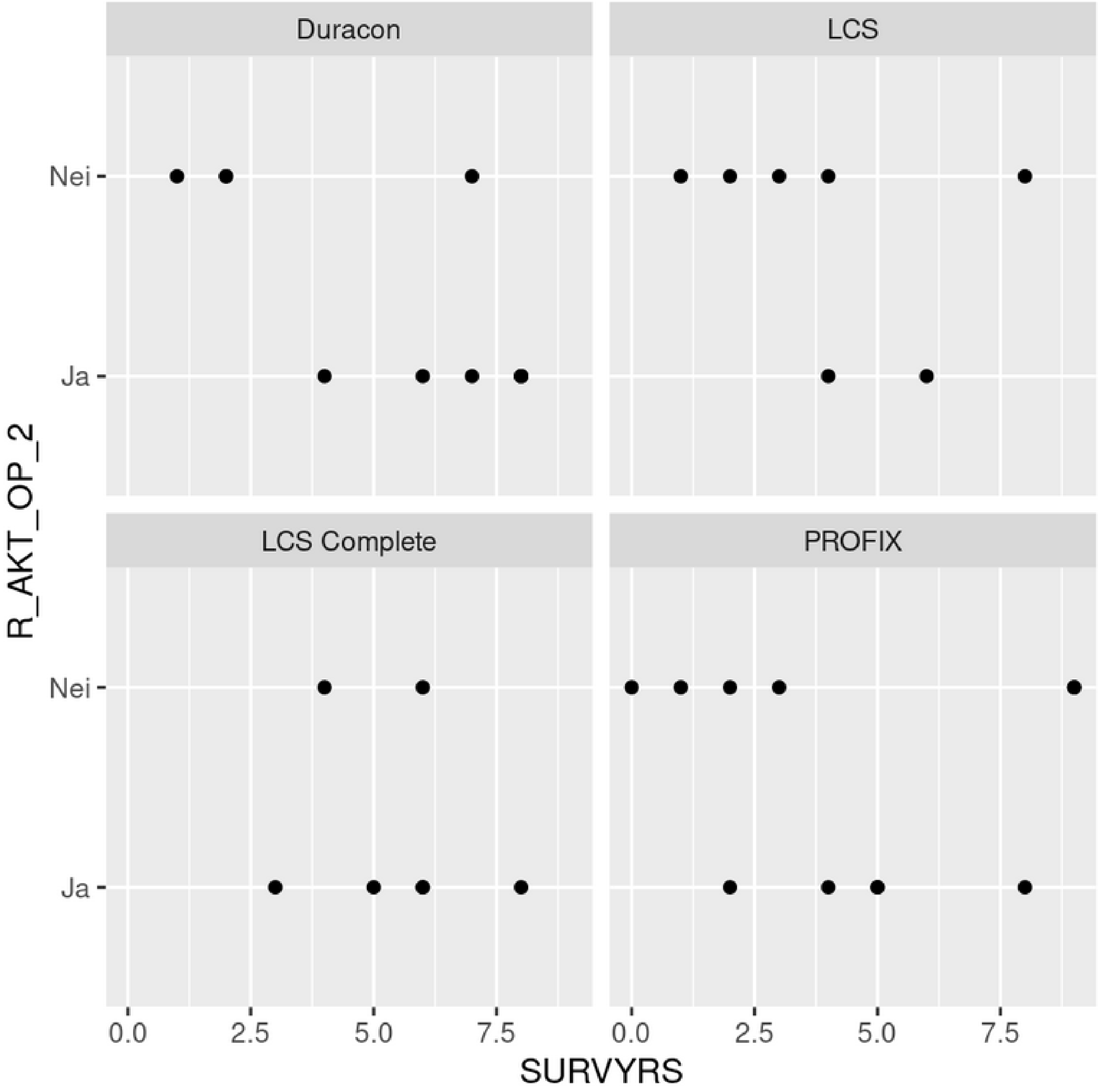


Figure 35: rpy2 plots