Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

Tiril Sandaker Syslak

Exploring Methods for Clustering Medical History of Patients with Suspected Bloodstream Infections

Master's thesis in Computer Science
Supervisor: Øystein Nytrø
Co-supervisor: Rajeev Bopche
June 2023

NTNU
Norwegian University of
Science and Technology

Tiril Sandaker Syslak

Exploring Methods for Clustering Medical History of Patients with Suspected Bloodstream Infections

Master's thesis in Computer Science
Supervisor: Øystein Nytrø
Co-supervisor: Rajeev Bopche
June 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Abstract

Bloodstream Infection (BSI) is a serious medical condition where bacteria are present in the blood, which often escalates to the severe and high-mortality condition, sepsis. Early detection and treatment of BSIs are critical, but the diagnostic process is challenging due to non-specific symptoms and time-consuming laboratory tests. Recent applications of machine learning, specifically clustering algorithms, in healthcare have shown promising capabilities in uncovering hidden patterns in large datasets. However, the application of these algorithms faces limitations in dealing with the complex structure of medical data. This research, conducted in collaboration with the Computational Sepsis Mining and Modelling (CoSem) research group, aims to explore these limitations and the potential of clustering algorithms in a clinical context, especially for identifying risk factors for BSI.

Following a Design Science Research approach, this study commenced with a thorough literature-review to identify relevant features to describe a patient's medical history and to explore the use of clustering algorithms in a clinical context. An extensive analysis was conducted on a complex dataset consisting of 35,694 patients with at least one suspicion of BSI, resulting in a selection of 10 variables to describe each patient's medical history and condition.

Inspired by the exploration of two existing algorithms, a novel approach, Single and Set values Clustering Algorithm (SASCA), was developed to effectively cluster medical data. This algorithm revealed both expected and unexpected clinical relationships among 30 generated clusters. The findings suggest that the application of clustering methods, particularly SASCA, is able to differentiate patients based on their medical history. To maximize the clinical utility of clustering algorithms in similar contexts, the study concludes that these should be used as preliminary tools for further analysis. This approach underscores the necessity of a well-defined interdisciplinary collaboration.

Sammendrag

Bakteriemi er en infeksjonstilstand der bakterier forekommer i blodet, og kan videre føre til blodforgiftning, eller sepsis. Sepsis er en alvorlig sykdomstilstand med høy dødelighetsrate. Tidlig påvisning og deteksjon av bakteriemi er avgjørende, men diagnostisering er utfordrende på grunn av uspesifikke symptomer og tidkrevende laboratorietester. I den siste tiden har bruk av maskinlæring, spesielt klyngingsalgoritmer, i helsevesenet vist lovende evne til å avdekke skjulte sammenhenger i store datasett. Imidlertid ser man begrensninger ved disse algoritmene når de skal håndtere den komplekse strukturen i medisinske data. Denne oppgaven, som er utført i samarbeid med forskningsgruppen Computational Sepsis Mining and Modelling (CoSem), har som mål å utforske disse begrensningene og potensialet klyngingsalgoritmer har i en klinisk kontekst, spesielt med tanke på avdekking av risikofaktorer for bakteriemi.

Forskningen følger en tilnærming av Design Science Research, og startet med en grundig litteraturgjennomgang for å identifisere relevante variabler for å beskrive en pasients medisinske historie og for å utforske klyngingsalgoritmer anvendt i en klinisk sammenheng. En omfattende analyse ble utført på et komplekst datasett bestående av 35,694 pasienter med minst én mistanke om bakteriemi. Analysen resulterte i et utvalg av 10 variabler for å beskrive hver pasients medisinske historie og tilstand.

Inspirert av utforskningen av to eksisterende algoritmer ble en ny tilnærming, Single and Set values Clustering Algorithm (SASCA), utviklet for effektivt å klynge medisinske data. Denne algoritmen avdekket både forventede og uforventede kliniske sammenhenger blant 30 genererte klynger. Funnene antyder at anvendelsen av klyngingsmetoder, spesielt SASCA, er i stand til å differensiere pasienter basert på deres medisinske historie. For å maksimere den kliniske nytteverdien av klyngingsalgoritmer i lignende sammenhenger, konkluderer studien med at disse bør brukes som innledende verktøy for videre analyse. Studiet understreker nødvendigheten av et godt definert tverrfaglig samarbeid.

Preface

This Master's thesis was conducted during the Spring of 2023 as a finalization of the Computer Science degree at the Norwegian University of Science and Technology (NTNU) in Trondheim. The work has been in collaboration with the CoSem group under the guidance and supervision of Øystein Nytrø and Rajeev Bopche.

My deep gratitude goes to my supervisor Øystein Nytrø, and co-supervisor Rajeev Bopche, for valuable insights during our meetings. I am also profoundly thankful to the rest of the CoSem group for providing important feedback and discussions to improve the interpretation of my findings. A special thank you also goes to my fellow student Inger-Ane Sætra Schefte for much appreciated discussions regarding this specific project.

I would also like to take the opportunity to thank all the amazing people I am surrounded by. The work of this thesis would not have been the same without my family. I would like to express my greatest gratitude to you for always cheering me up and believing in me, and also giving motivational nudges when needed. A huge thank you to all my friends, as well as the student association Abakus, for making the past five years the best. Not to forget the best roommates I could ask for, Amanda Grodås and Ingrid Yttervoll. I will forever appreciate our friendship.

Tiril Sandaker Syslak
Trondheim, 18th June 2023

Contents

List of Figures

List of Tables

Acronyms

BCSS Between-Cluster Sum of Squares.

BSI Bloodstream Infection.

CDSS Clinical Decision Support System.

CH Calinski-Harabasz.

CoSem Computational Sepsis Mining and Modelling.

DDSCA Demographics and Diagnosis Sequences Clustering Algorithm.

DSR Design Science Research.

GMU General Medicine Unit.

IC Information Content.

ICD International Classification of Diseases.

ICU Intensive Care Unit.

MASPC Maximal-frequent All-confident pattern Selection.

MFA Maximum-Frequent All-confident Itemsets.

MFI Maximum-Frequent Itemsets.

SASCA Single and Set values Clustering Algorithm.

WCSS Within-Cluster Sum of Squares.

XAI eXplainable Artifical Intelligence.

1. Introduction

This thesis explores the application of various clustering algorithms to medical data from patients suspected of having a bloodstream infection. The goal of the clustering is to uncover potential patterns in patient history that may correlate with the outcome of the patient.

The upcoming chapter provides a broad overview of the key elements of the thesis. It begins with the background and motivation in Section 1.1, laying the foundation for the study's direction. The thesis goal and research questions are then outlined in Section 1.2. In Section 1.3, the chosen research method will be introduced, followed by an outline of the thesis contributions in Section 1.4. Finally, Section 1.5 provides an overall summary of the thesis structure.

1.1. Background and Motivation

Bloodstream Infection (BSI) is a serious medical condition that places a significant burden on healthcare systems (Viscoli, 2016). Characterized by the presence of pathogenic microorganisms in the bloodstream, BSI can often progress into sepsis, a condition with severe health implications and high mortality rates. Early detection and treatment of BSI is crucial to prevent this progression. However, diagnosing BSI is challenging due to their non-specific symptoms and the need for time-consuming laboratory tests for confirmation. Furthermore, the patterns indicating who might have an increased risk of developing BSI remain unclear.

Over the past years, machine learning has shown promise in its ability to uncover hidden patterns and insights from large volumes of data. It has the potential to revolutionize diagnostics, treatment planning, and patient outcomes. Clustering algorithms, specifically, have gained substantial attention in medical research due to their capacity to group similar patients based on different features. This enables a deeper understanding of patient subgroups, which can inform personalized treatment strategies and improve patient outcomes.

Computational Sepsis Mining and Modelling (CoSem) research group at NTNU Health works with the combination of computer science and research of Sepsis (NTNU, a). CoSem focuses on the extraction and analysis of complex health data to improve infectious disease management, and contributes with further research and technology within the topic. With this, the research group aims to establish a platform-agnostic, case-based decision support system. The group is a part of the Gemini centers, a research cooperation between NTNU, SINTEF, UiO, St.Olavs and NTNU Social Research.

This research is conducted as a collaboration with CoSem, and aims to further explore how clustering algorithms can be applied to medical data of patients with suspicion of BSI. Analyzing the medical history of these patients involves both single values, like demographics and aggregated features, and set values, like set of prior diagnosis codes. While clustering algorithms have shown value in healthcare, they face limitations when dealing with medical data of varying structure. This gap represents a significant limitation, given the importance of understanding a patient's full medical trajectory in analysing conditions like BSI. This thesis is therefore designed to explore and address this gap.

1.2. Goals and Research Questions

Given the challenges identified in the previous section, the research goals and questions have been formulated as follows:

Goal *To explore the application of clustering algorithms for grouping patients suspected of having a bloodstream infection, and to analyse how and which features of the medical history relate to patient outcomes.*

This goal involves exploring different clustering algorithms, and see how they can be applied to analyse the history of patients with suspected bloodstream infections. This will be done by a thorough examination of different clustering methodologies, their application in a real-world healthcare context, and the subsequent analytical steps necessary to interpret their results.

Furthermore, another part of this research involves understanding how various features from a patient's medical history might connect to patient outcomes. The aim is to analyse these features, and seek patterns and correlations that can potentially be tied to the patient outcomes.

Research question 1 *What are the relevant features to be used to describe a patient's medical history in the context of clustering?*

This question aims to identify the important characteristics or features in a patient's medical history that should be considered when clustering patients. These features could include demographics, the previous diagnoses, number of hospital visits among others. This question is crucial for determining the input to the clustering algorithm and will involve a review of existing literature, a preliminary exploration and a feature selection process.

Research question 2 *How can the application of clustering help differentiate patients with varying outcomes in suspected bloodstream infection cases, considering their medical history?*

This question seeks to investigate how the different clustering algorithms can be applied to separate patients into distinct groups based on their medical history. The underlying

hypothesis is that patients with similar medical histories may exhibit similar outcomes when they are suspected of having a bloodstream infection. This question will involve applying the clustering algorithms to the provided data and analysing the resulting patient clusters.

Research question 3 *What is the clinical utility and potential of clustering methods in revealing relationships between relevant features of a patient's medical history and patient outcomes in suspected bloodstream infection cases?*

This question aims to explore the clinical utility and potential of the identified clustering methods in discovering relationships between the relevant features from a patients medical history and their outcomes in the context of suspected bloodstream infections. The objective is to understand the practical application and value of these methods in a clinical setting. This exploration will help reveal how the clusters and their features might correlate with patient outcomes, providing insights in the underlying structure of the data. This analysis will further investigate the influence of the selected features on patient outcomes, thus enriching our understanding of the clinical significance and potential of the application of a clustering algorithm in a clinical context.

1.3. Research Method

The research methodology followed in this study is rooted in a design science research approach, which balances theoretical understanding with practical application (Peffers et al., 2007; Hevner, 2007). Details of the methodology is given in Chapter 4, but this section aims to give a summary of the applied methods.

The initial step of the research involves an in-depth literature review, focusing on both the relevant features to describe a medical history, and examining the field of machine learning employed in medical data contexts. This step allows to build a solid foundation of existing knowledge and to identify the potential gaps this research can fill. Some parts of this step was done as a part of the course TDT4501 - Specialization project, done in preparation for this Master's Thesis during the fall of 2022.

Parallel with this review, the provided dataset will be explored. This is to get familiar with the content and connections within the data, and try to both get an idea of how the medical history of a patient can be presented, and how this can be used as input to a clustering algorithm. Following this will be the data selection and aggregation, data preprocessing and exploratory data analysis. This analysis helps to gain insight into the selected data, including identifying patterns.

The selected data will then be used as input for various algorithms with the goal of clustering the patients. This stage will be an iterative process, where insights gained will inform the development of a new clustering algorithm especially suitable for the purpose of this study. This proposed algorithm is intended to improve upon existing methods by effectively handling the unique structure of medical history data. Once developed, the algorithm will be applied to the dataset according to the established experimental plan, before the resulting clusters are validated by clinicians.

The final step of the research process is the evaluation, involving both evaluation of the algorithm's performance by analysing the compactness of the clusters, and the clinical analysis of the cluster result.

1.4. Contributions

As a part of this master's thesis, the key contributions are as follows:

- Introducing a novel approach for clustering medical data, SASCA, emphasizing the benefits of tailored approaches
- An evaluation of the potential of MASPC and DDSCA when applied to a complex dataset with different objectives
- Investigation of the clinical utility of clustering, establishing its potential as an initial step for further analysis
- Exploration and identification of features for describing medical data
- Highlighting the importance of precise ICD coding
- Providing an in-depth exploration and description of the HUNT dataset

1.5. Thesis Structure

- Chapter 2 introduces the clinical theory necessary for domain knowledge, before introducing some relevant theory of machine learning.
- Chapter 3 reviews existing literature and studies closely related to the research topic, both how to describe a medical history with features and the application of machine learning in a clinical context.
- Chapter 4 describes the research design and the methodology employed for the study. It details the process from data understanding and selection to the application of existing clustering algorithms and development of a new algorithm.
- Chapter 5 delves into the dataset used in this study, detailing the selection process, preprocessing steps, and an exploratory analysis. Additionally, it outlines the technological environment employed and any requisite approvals.
- Chapter 6 presents the details of the experiments and the results obtained from the implementation of the clustering algorithms on the dataset. It provides an analysis of the results, illustrating the patterns and trends identified.
- Chapter 7 evaluates the results of the study, including both the data selection, experiments, results and the methodology.

- Chapter 8 discusses the results of the study in relation to the research questions and goal, as well as discussing the limitations of the study.
- Chapter 9 summarizes and concludes the research, how the research answer the research questions and the provided contributions. It ends with suggestions for future work.
- Appendix A contains the PostgreSQL query to select the data utilized in this study.
- Appendix B provide the full code implementation of SASCA in Python, used to form the clusters in the experiments.

2. Background Theory

In the exploration of medical data through machine learning, both clinical and computational knowledge are essential. Therefore, this chapter offers a comprehensive overview, highlighting key concepts in both Clinical Theory (Section 2.1) and Machine Learning (Section 2.2).

2.1. Clinical Theory

The following section aims to provide the necessary domain knowledge concerning Bloodstream Infection (BSI) and the International Classification of Diseases (ICD). It is worth noting that this section builds upon work carried out during the preparatory work done in the specialization project.

2.1.1. Bloodstream Infections

A Bloodstream Infection (BSI) is a severe medical condition characterized by the presence of bacteria in a patient's bloodstream. Clinically, these infections are recognized when systemic signs of infections are present together with a positive blood culture test (Timsit JF, 2020). Introducing details of the condition like detection method, consequences, treatment and prevention is important domain knowledge when looking in to the experiment.

BSIs can be broadly categorized into primary or secondary infections (Centers for Disease Control and Prevention, 2022). Primary BSIs originate directly in the bloodstream, often due to compromised skin integrity. These infections frequently result from medical procedures, such as the insertion of venous catheters, and can be directly associated with hospital settings and treatments provided therein. Such infections are commonly referred to as healthcare-associated or nosocomial infections. In contrast, secondary BSIs arise from infections originating elsewhere in the body and subsequently spreading to the bloodstream. The source of these infections can range from the urinary tract to the lungs.

Consequences

A systematic review by Goto and Al-Hasan (2013) estimated BSI-incidents and BSI-deaths among patients, both all kinds of BSI and nosocomial BSI (Goto and Al-Hasan, 2013). In Europe, the estimated number of BSI episodes exceeded 1.2 million, with around 15% ending in deaths. The number of nosocomial episodes was 240,000, with a slightly higher death rate. Due to the study only focusing on numbers from North

America and Europe, it is reasonable to compare the number of deaths from BSI to the number of deaths related to human immunodeficiency virus (HIV), tuberculosis and malaria, i.e. the world's biggest infectious disease killers.

One of the reasons for the high mortality rate is the high risk of developing sepsis, which is the body's response to a BSI. Sepsis is a life-threatening condition, typically recognized using either the traditional Systemic Inflammatory Response Syndrome (SIRS) criteria, or the more recent Sequential Organ Failure Assessment (SOFA) and its quicker alternative, the quick SOFA (qSOFA). While SIRS focuses on identifying systemic inflammatory response, SOFA and qSOFA emphasize the assessment of organ dysfunction, providing a more comprehensive measure of disease severity. In Norway, sepsis is the second leading cause of death after cardiovascular diseases (Waagsbø, 2022).

In addition to severe consequences for the patient, BSIs also impose considerable economic burdens on healthcare institutions. According to a study by Jarvis (1996), BSI in the United States result in the patient staying one to three extra weeks, incurring extra costs of around 3000-40000$ per patient (Jarvis, 1996). False-positive tests, in comparison with true negatives, can increase costs by 50% (Bates et al., 1991).

Detection and Treatment

Blood cultures are used as a primary method for identifying bacteria in the bloodstream. This is a technique that collects samples of blood from the patient, before the samples are incubated under specific conditions to promote the growth of pathogens (Tønjum). Positive results will be further analyzed to find the specific pathogen causing the infection. This is an important step to finding the correct treatment, as the given antibiotics will be tailored to the specific pathogen causing the infection.

Generally, bacterial pathogens are categorized as either gram-negative or gram-positive. Laupland and Church (2014) identifies the gram-negative *Escherichia coli (E.coli)* as the most prevalent bacterial cause of BSI, and the gram-positives *Staphylococcus aureus (S. aureus)*, and *Streptococcus pneumoniae (S.pneumoniae)* as the second and third, respectively (Laupland and Church, 2014). The wait time for test results can vary, and slow-growing microorganisms can be challenging to identify due to this delay (Peker et al., 2018). Additionally, false-positive results can occur due to contamination during sample collection or processing (Universitetssykhuset Nord-Norge).

Prevention

It has been asserted that BSI, particularly those related to catheters, are largely preventable (Patil et al., 2011). A high correlation has been observed between the number of attempts at catheter insertion and infection rate. Consequently, proper catheter insertion, preferably by experienced medical professionals, can significantly reduce the incidence of nosocomial infections. The duration of catheterization and the method of catheter replacement also play essential roles in preventing infections (Garnacho-Montero et al., 2008; Cook et al., 1997).

In light of these considerations, the number of infections could be reduced with more cautious procedures, particularly for patients with an increased risk.

2.1.2. International Classification of Diseases

The International Classification of Diseases (ICD) is an international standard diagnostic tool, developed by the World Health Organization (World Health Organization). The primary aim is to provide a comprehensive classification system for diseases, disorders, injuries and other health conditions, which enables healthcare professionals to systematically record, report and analyze health data. This ensures consistency and comparability of health information across different hospitals, regions and countries. In addition to being an important tool for analyses and evaluations, the codes also play a crucial role in medical billing and reimbursement, as they facilitate the communication of diagnostic information between healthcare providers and insurance companies.

In 2015 WHO published ICD-10, which now is the standard at the most hospitals, including St. Olavs Hospital. However, the improved version ICD-11 was introduced in 2019, and implemented at some hospitals in January 2022 (World Health Organization). This version introduce updated classification structure, inclusion of new diseases and better support for electronic health records. As this research use data from before the introduction of ICD-11, the tenth version will be in main focus. However, to ensure future work, it is important to make the research adjustable for the newer version. It is also worth mentioning that some researches also use the prior version, ICD-9. This system consists of only digits, and it is not a straightforward one-to-one mapping from the ninth version to the tenth, as will be further described in Chapter 4 and Chapter 6.

The ICD-10 codes consist of a letter followed by two or more digits. The codes are hierarchical, with each sign in the code providing additional specificity about the diagnosis. As a result of the specificity of the coding, the standard also provides information regarding cause and consequences. An example of this structure can be shown in the difference between ICD-10 code A03.0: *Shigellosis due to Shigella dysenteriae* and A03.1: *Shigellosis due to Shigella flexneri*. Both of them represent the same diagnosis, but with different cause. The hierarchy can be represented as a tree, where the root note represent all ICD-codes, and each level beneath represent a new level of specificity. A visual representation of this structure is provided in Figure 2.1.

A limitation with the use of ICD-codes as a basis for research related to disease tracking is the economical motivation of the coding. When clinicians do the coding, they do it to ensure that the hospital gets enough funding for the procedures and treatment related to the disease, and does not consider the research aspects. Additionally, with the introduction of Helseplattformen, a new platform introduced at St.olavs in 2022 (St. Olavs hospital), the reliability of the coding is decreasing. In a meeting with CoSem (CoSem, 2023), the clinicians admitted that the number of diagnoses coded has decreased with the release of the new platform. In addition to the fact that this naturally will lead to problems with funding, it will also create problems for future research like this. Categorization of diagnoses would present a significant challenge if the associated codes are either inaccurate or absent from the dataset.

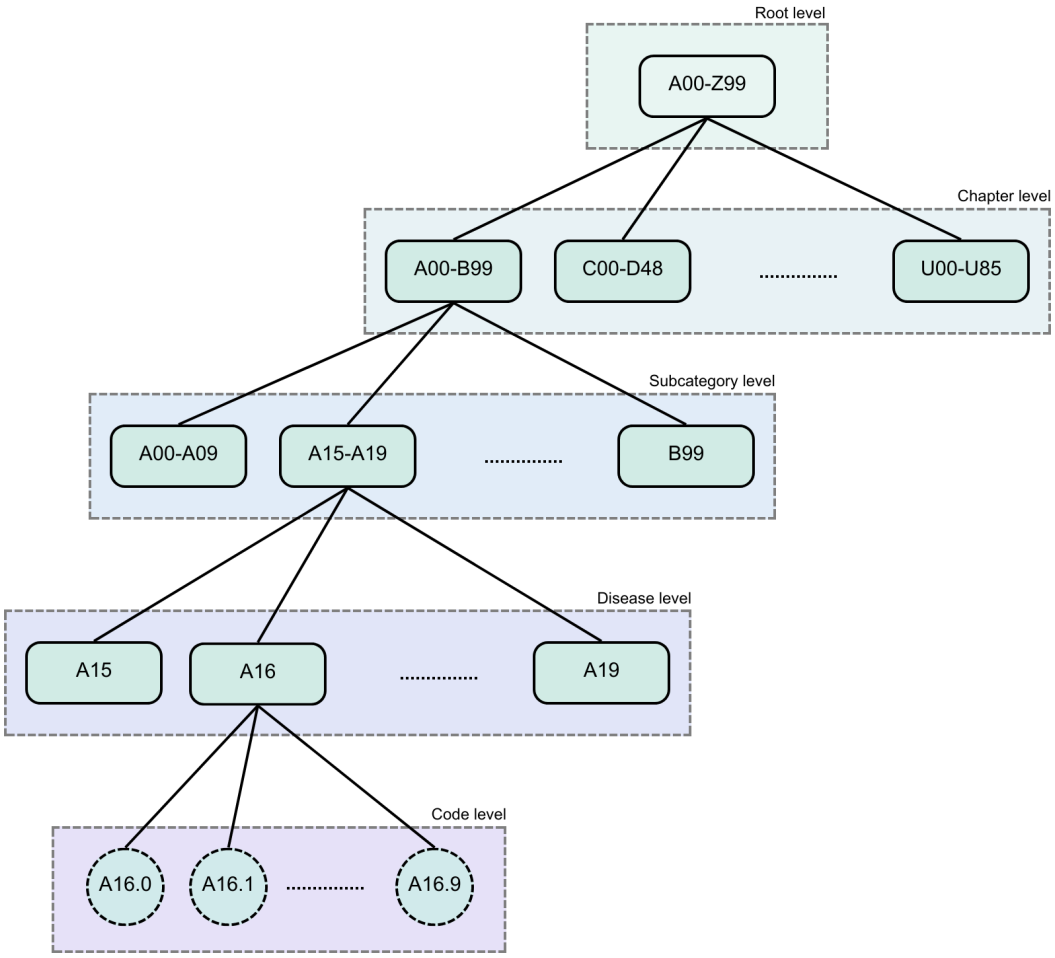


Figure 2.1.: ICD hierarchy representation

2.2. Machine Learning

Moving further to the machine learning theory, this section starts with delving into the significant aspects of machine learning utilization in healthcare. Subsequently, the differences between supervised and unsupervised learning are discussed, followed by presentations of various distance measures and evaluation metrics. The first section is based upon work done in the preparatory project.

2.2.1. Machine Learning in Healthcare

Healthcare is one of many domains benefiting from computer science in general and lately machine learning specifically. A system of the combination of these two is known as a Clinical Decision Support System (CDSS). This type of support could be done by gathering, filtering, and visualizing clinical data. Support from machine learning could help healthcare in taking decisions.

For my project, the CDSS will be the communication of the risk factors to the clinicians. However, it is important that the findings will be a support, and will not replace the reasoning done by the clinicians. To actually make use of the results from the machine learning algorithm in the reasoning, it is important to understand the details behind the result. This is known as eXplainable Artifical Intelligence (XAI) (Arrieta et al., 2020).

2.2.2. Supervised Versus Unsupervised Approaches

Machine learning algorithms are often categorized as either supervised or unsupervised. Supervised learning is a type of machine learning that involves learning from labeled data, where each input data point is associated with an output label or target value. The goal of supervised learning is to create a model that can predict the output label for new data points, based on the patterns and relationships it has learned from the training data. One of the main usage tasks of supervised learning is classification, where the goal is to categorize data points into distinct classes.

Unsupervised learning, on the other hand, is a type of machine learning that usually deals with unlabeled data, meaning there are no known output labels or target values associated with the input data points. The primary goal of unsupervised learning is to discover hidden patterns, structures, or relationships within the data that may not be as easy to find through normal analysis. This is often achieved through techniques like clustering, where similar data points are grouped together.

Although supervised learning models can be highly accurate in predicting outcomes, they can also be complex and challenging to interpret. For individuals without a computer science background, such as clinicians, prediction models can seem like black boxes with limited transparency. The reasoning behind the models' decisions can be difficult to comprehend, making them less suitable for interpretation as a CDSS.

In this project, the primary aim is not to predict outcomes directly, but rather to explore underlying patterns in the data. This makes clustering a more appropriate tool

for the task, especially since it allows for discovering novel associations while still aligning with the principles of XAI.

2.2.3. Distance Measurements

The selection of an appropriate distance measure plays a crucial role in the performance of a clustering algorithm. Such measures quantify the dissimilarity between data points, which the algorithm uses to group the points into similar clusters. This section introduces a few distance measures, each chose for its specific relevance to the requirements of the study's data and method.

Euclidean Distance

Euclidean distance is a widely used metric for measuring the distance between two vectors, or points, in the Euclidean space. This space refers to a n-dimensional space where each point is given by coordinates, one for each dimension. Given two points in the Euclidean space, p with coordinates $(p_1, p_2, ..., p_n)$ and q with coordinates $(q_1, q_2, ..., q_n)$, the euclidean distance is calculated as the square root of the sum of the squared differences between their corresponding coordinates. This is summarized in Equation (2.1).

The minimum possible Euclidean distance between two points is 0, which occurs when the two points are identical. However, the maximum possible distance is not bounded, as it depends on the largest possible difference between the coordinates of any two points the dataset. To interpret with this, it is common to normalize the data before calculating the distance. When the data is normalized before calculation, the result is within a range of 0 and 1.

$$d_{euc}(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + ... + (p_n - q_n)^2} \tag{2.1}$$

Weighted Edit Distance

To understand the concept of weighted edit distance, it is essential to first understand the basics of edit distance. This measurement, which is also known as Levenshtein distance, is used for comparing similarities between two sequences. This is done by calculating the minimum number of transformation needed to change sequence one, s_1, to become sequence two, s_2. The transformations can be a deletion with a cost of 1, an insertion with a cost of 1 or a substitution with a cost of 1. An example where this measurement is appropriate is when comparing two strings, and each letter substitution is counted as equal cost.

However, in some scenarios, the different substitutions are not uniform. When comparing strings and taking the likelihood for a spelling mistake into account, it is obvious that substituting a with e should have lower cost than substituting a with p, as there is much more likely to mix up the two vowels. Weighted edit distance, or weighted Levenshtein distance, facilitates this likelihood by allowing for custom costs, i.e weights, for each operation. This makes the distance measurement more flexible and facilitates adjustments for each specific context. With varying costs, the measurement enables more

accurate comparisons between the sequences. The formula for computing the distance between the sequences s_i and s_j, denoted d_{WE}, is given by Equation (2.2). The tail represent the sequence without the first element.

$$d_{WE}(s_i, s_j) = \begin{cases} |s_i|, & \text{if } |s_j| = 0 \\ |s_j|, & \text{if } |s_i| = 0 \\ d_{WE}(\text{tail}(s_i), \text{tail}(s_j)), & \text{if } s_i[1] = s_j[1] \\ \min \begin{cases} d_{WE}(\text{tail}(s_i), s_j) + 1, \\ d_{WE}(s_i, \text{tail}(s_j)) + 1, \\ d_{WE}(\text{tail}(s_i), \text{tail}(s_j)) + \text{sub}(s_i[1], s_j[1]), \end{cases} & \text{otherwise} \end{cases} \tag{2.2}$$

Jiang-Conrath Distance

The Jiang-Conrath (JC) distance is a semantic similarity measure that evaluates the degree of relatedness between concepts by examining their information content and their positions in a hierarchical structure. The Information Content (IC) of a node is a measure of the provided specificity of the node, and favors concepts that are less general. Hence, a higher IC means that the code gives more specific information. IC can be calculated by different formulas for different use cases, and Sánchez et al. purpose a new method for ontology-based IC (Sánchez et al., 2011). Their calculation of the IC for a node a is given by the equation Equation (2.3), where |leaves| describes the number of descendants from the current node that are also leaves in the tree, |ancestors| describes the number of ancestors for the current node, and L is the total number of leaves in the tree.

$$IC(a) = -\log_2 \left(\frac{\frac{|leaves(a)|}{|ancestors(a)|+1} + 1}{L + 1} \right) \tag{2.3}$$

The Jiang-Conrath distance uses the given IC equation to calculate the distance between two concepts, determined by the difference between their individual IC and the IC of their least common ancestor (LCA), i.e the most specific concept that is an ancestor of both concepts. The calculation of the JC distance between two nodes a and b is done by using Equation (2.4).

$$d_{JC}(a, b) = IC(a) + IC(b) - 2 \times IC(LCA(a, b)) \tag{2.4}$$

2.2.4. Evaluation Metrics

An important part when using clustering algorithms for analysis is to evaluate the provided clusters. This evaluation is often separated in to compactness within the clusters and separation between the clusters, often referred to as the cohesion and separation. The two values can be measured with the two metrics Within-Cluster Sum of Squares (WCSS) and Between-Cluster Sum of Squares (BCSS), respectively. There

are several ways of calculating WCSS and BCSS, but to reduce complexity, this research employs a center-based approach.

WCSS is, as the name suggests, a measure for evaluating the compactness of the resulting clusters. In this context, compactness refers to how close the data points within the cluster are grouped. The goal is to minimize the measure, as a lower WCSS value indicates a higher degree of compactness. This means that the algorithm has captured similar data points in the same group. The score is calculated by summing the squared distances between the data point and the centroid of its respective cluster. A lower value indicates a more compact cluster.

The BCSS value, on the other hand, capture the separation between the clusters. Separation refers to the distance between the different clusters, with larger distances indicating better separation. The BCSS is calculated by summing up the squared distances between each cluster center and the overall data center, multiplied by the number of records in each respective cluster. This measure evaluates the spread of the centers, with larger values indicating greater separation between clusters.

Both WCSS and BCSS are essential in evaluating the quality of clustering results. The two metrics should be considered when choosing the optimal values in a clustering approach. A good choice should minimize the WCSS while maximizing the BCSS, thus achieving a balance between compact and well-separated clusters. The Calinski-Harabasz (CH) Index is a popular metric for this exact purpose (Caliński and Harabasz, 1974).

The CH index gives a comprehensive measure of the performance of the algorithm, and can be a helpful tool when determining the optimal values for the clustering with the given algorithm and dataset. It is also a good metric when comparing different algorithms on the same dataset, to find the one making the most dense clusters. The index is calculated by the formula given in Equation (2.5), where n_k is the number of records in cluster k, c_k is the center of cluster k, c is the center of all data, n is all records and K is total number of clusters.

$$CH = \frac{traceB}{traceW} \times \frac{(n-K)}{(K-1)} = \frac{\sum_{k=1}^{K} n_k ||c_k - c||^2}{\sum_{k=1}^{K} \sum_{i=1}^{n_k} ||x_i - c_k||^2} \times \frac{(n-K)}{(K-1)} \tag{2.5}$$

3. Related Work

This chapter aims to explore and review existing research related to this study. Given the dual focus of this research — exploring the features that describe a patient's medical history, and investigating the application of clustering algorithms on this medical data — this literature review is likewise twofold. The first section of the chapter, Section 3.1, details the research related to describing a patient's medical history, with an extra focus on how various elements of the history can relate to bloodstream infections. It also explore a potential gap in how to describe a patient's medical history. Subsequently, Section 3.2 elaborates on how previous studies have employed machine learning algorithms in the healthcare sector.

3.1. Describing Medical History and Risk Factors

This section seeks to delve into prior research that has examined the relationships between medical history and patient outcome, particularly concerning BSI. For the literature review, a modified version of the process proposed by Kofod-Petersen was adopted (Kofod-Petersen, 2012). The complexity of the process was intentionally toned down to align with the scope and context of this study.

Google Scholar was utilized as the primary search domain, the following search terms were used to ensure relevance of the returned articles:

```
(Medical history OR Risk factors OR Electronic health records) AND
(Patient outcomes OR Bloodstream infections OR Bacteremia)
```

These terms were chosen to cover both studies representing the medical history of a patient, and risk factors associated with BSI. A selection of the resulting papers' titles and abstracts were quickly reviewed to gauge their relevance to the research topic, and the references in relevant papers were also examined. Papers that appeared to be in alignment with the study's aims were selected for a more in-depth review.

3.1.1. Features Describing Medical History

Schmidt et al. (2012) conducted a study to examine trends in the first-time hospitalisation for acute myocardial infarction (Schmidt et al., 2012). They analysed the trends for each sex and age over a 25-year period, using all diagnoses given to the patient within 5 years prior to the myocardial infection. Their findings suggested that age, sex and prior diagnoses are relevant features in describing a patient's medical history.

Sancho-Mestre et al. (2016) also found the same features relevant to describe the history, in their study that estimated the comorbidities related to diabetes, with results indicating that elderly patients and women suffer more than younger people and men (Sancho-Mestre et al., 2016).

Wu et al. (2010) included variables related to the the most recent visit as a variable, as well as the duration of the gap between two interesting episodes (Wu et al., 2010). They show that aggregated variables like this can contribute in describing a medical history to be used in a machine learning research.

Some studies use mortality prediction models, like the Acute Physiology and Chronic Health Evaluation (APACHE), Mortality Probability Models (MPM), and Simplified Acute Physiology Score (SAPS) (Zimmerman et al., 2006; Higgins et al., 2005; Le Gall et al., 1993). These models utilise a combination of physiological variables, patient demographics, and specific disease factors to predict the risk of patient mortality. They include aspects of a patient's medical history such as prior chronic diagnoses and length of stay prior to admission.

3.1.2. Risk Factors of BSI

Several studies have attempted to identify and evaluate risk factors associated with the development of BSIs and their potential impacts on patient outcomes, particularly mortality.

A study by Pittet et al. (1997) discovered that older age, extended current hospital stay, being male, and diagnoses of cancer or diseases of the digestive system were associated with higher mortality in patients with BSIs (Pittet et al., 1997). Another investigation by Laupland et al. (2008) highlighted risk factors for Staphylococcus aureus BSIs, which included a range of comorbidities such as hemodialysis, HIV, and cancer (Laupland et al., 2008b). In terms of Escherichia coli BSIs, the same authors identified that either very young or older patients, and females aged 1-59 had an increased risk, as well as patients with comorbidities such as dialysis, organ transplant and cancer (Laupland et al., 2008a). These findings suggest that both demographic and medical condition variables can influence the risk of developing BSIs and the associated mortality rate.

Some studies have also examined risk factors associated with a patient's history other than their diagnosis. Baek et al. (2021) employed a binary representation of admission within three months prior to BSI, and prior use of antimicrobials and medical devices within the same three months, to characterize this history (Baek et al., 2021). They found that patients admitted to the hospital, especially to a long-term care hospital, within three months prior, had an increased risk of community-onset extended-spectrum β-lactamase-producing Escherichia coli (CO ESBL-EC) BSI.

Fram et al. (2015) used features like different demographic values, comorbidities, and previous treatment, along with prior and current insertion of venous catheters and number of previous hospitalizations (Fram et al., 2015). This study also concluded that recent hospitalization increased the risk of developing BSI.

In summary, several studies have employed features such as age, sex and previous diagnosis to describe a patient's characteristics. However, when investigating the history

for outcomes related to BSI, features like prior hospitalization and time since last stay are more relevant. Despite these insights, it remains challenging to find studies using using attributes like counts of General Medicine Unit (GMU) and Intensive Care Unit (ICU) admissions, and the total length of stay prior to the suspected BSI episode. This indicates a gap in the current literature and map point to potential avenues for future research.

3.2. Applications of Machine Learning Algorithms in Healthcare

Clustering algorithms have been widely used in various healthcare contexts, from predicting patient outcomes to diagnosing diseases and determining treatment. To identify relevant research in this field, the same modified literature review approach as for the risk factors was used. The following search terms were applied using Google Scholar:

```
(Clustering algorithms OR Machine learning OR Artifical intelligence)
AND (Healthcare OR Medical history OR Bloodstream infections)
```

Despite the primary focus on clustering algorithms, these search terms also intentionally capture the studies using prediction models. As the number of studies specifically related to BSI is limited, more general approaches were also included. The results from the search as well as their references were again briefly reviewed, before going in-depth of the ones that appeared to be more relevant.

3.2.1. Machine Learning and Bloodstream Infections

Zaobi et al. (2021) conducted a cohort study using electronic medical records of patients already infected in Tel-Aviv (Zoabi et al., 2021). The study employed an inclusive model with over 600 features and a compact model using 45 features, both of which utilized a gradient-boosting machine model built with decision tree base-learners. The two models produced area under the receiver-operating characteristics curve (AUROC) of 0.82 and 0.81, respectively. Low albumin levels, high red cell distribution width and high creatinine influenced the outcome the most.

Pai et. al (2021) used five different algorithms to identify the best performing early prediction model based on patients' basic characteristics, vital signs, laboratory data, and clinical information (Pai et al., 2021). The study concluded that XGBoost had the highest sensitivity, while Random Forest had the highest specificity. The features indicating an increased risk of BSI included high prothrombin time, lower platelet count and lower albumin.

No studies were found that used medical history as input to clustering algorithms for analyzing risk factors of BSI, which shows a gap in the existing literature that this study will cover.

3.2.2. Clustering Algorithms Using Medical History

Given that this study aims to utilize clustering algorithms for grouping patients, the relevant findings deserve particular attention. Although our goal is specifically related to BSIs, algorithms that use similar dataset, containing both single and set values, are highly relevant for an in-depth review.

MASPC

Zhong et al. (2020) introduced the Maximal-frequent All-confident pattern Selection (MASPC) method as a strategy for clustering patient data containing both single and set values (Zhong et al., 2020). The goal of the algorithm is to discover frequent and correlated diagnosis codes, dividing the process into two parts: MAS and PC. The MAS algorithm finds the maximal-frequent all-confident itemsets within the set of diagnosis codes for each patient. The algorithm uses FPMAX to find the Maximum-Frequent Itemsets (MFI) (Grahne and Zhu, 2003). Frequent itemsets include itemsets that occur at least a minimum support (*minSup*) times, and maximum means that there are no larger itemsets with the same items (i.e superitemset) that are also frequent. The authors utilized an *allConf* threshold to find the correlated MFI's, named Maximum-Frequent All-confident Itemsets (MFA). The *allConf* threshold describes the minimum probability of all items in the itemset appearing in a patient's diagnosis set if one of the items is present. The algorithm only retains MFAs that do not share diagnosis codes with other MFAs, or share diagnosis codes with other sets, but with a minimum overlap (*minOv*) of records that contain both sets.

After choosing the MFA's for the set value in the dataset, the PC algorithm makes a binary representation of the dataset with each accepted MFA as a column together with the single values. This dataset is used as input for an agglomerative average-linkage hierarchical clustering, which constructs the final clusters (Bar-Joseph et al., 2001). The records without diagnosis from any of the accepted MFA's from MAS will remain unclustered.

DDSCA

Following the introduction of MASPC, Zhong et al. (2021) developed a new algorithm, Demographics and Diagnosis Sequences Clustering Algorithm (DDSCA), which takes into account the order of diagnosis codes, making it more suitable for datasets with a history of diagnosis codes (Zhong et al., 2021). DDSCA forms clusters based on the pairwise distance in demographic values and the pairwise weighted edit distance between sequences of diagnosis codes. To measure the distance between demographics, the algorithm constructs an RS-tree, a hierarchical representation of the relationship between records, using agglomerative average-linkage hierarchical clustering (Bar-Joseph et al., 2001). The algorithm uses a Weighted Edit Distance to measure the distance required to transform one sequence into another. DDSCA combines these two measurements into the d_{JCE} measure, as shown in Equation (3.1), where w_{dem} and w_{diag} are the weights for

the demographics and diagnosis, respectively, and d_{JC} and d_{WE} are the Jiang-Conrath distance for demographics and weighted edit distance for diagnosis, respectively.

$$d_{JCE}(r_i, r_j) = \sqrt{w_{dem} \cdot d_{JC}(r_i^{dem}, r_j^{dem}) + w_{diag} \cdot d_{WE}(r_i^{seq}, r_j^{seq})} \tag{3.1}$$

The cluster construction process minimizes the maximum intercluster distance (Gonzalez, 1985). This phase identifies the k centers iteratively, starting with a randomly chosen point from the dataset. Each subsequent center i $(0 < i < k)$ is identified as the data point with the maximum distance to its nearest center. When all k centers are identified, clusters are formed by assigning each data point to its nearest center.

4. Methodology

The methodology used in this research aligns with the principles of the Design Science Research (DSR) approach (Hevner, 2007). This approach is particularly well-suited for iterative problem-solving processes, reflecting the research process employed here. Initial explorations and preliminary experiments laid the foundation for the development of a new algorithm, SASCA. The implementation of the process is described in Section 4.1, before explaining the specific steps taken during the process, emphasizing the iterative, trial and error process. Section 4.2 describes the initial understanding of the data, followed by Section 4.3 that details the preliminary experiments conducted. The chapter concludes with a description of the novel approach, SASCA, in Section 4.4.

4.1. Overview of Process and Research Plan

Before delving into the details of each step conducted as a part of the approach, it is crucial to understand the foundation of the process. The research, as outlined in Section 1.2, aims to explore various clustering algorithms to group patients suspected of having BSIs. This exploratory nature of the research requires an iterative approach, which is well-aligned with the DSR framework.

Given that the methodology is based on a cycle of experimentation and refinement, it is challenging to propose a step-by-step plan. Instead, the process happens more fluidly, as illustrated in Figure 4.1. The initial step relies on defining the problem, which is based on the domain knowledge acquired. As a part of this research, the results of this first step is elaborated in Chapter 2. Once this foundation is set, the research progresses into the heart of the DSR study, expressed as the design cycle by Hevner et. al (2007) (Hevner, 2007).

During this stage, the data will be explored to understand the potential it has. Simultaneously.a thorough review of related work is carried out to identify promising methods that could be adopted to this specific problem. The results of this step is explained in Chapter 3. Each of the potential methods will be adjusted to fit the specific dataset and problem context. Following the implementation, an evaluation is conducted. Dependent on the results of each iteration, this evaluation could involve both clinical and computational aspects, and the results determine whether to conclude the study or to apply the gained insights for repeating the process with a different algorithm. To address the feasibility of clustering in a clinical context, the final clusters will also be evaluated in a clinical context by professionals from the CoSem group. All of these steps are detailed in Chapter 6.

The subsequent sections of this chapter will detail the initial understanding of the

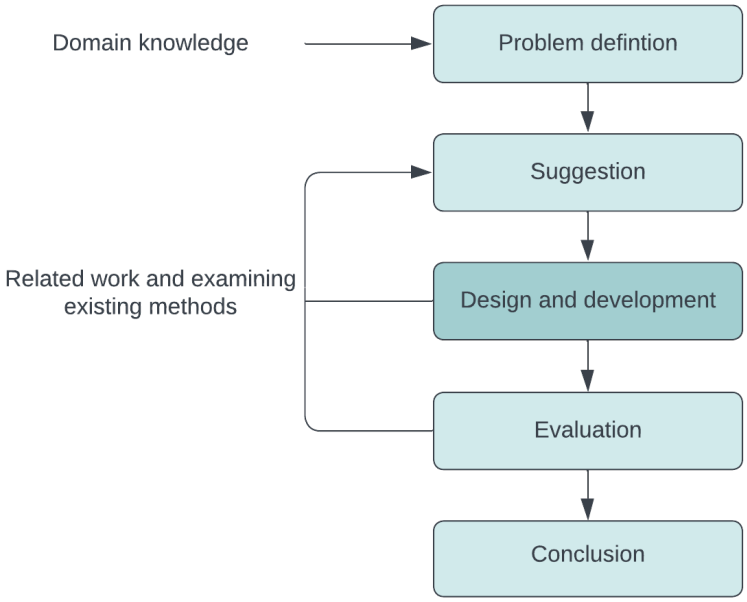


Figure 4.1.: Experimental process

Name	Content description	No. of rows	No. of features
aninopphold	Details of each ICU stay	295,480	11
dimresh	Department hierarchy	2,348	46
doculive	Description of documents written	11,763,721	14
nimesaktivitet	Details of each GMU stay	3,147,336	49
nsml	Laboratory tests and results	1,976,174	13
pasient	Demographics for each patient	35,695	3
trfl	Microbiological tests and results	2,812,961	15

Table 4.1.: Overview of all tables in the provided dataset

data before explaining the specific algorithms implemented within the design cycle. It is important to note that the steps have been created progressively as the research unfolded, and the sections are written in retrospective. This is done as the foundation from each step is used to form the following.

4.2. Initial Data Understanding and Preparation

The initial phase of the research involved gaining familiarity with the complex dataset provided, and was an essential part. Since the dataset was provided without any descriptive information, it was hard to understand the relationships between each feature and the information they were intended to reflect. Hence, this exploration phase required significant time and effort.

To facilitate the process, we utilized Python, specifically libraries such as Pandas, Matplotlib, and Seaborn. These tools enabled us to apply descriptive statistics and data visualization to gain an overview of the data structure and content. The objective of the exploration was to understand the individual characteristics of each table, identify interconnections, map the features, and understand how the provided tables could be utilized to describe each patient's history. The findings from this exploration, including a general description of the content, the number of rows, and the number of features of each table, are summarized in Table 4.1.

Through this exploration, we discerned that all tables, except ***dimresh***, were connected by the *ppid* key. This ***dimresh*** table, which provides information about the hierarchical representation of departments, was concluded irrelevant for the purpose of this research and was consequently ignored in subsequent exploration. Similarly, ***doculive***, which only provided information about the documents written and not the actual content, and hence no essential patient history information, was also excluded. The microbiological tests and results given in ***trfl*** could potentially provide information about a patient's history,

but to limit dimensionality and complexity, this table was also excluded. However, it may prove to be a useful resource for future research.

Following this, the exploration focused on identifying columns that best described the patients' history. The tables ***aninopphold*** and ***nimesaktivitet***, representing the individual stays at the ICU and GMU, respectively, were obvious starting points. To avoid repetition, the details of the process of selecting features will be revisited and expanded upon in Chapter 5.

After finalizing the features, a resulting dataset was derived from joining and aggregating columns of ***aninopphold***, ***nimesaktivitet***, ***nsml*** and ***pasient***. This dataset aimed to numerically represent the historical context of each patient prior to their latest blood culture, and the outcome of the latest episode with suspected BSI.

4.3. Preliminary Experiments with Existing Methods

Having understood and prepared the provided data, another important part of this research was the preliminary experiments done with existing clustering algorithms. As a part of this, both MASPC and DDSCA as introduced in Chapter 3, were explored and partially implemented. However, during the implementation, it became evident that both methods had limitations in the context of our research. This section will outline an overall description of the initial efforts with the mentioned algorithms, while details of the implementation can be found in Chapter 6. The challenges encountered for each algorithm will be addressed, together with the insights that subsequently directed us towards the development of our novel approach, SASCA.

4.3.1. MASPC

MASPC is proved to be both effective and provides valuable results when looking at how diagnosis are correlated. Since the algorithm consider both the demographics values and diagnosis codes, it is well suited for this research.

To implement MASPC algorithm, the inital step was to transform our data into a format suitable for the algorithm, which expected binary values. This included discretizing the numerical values to transform continuous features into categorical ones, before applying one-hot encoding to achieve the binary representation. As the dataset used in the original algorithm consists of ICD-9 codes, some adjustments in the implementation of Apriori and FPMAX needed to be done in order to handle the alphanumeric ICD-10 codes in our dataset.

During the implementation, we encountered a series of challenges. The wide range of values for numerical features presented a hurdle in deciding how to discretize the data without loosing valuable information. Moreover, setting the minimum support (*minSup*) and minimum confidence (*minAc*) parameters proved to be a challenging task. We needed to strike a balance where we would discover enough frequent itemsets without decreasing the *minSup* and *minAc* values too much, which could lead to false associations.

However, the most significant limitation emerged when we realized that the MASPC

algorithm could not adequately handle our research question. This was due to the fact that BSI was not specifically coded in the dataset, and as a result, the algorithm was finding sets of codes that did not include or consider BSI. This critical limitation led us to discard the MASPC algorithm for this particular application. As a lesson from this, we learned that we need to consider single diagnosis codes in the history, and not sets of codes.

4.3.2. DDSCA

The next approach in our exploratory process was to apply the DDSCA algorithm, as it aimed to discover similarities in patient history, taking both demographics and diagnosis codes into account, with an emphasis on the order of the codes.

The implementation of DDSCA involved implementing the provided source code with necessary modifications to fit our dataset. To prepare the single values for hierarchical clustering, it was again necessary to discretize all numerical features. Another necessary change was the recreation of the ICD hierarchy to fit ICD-10 codes, as the original paper only included a textual representation of the hierarchy of ICD-9 codes. We also developed additional code to represent the hierarchy of demographic values fitted to our dataset.

Like for MASPC, the decision to discretize the features may result in the loss of critical information. Another challenge with the implementation was the task of recreating the ICD hierarchy for ICD-10. This ended up being a partly manual approach, since no pre-existing textual representation could be found. Additionally, the seemingly arbitrary hierarchy of demographic values in the original code also proved to be a challenge. Our solution involved deriving the hierarchy from a dendrogram, which necessitated additional coding to discover the edges.

Despite these efforts, the main limitation that led us to discard DDSCA was its computational inefficiency when applied to our dataset. The increased complexity arising from the large number of records and features in our data rendered the distance computations between record excessively time-consuming. This realization led us to the conclusion that we needed a more efficient and scalable approach for our research context.

4.4. SASCA: A Novel Approach to Medical Data Clustering

Taking the advantages of the two aforementioned algorithms and adjusting it for our dataset and research, Single and Set values Clustering Algorithm (SASCA) was made. The algorithm is heavily inspired by DDSCA, but without considering the order of the diagnosis and using euclidean distance for the single values. This way, the use case for history dataset with both single and set values are preserved, but the complexity is decreased. In contrast to MASPC it still consider similarity between two sets of diagnosis codes with some of the same codes, even though the set of codes does not include a maximum frequent itemset.

DDSCA focus on the demographic values with categorical values as single values, and agglomerative average-linkage hierarchical clustering is hence a suitable measurement to

calculate similarity. In contrast, the dataset in this research only includes demographics as birth year and sex. Additionally to the fact of lacking attributes, the goal of this research made it more meaningful to look at other attributes, aggregated from the history. Unlike categorical values, aggregated numerical values represent continuous measurements that can be directly compared using arithmetic operations. Therefore, Euclidean distance is chosen as an appropriate measurement to calculate the distance between the normalized single values of the data points.

The similarity between set values in SASCA use the natural hierarchical tree structure of the ICD-codes. The distance between a set of codes s_i of length n to a another set of codes s_j with length m is calculated by finding the closest code in s_j for each code in s_i, using Jiang-Conrath distance. The measure is normalized by dividing by n. To find the symmetric distance between the two sets of codes, the operation is done in the reversed direction as well. In the end the result is normalized again by dividing by 2, to ensure that the distance does not exceed 1. The range of this measurement lies between 0 and 1. A value of 0 indicates that the two lists are identical, meaning that for each code in s_i there exists a code in s_j with zero JC-distance and vica versa. On the other hand, a value of 1 indicates that there is no similarity between the codes in s_i and s_j. The calculation is summarized in Equation 4.1, where c_i is ICD-code at index i in s_i and c_j is the code at index j in s_j

$$d_{JCS}(s_i, s_j) = \frac{1}{2}\left(\frac{1}{n}\sum_{i=1}^{n}\left(\min_{1\leq j\leq m} d_{JC}(c_i, c_j)\right) + \frac{1}{m}\sum_{j=1}^{m}\left(\min_{1\leq i\leq n} Dd_{JC}(c_i, c_j)\right)\right) \tag{4.1}$$

To adjust the importance of single and set values, each of the distance measures are weighted with w_{single} and w_{set}, respectively. To make sure that the clustering also weights the end outcome of the patient, this feature is extracted from the dataset and handled by its own. Since this is a 1-dimensional point, the distance is found only by taking the absolute value of the subtraction between the two points, and normalized by dividing by the maximum possible value of outcome. The final distance measure for each pair of records r_i and r_j is given by Equation 4.2.

$$\begin{aligned} d_{SASCA}(r_i, r_j) &= \sqrt{d_{single} + d_{set} + d_o} \\ &= \sqrt{w_{single} \cdot d_{euc}(r_i^{single}, r_j^{set}) + w_{set} \cdot d_{JCS}(r_i^{set}, r_j^{set}) + w_o \cdot \frac{1}{3}|(r_i^o - r_j^o)|} \end{aligned} \tag{4.2}$$

This measure is used to find the new centers iteratively using k-centers. The algorithm is given in Algorithm 1, and the steps will now be described. An arbitrarily selected record from the dataset is chosen as the first center, and hence the first cluster (line 1-3). Until all k centers are found, each iteration find the new center as the record that has the closest center furthest away (line 4-7). When all centers are found, each of the remaining records are assigned to the cluster that has the closest center to the given record (line 8-10). The algorithm return the set of clusters, U.

Algorithm 1 SASCA ($\mathbf{D}, w_{single}, w_{set}, w_{outcome}, k$)

Input: dataset $\mathbf{D}$, weights $w_{single}, w_{set}, w_{outcome}$, the number of clusters k
Output: a set of clusters U

$c \leftarrow$ arbitrary record from $\mathbf{D}$
$C \leftarrow c$
$U \leftarrow \{c\}$
for $i = 2$ to k **do**
    Select a record c from $\mathbf{D}$ such that $c \notin C$ and $\min_{c' \in C} d_{SASCA}(c, c')$ is maximized
    $U \leftarrow U \cup \{c\}$
    $C \leftarrow C \cup c$
for $r \in D$ **do**
    if $r \notin C$ **then**
        Assign r to the cluster in U whose center c has minimum $d_{SASCA}(r, c)$
return U

5. Data

In this chapter, the dataset used for the study will be presented. Starting with an overview of the dataset's characteristics in Section 5.1, followed by the data selection process in Section 5.2, and the preprocessing steps in Section 5.3. Preliminary investigations and pattern recognitions are explored in Section 5.4. The chapter then presents the computational environments used in Section 5.5 and finish off with the necessary ethical considerations and approvals in Section 5.6.

5.1. Description of the Data

The dataset utilized in this study originates from St. Olavs Hospital in Trondheim and is part of the HUNT Research Centre (NTNU, b). It contains demographic details and associated hospital visit characteristics for 35,695 patients, of whom 35,694 had at least one episode with suspicion of Bloodstream Infection (BSI). Further in this research, a single distinct hospital stay will be referred to as an *episode*, and an episode where the patient is suspected of having a BSI will be denoted as a *suspected episode*.

The period for these suspected episodes extends from January 1, 2013, to March 7, 2020. Together the suspected episodes, the dataset also includes all other episodes for these patients occurring from April 20, 1999, to May 31, 2020.

As briefly introduced in Chapter 4, each of these episodes includes which laboratory tests (***trfl***) and microbiology tests (***nsml***) that were taken within the same time frame as for the episode, if any, and the corresponding results. With the table ***nimesaktivitet***, it also includes more detailed information about the stay, like department, duration, ICD-codes and the degree of urgency. For a clearer understanding, it is recommended that the reader reviews Table 4.1. As previously noted in the corresponding section, Section 4.2, the provided data was separated in to seven different tables, but only four of them were used in this research. These four tables with a short description and relevant columns are given in Table 5.1.

In addition to the data provided from HUNT and St.Olavs, public data from E-helse is also used as a basis for the ICD-hierarchy. This was collected from Direktoratet for e-helse medisinske kodeverk.

While the data used in the research provides valuable insights, there are several limitations that should be considered. Firstly, the dataset only contains history for the patients at St. Olavs, and does not take into account that the patient may have visited other hospitals in between. In addition, data from only one place might not be representative of the broader population. Another limitation is that the dataset only includes patients where there have been ordered at least one blood culture, i.e patients

Tablename	Description	Relevant features
pasient	Demographics for each patient	**ppid** - id of patient, **fødtår** - year of birth, **kjønn** - sex
nimesaktivitet	Each stay at the GMU	**ppid** - id of patient, **episodeid**- id of episode, **inndatotid** - start datetime, **utdatotid** - end datetime, **pdxkoder** - primary ICD-codes
aninoppgold	Each stay at the ICU	**ppid** - id of patient, **aninoppholdstart** - start datetime, **aninoppholdslutt** - end datetime
nsml	Laboratory tests	**ppid** - id of patient, **date_req** - date of the test, **matr_desc** - name of test, **micr_prt_name** - pathogen or NaN

Table 5.1.: Description of each table used for the data selection

with already suspected BSI. This may affect the result as we do not get a general case. It is also worth mentioning the level of details and type of data. It is hard to make a general structure for each episode, because each episode is different and include different data points. On the other side, the data lack detailed information about the stay, like journal notes and detailed demographics for the patients. Furthermore, It is important to address that the data is manually coded, which introduces the possibility of human error. This includes both subjective judgements and miscoding due to the platform. Lastly, as also mentioned in Section 2.1.2, the billing purpose of the ICD coding makes the codes less reliable.

5.2. Data Selection

In the data selection process, DBeaver and PostgreSQL were employed to efficiently and directly extract and manipulate the relevant data from the database (DBeaver Corp and contributors, 2023). Having direct access to the database offers several advantages. First, by querying the database directly, data extraction can be tailored to the specific needs of the clustering analysis, reducing the need for extensive preprocessing. The ability to manipulate the data directly at the source ensures that only relevant information is extracted. Second, direct database access allows for the efficient aggregation of features using PostgreSQL. This enables the rapid computation of summary statistics, and derived

variables directly within the database, saving both time and computational resources.

The goal of the data selection was to summarise the history of each patient while simultaneously reducing the dimensionality of the data. In this process, we made a series of decisions to effectively encapsulate the patients' histories. Firstly, the history was decided to be prior to the last blood culture taken. This means that for patients with more than one episodes with suspicion of BSI, only the last counted for the analysis. The choice of the features selected was based on the findings in the literature review presented in Section 3.1 and with the goal of maximizing available data.

Note that in the data selection, an episode is defined as a distinct stay at the hospital. This stay is either a stay at the GMU which may or may not have any associated ICU stays, or an individual ICU stay in cases where it is not associated with any GMU stay.

The first feature chosen was the number of episodes prior. This was chosen as it is an indicator of the frequency of distinct episodes, providing insights into the patterns of activity leading up to the last suspected episode. This number however does not account for the length of each visit, which may vary significantly. Hence, the total duration was chosen as another feature. This can provide insights into the severity or complexity of the patient's condition, as longer stays may be associated with more intensive or complex treatments. As mentioned in Chapter 2, a longer stay is also associated with a adverse patient outcome, so this feature could provide useful information when analysing with respect to prior research.

To also provide information regarding the level of intensity, the number of ICU stays as well as total ICU duration were also included. The choice of both count and duration can be defended by the same arguments as for the GMU episodes. With a suspicion that the outcome of a BSI can be related to the last hospital visit, the total duration within the last episode and the total duration since the last episode were included. Another important choice of feature was regarding the ICD codes. The provided dataset include both the primary and the secondary ICD diagnosis code, but to reduce the dimensionality for this research, only the primary codes were selected and collected as a list for each patient.

As the research aims to analyse risk factors in the patients history with respect to the outcome of a suspected BSI, it is also crucial to find features describing the time after the suspected episode. In addition to the result of the blood culture, the data selection also includes finding total duration in the post 60 days after the suspected episode. This way also the overall condition of the patient the following 60 days is taken into account. To provide demographics of each patient, the birthyear and sex were also included.

In order to extract the necessary features, a series of aggregations were performed. Specifically, select queries were used to identify rows of interest, and these were subsequently joined to gather the desired features. This process culminated in a fairly complex PostgreSQL query, the full details of which can be found in Appendix A.

The extracted features, along with the criteria for their selection, are detailed below. Please note that the term episodes here refers to the joined tables ***nimesaktivitet*** and ***aninopphold*** unless explicitly stated otherwise.

- **no_episodes_prior**: The count of distinct hashids from the episodes with a start

date before the date of the last blood culture.

- **total_duration_prior**: The sum of durations of all episodes with a start date before the date of the last blood culture.
- **total_icu_count**: The count of distinct records from ***aninopphold*** with a start date prior to the date of the last blood culture.
- **total_icu_duration**: The sum of durations from episodes in ***aninopphold*** that has a start date before the date of the last blood culture.
- **dur_since_last_ep**: The subtraction of the end date of the last episode prior and the last suspected episode from the date of the last suspected episode. The last episode prior to the last suspected episode is found as the episode with maximum date from the episodes where the start date of the episode is before the date of the last blood culture taken.
- **duration_last_ep**: The duration of the last episode that occurred prior to the last suspected episode.
- **icd_codes**: The list of all primary ICD-codes. This is derived from merging the primary ICD codes of each episode that has a start date before the date of the last blood culture.
- **micr_prt_name**: The pathogen found in the blood culture, or null if no pathogen is found. This is derived from merging the *micr_prt_name* of each blood culture record that shares the date of the last suspected episode and the *ppid* of the given patient.
- **total_duration_post_60d**: The sum of the duration of each episode with an end date after the date of the last blood culture and start date before the date of the last blood culture plus 60 days.
- **sex**: The sex of the patient, derived directly from the ***pasient*** table for the corresponding *ppid*.
- **birthyear**: The year of birth for the patient, also directly extracted from the ***pasient*** table for the corresponding *ppid*.

5.3. Data Preprocessing

The direct access to the database facilitated careful selection of relevant data, thereby reducing the need for extensive preprocessing. Initially, demographic data from 35,695 patients were available. However, one patient lacked any ordered blood culture and was thus excluded in the selection. The final dataset comprised the medical histories of 35,694 patients.

The handling of null values, e.g. patients with no prior history before the last suspected episode, happened already in the selection. Hence, the preprocessing phase could focus on preparing the existing columns for the clustering algorithm. This involved the following steps:

1. **Splitting diagnosis codes**. The selected data contained composite values for diagnosis codes within a single column. To facilitate analysis, we split these values to achieve a list of individual codes. This transformation made it easier to handle and analyze the diagnosis codes in our later stages of the research.

2. **Calculating age**. As the data provided the year of birth and the date of the last suspected episode, the age could be calculated by subtracting the years from each other.

3. **Discretizing post hospital stay**. We classified post-hospital stay into two categories: short and long. This binary representation was a prior step for making the outcome categories, as described in the next step.

4. **Creating outcome categories** We created four distinct categories for the outcome variable. This was achieved by considering two factors: the length of the post-hospital stay (short or long) and the result of the blood culture (negative or positive). The result of the blood culture was found by the column representing the pathogen found, *micr_prt_name*, where the presence of a pathogen proved a positive blood culture.

5. **Normalize numerical values**. Finally, we normalized the numerical features to ensure fair and accurate comparisons. Normalization transforms the numerical values to a common scale between 0 and 1, and eliminates any biases due to differing units and value ranges.

5.4. Exploratory Data Analysis

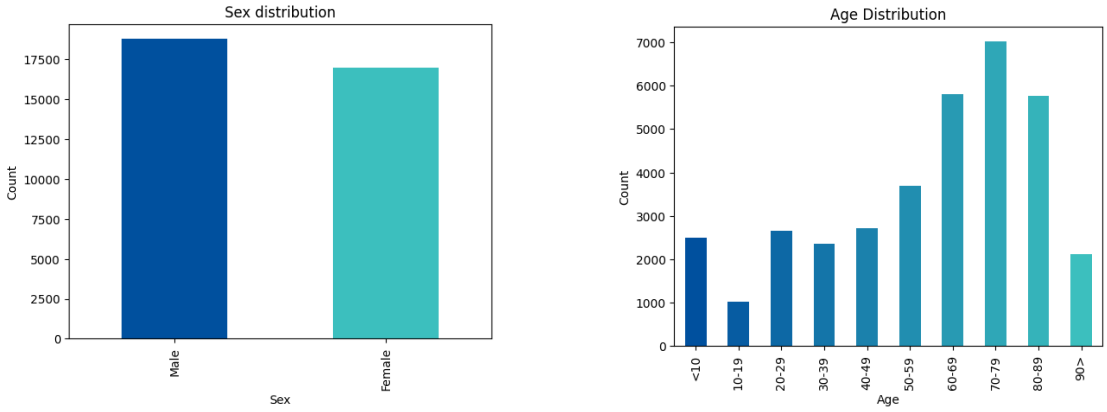
To better understand the dataset, the exploration part is crucial. This part will help gaining a deeper understanding of the data, and identify potential patterns and relationships. We will start by looking into statistics and characteristics of the selected data in general, before we will compare the data in the four outcomes.

5.4.1. Overview of the Selected Data

After the selection, we have the history and outcome of 35,694 patients with suspected BSI. History now refers to the attributes explained above, i.e. the number of episodes, total duration, the number of ICU stays, total ICU duration, duration of last episode, duration since last episode and all primary diagnosis codes given for the patient prior to the last suspected episode. The average and maximum values for each of the features are summarized in Table 5.2.

Feature	Average value	Maximum value
Number of episodes	25	1553
Total duration	31d	1430d
Number of ICU stays	1	121
Total ICU duration	2d	349d
Duration last episode	38h	129d
Duration since last episode	153d	3542d (>9y)

Table 5.2.: Average and maximum value for each historical feature selected



(a) Distribution of sex

(b) Distribution of age

Figure 5.1.: Distribution of sex and age in the general cohort

The distribution of sex in the general cohort is quite even, with a slight predominance of male patients, while the age distribution of the dataset is skewed towards older individuals. The largest proportion of participants are aged 70-79, with 19,64% of all patients. The patients with the youngest age when an episode happened were not even 1 year, while the oldest was 105 years old. Visualization of the distribution of the sex and the age can be shown in Figure 5.1a and Figure 5.1b, respectively.

Within the episodes, the diagnosis with the most occurrences is ICD Z491, signifying *Extracorporeal dialysis*. The high frequency of this code can be attributed to the routine nature of dialysis procedures for patients with kidney failure, where a blood culture is automatically collected. Consequently, each dialysis procedure is counted as a *suspected episode* in our definition, regardless of an actual BSI suspicion. Considering that dialysis procedures are commonly performed three times a week, it is understandable why this diagnosis contributes prominently to the overall distribution.

The distribution of the top 20 diagnoses within the selected data is shown in Figure 5.2.

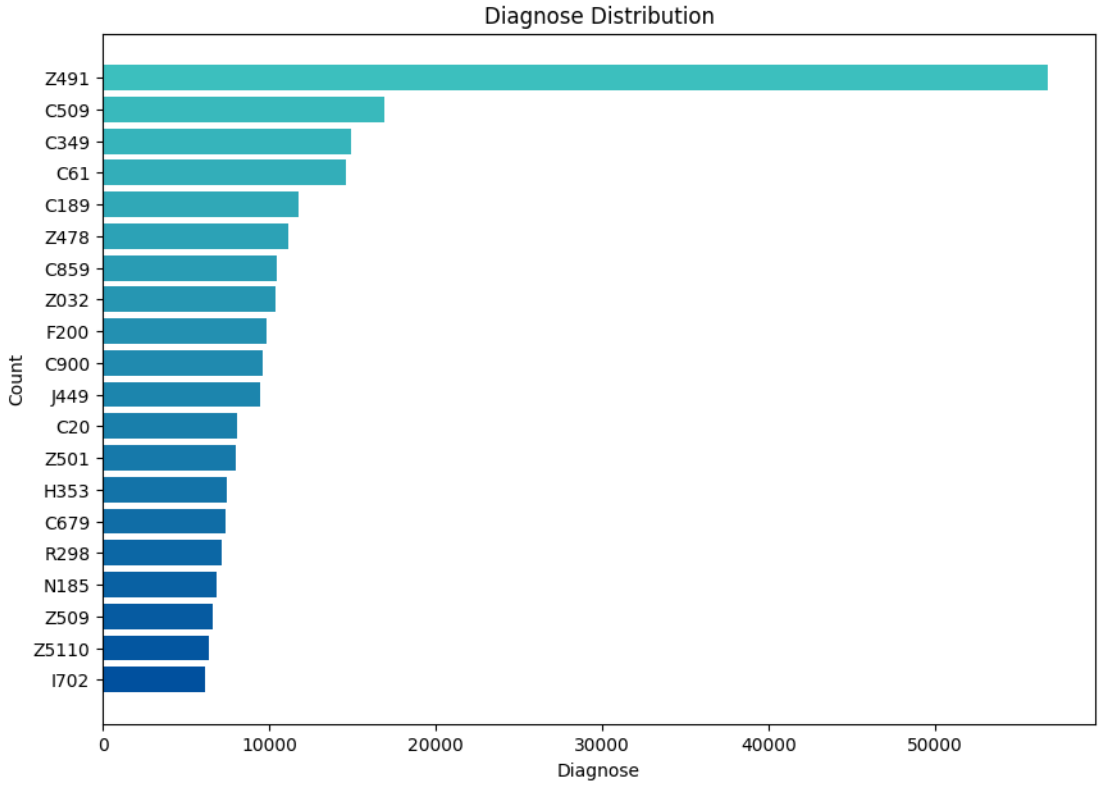


Figure 5.2.: Distribution of top 20 diagnoses in the general cohort

Due to the overrepresentation of Z491, a separate figure, Figure 5.3, presents the distribution of other diagnoses. This alternative visualization facilitates a better understanding of the frequency and diversity of conditions across the patient population.

5.4.2. Plots of Outcomes

When examining the results from the last suspected episodes' blood culture, we found that 3012 patients tested positive, indicating a BSI. This represent 8,44% of the total episodes. However, this study does not only consider the number of patients with a positive result, but also accounts for their subsequent hospital stay duration. Hence, the analysis will focus more on the patients' outcome, divided into groups based on the blood culture results and the total hospital stay duration within 60 days after the suspected episode. The grouping resulted in four categories:

- **NS**: Patients with **N**egative blood culture and **S**hort hospital stay
- **NL**: Patients with **N**egative blood culture and **L**ong hospital stay
- **PS**: Patients with **P**ositive blood culture and **S**hort hospital stay

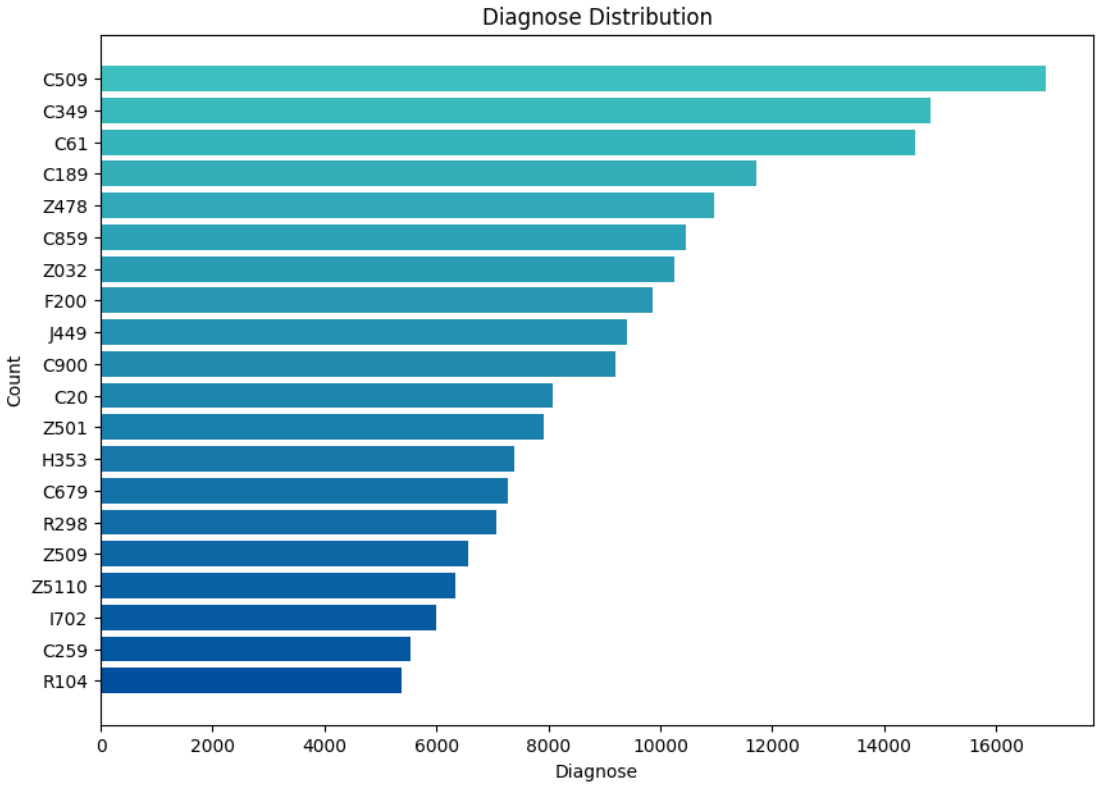


Figure 5.3.: Distribution of top 20 diagnoses in the general cohort, without Z491 - Extracorporeal dialysis

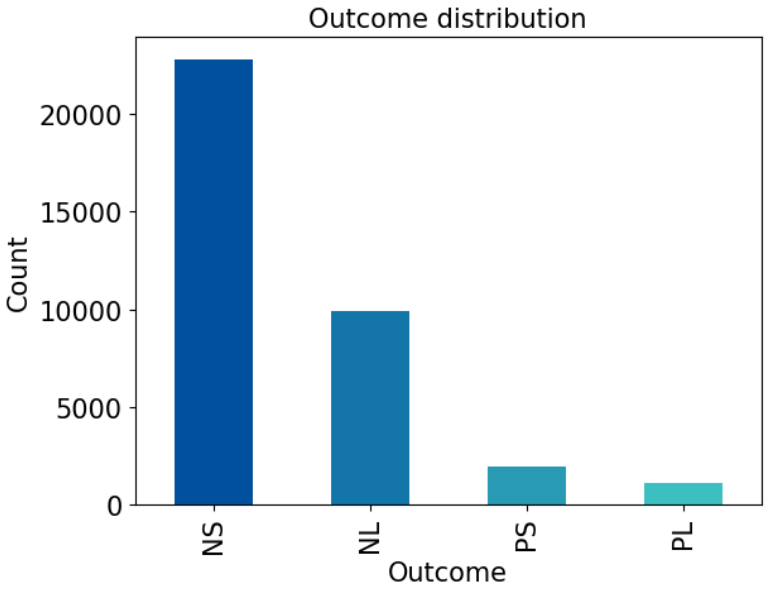


Figure 5.4.: Distribution of outcomes in the general cohort

- **PL**: Patients with **P**ositive blood culture and **L**ong hospital stay

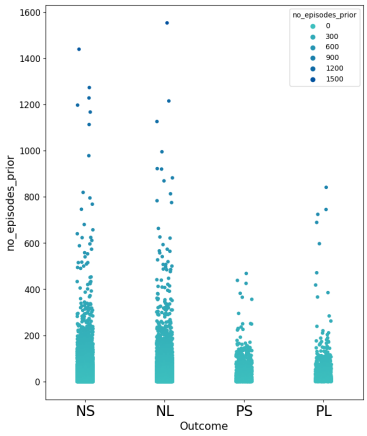
Figure 5.4 shows the distribution of these different outcomes.

As expected, the NS group - including patients with a negative blood culture and short subsequent hospital stay - contains the majority of the patients. The patients with positive outcomes are distributed between the two groups, PS and PL, with a noticable inclination towards those with a shorter hospital stay. This indicates that a positive blood culture not necessarily indicates an extended hospital stay. Further investigation is required to uncover more granular patient group details and better comprehend these patterns.

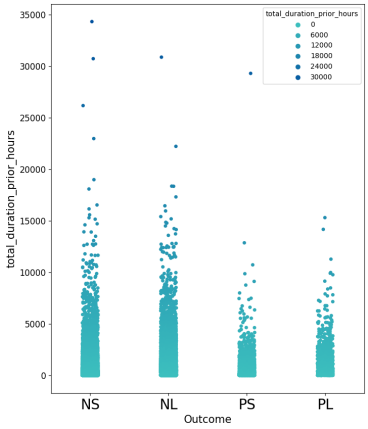
To achieve his, features from the medical history will be explored within each group, which should provide valuable insights. Figure 5.5 shows scatter plots of all outcomes against different aggregated attributes, while Figure 5.6 and Figure 5.7 show the age and sex distributions within each group, respectively.

Evaluating these scatter plots reveals some interesting findings. Before delving into the details of the plot, it is crucial to remember the uneven patient frequencies across the different groups. With the largest patient proportion resulting in a negative blood culture, it is expected that these groups contribute the most outliers and have records on a larger scale.

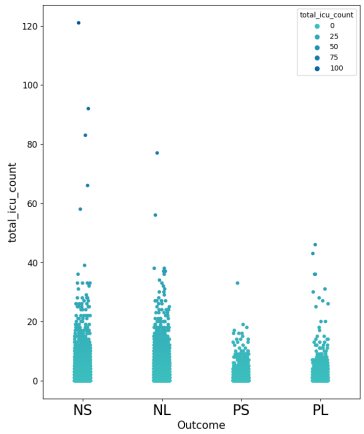
The two first plots hint a trend of higher numbers of episodes and total duration in patients that did not have a BSI. However, considering the PL group's smaller size, the two plots reveal a relationship between a greater number of episodes and total duration and a long positive outcome. Nonetheless, this finding cannot be used as an indicator because both the NS and NL groups also demonstrate higher number for these attributes.



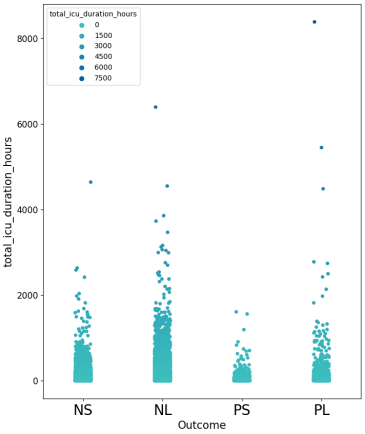
(a) Distribution of number of episodes for each outcome



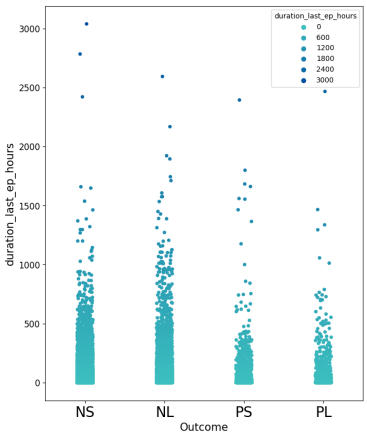
(b) Distribution of total duration prior for each outcome



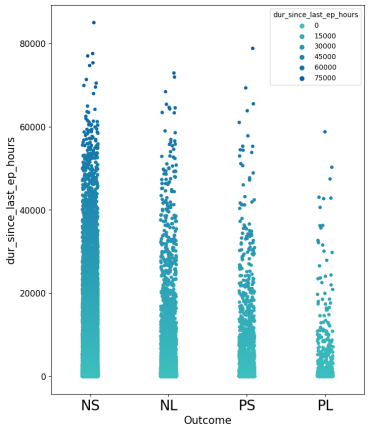
(c) Distribution of total ICU count for each outcome



(d) Distribution of total ICU duration for each outcome



(e) Distribution of duration last episode for each outcome



(f) Distribution of duration since last episode for each outcome

Figure 5.5.: Distribution of different attributes for each outcome

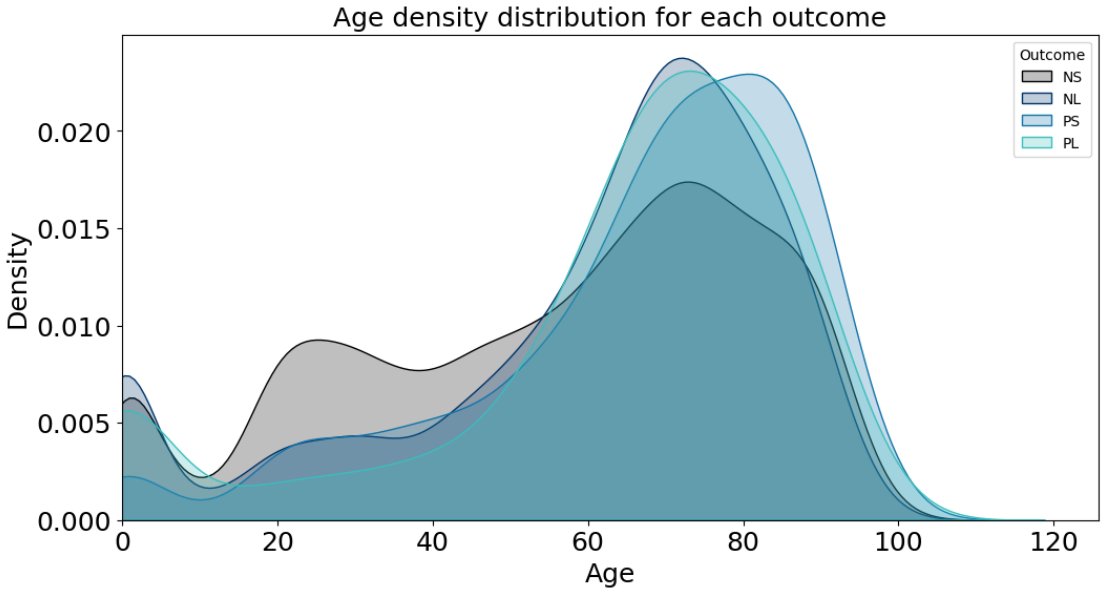


Figure 5.6.: Distribution of age for each outcome

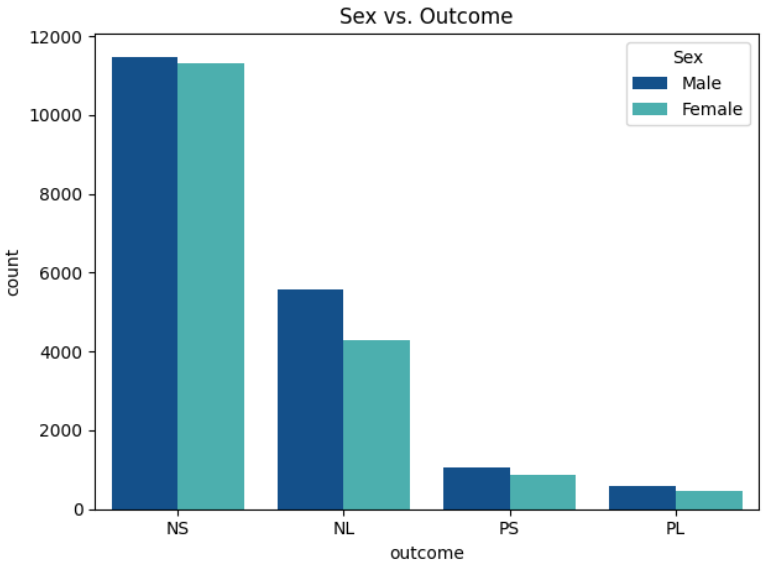


Figure 5.7.: Distribution of sex for each outcome

The distribution of total ICU count also seems related to a longer positive stay, but also longer negative stay. This finding is emphasized when looking at the total ICU duration, where a longer prior ICU duration indicates a longer following hospital stay. The plot of duration in the last episode reveals somewhat the same findings, where a long last episode could be related to a longer subsequent hospital stay.

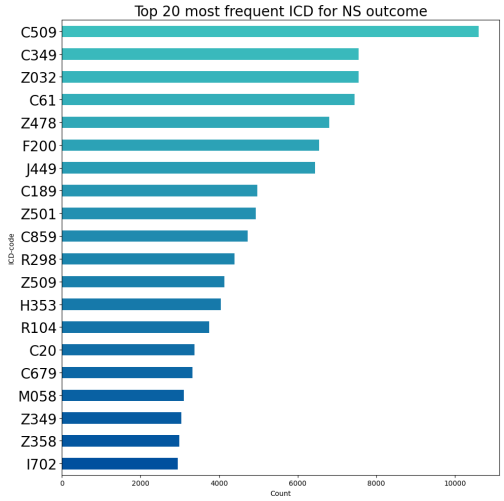
All the mentioned plots show more general relations between the attributes and the length of following hospitalization, and not findings related to the outcome of the blood culture. This could however be found in the last plot, showing the duration since last episode. In this relation, we can see that the group with negative blood culture and short hospital stay have a higher density on the upper scale. This means that patients with a long break since last episode tend to end with negative blood culture and short hospital stay following the test, and a short gap indicates a higher risk of infection with following long stay.

The distribution of sex did not yield any notable findings, as all groups exhibited a larger male patient proportion. However, the age distribution deserves closer attention. The NS group presents a more evenly distributed age density, with a notable peak in the twenties that exceeds the other groups. This group also shows a lower density of elderly patients. The age density in the PS group skews right, suggesting that older patients may have an increased risk of a positive blood culture, but a shorter subsequent hospital stay. The NL and PL group plots appear remarkably similar, hinting that age may be more closely associated with the duration of the subsequent stay than with the blood culture results.

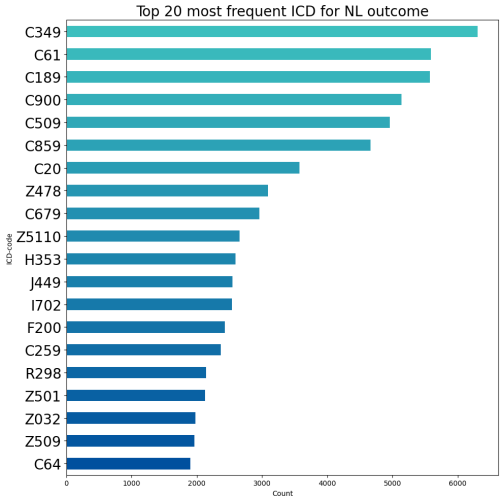
Moving further to analysing the historic diagnosis in each group, Figure 5.8 shows the top 20 most frequent ICD codes across the four outcome groups. Considering the marked overrepresentation of Z491, a characteristic that persists even when examining separate outcomes, this particular diagnosis code is again excluded from the plots.

The plot of the ICD codes for each group is only intended to be used as a distribution overview and not number of occurrences, as the x-axis is significantly different between the groups. When excluding the Z491, the specific diagnose being most frequent varies across groups, but all of them are a part of the same level "C00-C97: Malignant Neoplasms" indicating cancer. An interesting finding includes the position of C349, *Malignant neoplasm of unspecified part of bronchus or lung*, being at position 2 and 1 for the negative groups and 8 and 13 for the positive groups. This could indicate a relation between C349 and a somewhat decreased risk of BSI. The diagnose code C900, *Multiple myeloma*, appears high in all plots except for the NS group, where it is not present at all in the top 20. The presence of this diagnose, which is a type of bone marrow cancer, appears to indicate either a positive blood culture or an extended following hospital stay, or both. Another interesting finding is the appearance of C259, *Malignant neoplasm of pancreas, unspecified*, which is significantly higher in the plot for PL group compared to all the other groups. This could indicate a higher risk of BSI and a long subsequent hospital stay when diagnosed with pancreatic cancer.

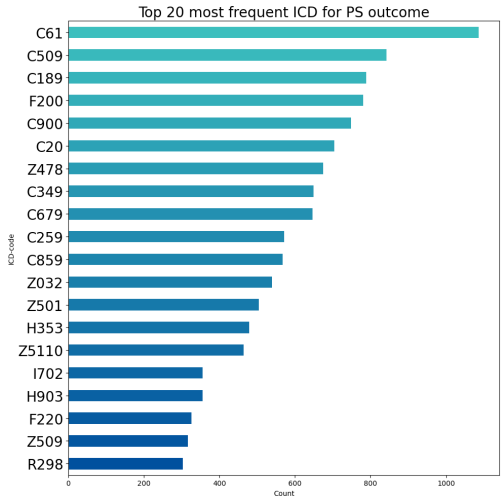
Both PL and NL have exclusively cancer diagnosis on top 5, while the top 5 for PS also includes F200, *Paranoid schizophrenia*. The top 5 for NS group differs with having



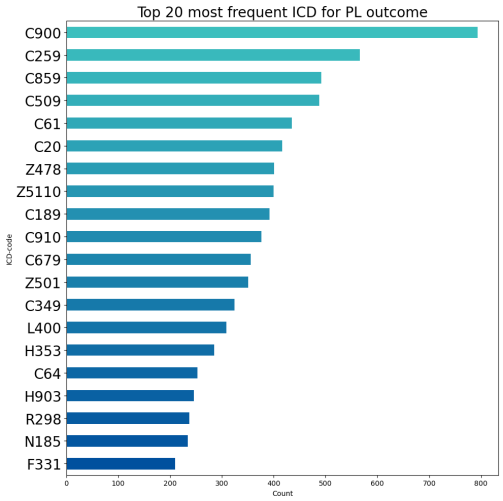
(a) Distribution of top 20 diagnoses for NS outcome



(b) Distribution of top 20 diagnoses for NL outcome



(c) Distribution of top 20 diagnoses for PS outcome



(d) Distribution of top 20 diagnoses for PL outcome

Figure 5.8.: Distribution of top 20 diagnoses for each outcome, without Z491

Z043, *Encounter for examination and observation following other accident* and Z478, *Encounter for other orthopedic aftercare.*

These findings show a trend that a blood culture is often ordered for patients with cancer in general, but there is no clear relation to a specific diagnose code and a positive blood culture, except for the already stated C259.

Moving further in the analysis, it is interesting to look at the specific pathogens for each positive group. The distribution of the top pathogens for each positive outcome is shown in Figure 5.9. An aggregated version that separates each pathogen to its belonging type, gram negative or gram positive, is shown in Figure 5.10

These findings partly support the theory introduced in Section 2.1.1, where E.coli and S. Aureus were identified as the most and second most prevalent pathogen, respectively. The position of S. Pneumoniae in our dataset however differs from the theory. Looking at the general category of the pathogen, there is a noticeable increased amount of gram positive bacteria compared to the gram negatives for the PL group. The gram-positive S.aureus can contribute to this bias, which is the only pathogen with significantly more occurrences in the PL group. A finding of this pathogen could hence indicate an increased risk of a prolonged hospital stay as a consequence of BSI.

In summary, key findings drawn from the outcome plots include a noticeable correlation between prolonged prior hospitalization - either at the GMU or ICU - and a subsequent extended hospital stay. When it comes to age, the most striking connection seems to be between younger patients and a short negative outcome. Patients with a cancer diagnose are frequently suspected of BSI, but there is only a significant correlation between the risk and pancreatic cancer. Furthermore, a positive blood culture with a growth of S. aureus might serve as a potential indicator of a longer hospital stay.

5.4.3. Comparing Numerical Characteristics

The visual representations in the previous section are valuable as they provide an intuitive way to understand and compare the different groups, highlight trends and quickly identify outliers. However, the nature of these plots sometimes prevents the precision required to identify specific values or discern subtle differences between groups. Therefore, the following section aims to delve into an examination of the numerical characteristics of each group. By using both of these analysis methods - visual and numerical - a more comprehensive and accurate exploration of the patient results is ensured. The exploration of the numerical characteristics is summarized in Table 5.3.

The characteristics emphasize all the findings from the plot. The mean age is youngest for the NS group, and oldest for patients in the PS group. The amount of males does not reveal much interesting, except being a bit smaller for the NS group. The number and duration of both GMU and ICU episodes confirm the earlier findings, with a prolonged prior stay resulting in a prolonged post stay, and the mean duration since last episode confirms that a shorter gap results in a prolonged stay.

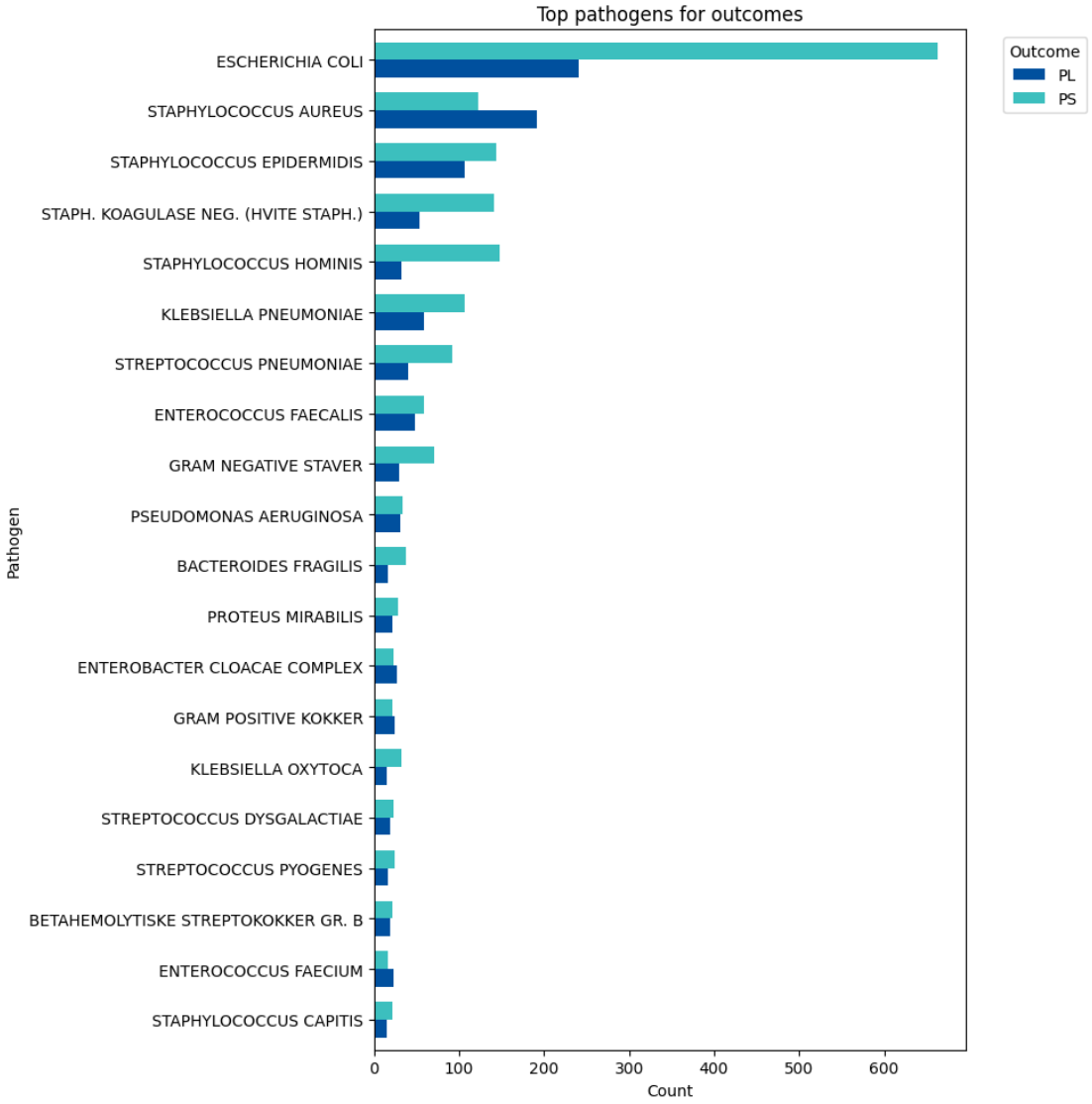


Figure 5.9.: Distribution of top pathogens in the groups with positive BC

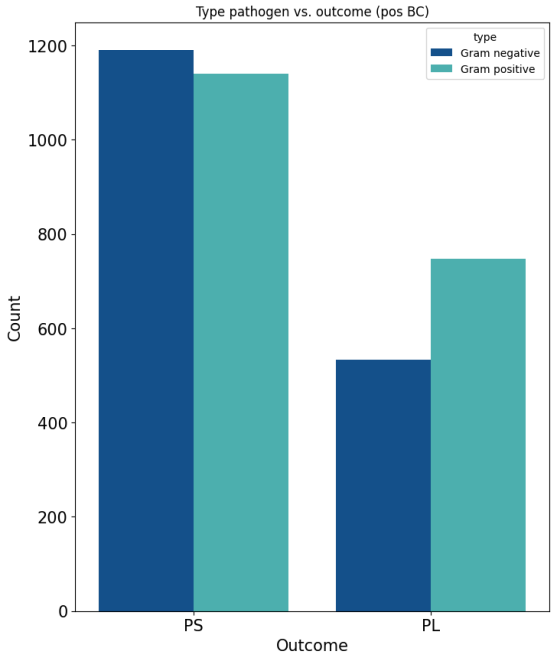


Figure 5.10.: Distribution of type of pathogens in the groups with positive BC

	All	**NS**	**NL**	**PS**	**PL**
Number of patients	35,694	22,794	9,888	1,943	1,069
Mean age	58	56	60	67	63
Amount male	52,5%	50,4%	56,5%	55,0%	55,8%
Mean no. of episodes	25	22	32	24	34
Mean total duration	31d 19h	22d 1h	52d 15h	27d 13h	55d 1h
Mean no. of ICU stays	1.43	1.08	2.19	1.24	2.38
Mean total ICU duration	2d 0h	0d 22h	4d 9h	1d 2h	4d 23h
Mean duration last episode	1d 15h	1d 8h	2d 5h	1d 19h	2d 9h
Mean duration since last episode	153d	182d	87d	185d	85d

Table 5.3.: Numerical characteristics across outcomes

5.5. Environments

The primary data environment employed in this research is HUNT Cloud, a cloud-based service that provided a dedicated lab for this study (HUNT Cloud). The laboratories within HUNT Cloud are digital environments with allocated cloud resources for storage, computation and data transfer. The utilization of this lab allowed for conducting the research process in a secure environment, facilitating the management of sensitive data in compliance with privacy and security regulations.

DBeaver, a comprehensive and widely used database management tool (DBeaver Corp and contributors, 2023), was utilized for the data selection process. DBeaver offers a user-friendly interface supporting a range of databases, including Postgres, which served as the database system in this study. It was was connected to the remote machine on the HUNT Cloud via a Secure Shell (SSH) connection, facilitating secure interaction with the remote database as if it was locally hosted, thereby enabling secure query execution and data retrieval.

Furthermore, access to the HUNT Workbench was granted through HUNT. This web-based workbench offers a range of tools including Jupyter Notebook, Python and Conda. Visual Studio Code (VS Code), a powerful code editor supporting various programming languages (Visual Studio Code, 2023), was also employed. The integration of VS Code with the workbench facilitated a more efficient and familiar coding environment. The VS Code extension *Remote - SSH* was leveraged to establish a secure connection with the remote machine. This simplified the process of working with the data in a familiar environment while maintaining the secure connection.

In addition to the software tools and resources, the research environment was further enhanced by the inclusion of a NVIDIA Tesla P100 GPU machine. GPUs are capable of handling complex calculations in parallel, resulting in significantly faster performance for computationally intensive tasks.

5.6. Agreements and Approval

Prior to the commencement of any medical or health related research in Norway, it is required to obtain a pre-approval from the Regional committees for medical and health research ethics (REK) (National Research Ethics Committees, 2019). This approval process ensures that all research is conducted in line with relevant laws, regulations and guidelines. The project that this research is a part of, with Lise Tuset Gustad as the project leader, was reviewed and approved by REK with the case number 2018/1201.

Furthermore, before getting access to the data, the HUNT Cloud User Agreement and a Non-Disclosure Agreement (NDA) needed to be signed. The HUNT Cloud User Agreement aims to clarify expectations and responsibilities regarding data handling, software usage and security management. The NDA ensures that sensitive and confidential information remains protected and undisclosed to unauthorized individuals. The legal basis of the agreement include The Health Research Act *(Helseforskningsloven)* §7 , The Universities and Colleges Act *(Universitets- og høyskoleloven)* §4-6, Act on Healthcare

Personell *(Lov om helsepersonell)* §21, The Health Register Act *(Helseregisterloven)* §17, The Public Administration Act *(Forvaltningsloven)* §13 and The Penal Code *(Straffeloven)* §209 and §210.

6. Experiments and Results

This chapter delves into the detailed execution and results of the experimental part of the study, building upon the data selection and preprocessing as outlined in Chapter 5. It begins with an elaboration of the preliminary experiments in Section 6.1, which describes the implementations and limitations of MASPC and DDSCA. It is important to remember that these descriptions are given in retrospect, and is elaborated in a compact way because of the early realisation that the experiment were not suitable for the specific context. The insights gained from these preliminary experiments are further used in the next experiment, detailed in Section 6.2

6.1. Preliminary Experiments

6.1.1. Experimental Plan

The preliminary experimental plan serves as an initial step towards our main goal. It seeks to assess the feasibility and relevance of existing methods in addressing our research questions, providing valuable insights into the suitability of these methods for our unique dataset. The steps are prepared in order to follow a structured approach towards the goal, and are as follows:

1. **Implement MASPC on the selected data**. In order to test the MASPC algorithm, the first step is to implement the algorithm to fit our dataset.

2. **Identify limitations and challenges with MASPC**. Throughout the implementation of the algorithm, any significant limitations and challenges encountered will be identified. This identification is a part of the evaluation step in the DSR approach.

3. **Implement DDSCA on the selected data**. After implementing and discarding MASPC, the next step is implementing the DDSCA algorithm.

4. **Identify limitations and challenges with DDSCA**. As for the MASPC, limitations and challenges encountered during the implementation of DDSCA will be identified, as a part of the evaluation of the experiment.

5. **Address the insights gained**. After the two experiments, an overall evaluation will be conducted. The experiences and insights gained will be used to guide the design and implementation of the SASCA algorithm, and includes potential modifications, improvements or aspects to be cautious about.

6.1.2. Implement MASPC on the Selected Data

The implementation of the initial preliminary experiment, MASPC, was done utilizing Python. Considering MASPC's requirement for a binary data representation, we refrained from normalizing values during preprocessing. Instead, numerical columns were discretized before applying one-hot encoding via pandas' `get_dummies`.

Next, the MAS phase code, which included MFA mining and pattern selection via Apriori and FPMAX, was refactored from the original Java subprocess within Python to exclusively use Python's `mlxtend` package (Raschka, 2018). This modification was necessary given the alphanumeric characteristics of ICD-10 codes, differing from the numeric structure of ICD-9 codes.

Following these adjustments, the algorithm was ready to run on our data set.

6.1.3. Identify Limitations and Challenges with MASPC

After solving the initial challenges faced when adjusting and implementing MASPC, and the code was ready for our dataset, we realised that the algorithm could not fulfill our research questions. Since the basis of the clustering relies on sets of diagnosis codes in the history, the algorithm is not able to find history where one specific diagnose can imply an increased risk of BSI. This approach would have been more suitable if the goal was to discover all kinds of relationships in the history of diagnosis codes or if the diagnose to be analyzed actually was coded in the history, which is not the case for BSI in our dataset.

6.1.4. Implement DDSCA on the Selected Data

Proceeding to the implementation of DDSCA, this phase also required some adjustments. The first stage involved recreating the textual representation of the ICD-hierarchy tailored for ICD-10 codes, as the original source code was built for ICD-9. This process was far from straightforward, requiring a semi-manual approach involving content copied from e-Helse's platform for finding codes (Direktoratet for e-helse medisinske kodeverk). This content was subsequently subjected to computational processing using Python. The objective was to generate a text file where each line represents the hierarchical path from an individual ICD-code to the root node A00-Z99.

To provide an insight into the structure and content of the created file, a sample excerpt is presented below.

```
A000 A00 A00-A09 A00-B99 A00-Z99
A001 A00 A00-A09 A00-B99 A00-Z99
```

This text file served as the basis for creating the hierarchy, represented as a direct tree using `networkx`'s DiGraph (Hagberg et al., 2008). The final tree consisted of A00-Z99 as a root node representing all codes, with all levels as internal nodes, and each distinct ICD code represented as leaves. This structure represent the same as in Figure 2.1. Subsequently, the distance matrix representing the distance between each pair of codes was constructed as per the original code, but adjusted to our newly created tree.

Having prepared the ground, we proceeded to construct the RS-Tree. Upon investigating the original code, we discovered that the logic behind the edges applied in the graph representation post-dendrogram construction was unclear. Seeing the chosen edges as somewhat arbitrary, we decided to introduce additional code to transpose the dendrogram's connections to the graph representation, thereby preserving the same edges between the vector of demographics values and the clusters.

With no further adjustments needed from the original source code, we were ready to execute the code on our dataset.

6.1.5. Identify Limitations and Challenges with DDSCA

During the implementation and execution of DDSCA, several limitations and challenges emerged. In the original dataset, the average number of ICD-codes per patient was 9.2121, with a maximum of 46. These numbers were significantly larger in our dataset, with an average of 26 and a maximum of 1462 codes. Moreover, when using all desired features describing the patients' history, the number of single values increased markedly from the original dataset to ours. These factors considerably escalated the complexity. The complexity stemmed from the similarity measure, which entailed the complicated Weighted Levenshtein for the diagnoses and the distance measure of the complex DiGraph for all the different values of demographics vectors. This resulted on not being able to fully run the algorithm on our dataset, as it was way too time-consuming.

6.1.6. Address the Insights Gained

The execution and critical evaluation of these two preliminary experiments revealed significant insights, shaping the direction of our subsequent research. Although MASPC is proved to find correlated codes, the lack of BSI coding in our dataset made it unable to address our research goals. This limitation underscored the need for a more tailored algorithm that could leverage the existing dataset structure effectively and align with our specific research goals.

The DDSCA experiment emphasized the importance of considering the computational feasibility of the chosen algorithm. Despite the theoretical appeal of DDSCA, its practical implementation faced challenges due to the substantial time required for distance computation given our large dataset. Our research does not necessitate considering the order of diagnoses, thus preserving the conceptual value of this algorithm while further developing its implementation to reduce complexity would be beneficial.

6.2. SASCA Experiment

6.2.1. Experimental Plan

As for the preliminary experiments, the experimental plan aims to provide a structured approach towards achieving the goal of this research. The reader is advised to revisit the

research goal and questions stated in Section 1.2. The experimental steps for the main experiment are as follows:

1. **Implement SASCA on the selected data**. The first step is to implement the newly developed clustering algorithm, SASCA, on the prepared dataset. This involves executing the algorithm and validating that it is functioning as expected.

2. **Parameter optimization and selecting the number of clusters**. Following a successful implementation of the algorithm, the second step focuses on tuning the parameters of the SASCA and determine the optimal number of clusters, k.

3. **Analyse the cluster results**. Upon achieving optimal clustering, the third step is to carry out an analysis of the cluster results. This analysis will aim to understand the characteristics of each cluster in a clinical context, and identify meaningful patterns within the patient histories.

4. **Relate cluster results to patient outcomes**. The next step is to relate the characteristics of each cluster to the associated patient outcomes. This will involve analyzing patterns within and between clusters, and correlating these with patient outcomes in order to identify potential risk factors for BSI and for increased following hospital stay.

5. **Validate methodology and results with clinicians**. The last step of the experiment is to validate the methodology and clustering results with clinical professionals. This involves presenting the methodology, the derived patterns, potential risk factors and other significant findings to clinicians for review. This validation is crucial, as it will address the relevance and feasibility of the clustering.

In the following sections of this chapter, each of these steps will be addressed in detail, describing the experimental setup and the achieved results for each step.

6.2.2. Implement SASCA on the Selected Data

Experimental Setup

In order to calculate the distance between each ICD-code, as also required by SASCA, the construction of the hierarchical representation was reused from the DDSCA implementation, detailed in the preceding section. This tree then served as the foundation for creating a distance matrix for the normalized JC-distances between pairs of codes. Given the computational complexity associated with calculating distances for all unique codes, we reduced the number of codes from 1487 to the 200 most frequent ones. This decision was informed by an analysis of the frequency distribution of the codes in the dataset. A plot of this distribution indicated that the chosen 200 codes encapsulated a substantial proportion of the data variability, covering 80% of the diagnoses in the history.

The subsequent step involved creating a separate normalized distance matrix for the single values' distances. This was accomplished by computing the Euclidean distance between each vector representing each patient's single values.

With these distance matrices prepared, the next step was the implementation of the k-centers algorithm. We initially set the weights $w_{single} = w_{set} = 0.45$ and $w_{outcome} = 0.1$, and the number of clusters k to 20, to establish a baseline for the clustering process. This was chosen for the sake of implementation and does not necessarily represent an optimal configuration. The process of optimizing these values is addressed in the next step. This served as a starting point from which we could begin our iterative process of identifying cluster centers. The first center was chosen randomly, and subsequent centers were identified following the algorithm detailed in Algorithm 1. Finally, we formed clusters by assigning each patient to the cluster that had the closest center. The complete Python code for SASCA can be found in Appendix B.

Results

With this implementation, we were able to successfully generate clusters based on the patient history. An analysis of these initial clusters was not conducted at this stage, but the successful implementation signaled that we could move further to the parameter optimization.

6.2.3. Parameter Optimization and Selecting the Number of Clusters

Experimental Setup

Once the initial implementation of SASCA was complete, the next phase of this study involved optimization of the weights and the number of clusters, k. The initial plan included using the CH index for optimizing both the number of clusters and weights. However, preliminary results indicated that the CH index was not effective in determining the optimal number of clusters. Consequently, we modified our approach to use the Within-Cluster Sum of Squares (WCSS) for this purpose.

In order to find the optimal k, SASCA was run with varying sets of weights. Subsequently, WCSS was computed for every tenth k up to $k = 100$ for each set of weights. Then, the *Elbow method*, which identifies the point at which additional clusters do not significantly improve the compactness, was employed to determine the optimal k. This was deemed as the *elbow* of the plot, illustrating the trade-off between the number of clusters and the within-cluster dispersion. This plot of the WCSS for each set of weights could also hint to the optimal value for the weights, even though the CH index would provide more valuable findings.

Hence, after finding the optimal k, the next step was to optimize the weights assigned to single, set, and outcome values. We systematically varied these weights while maintaining a balance among them and ran SASCA multiple times with each set of weights. This strategy allowed us to assess the impact of different weight distributions on the clustering results and to pinpoint the optimal weighting scheme.

To ensure unbiased results, given the random selection of the initial center, each weight configuration was tested three times. For each set, we computed the mean CH index over these three trials. The weight set yielding the highest mean CH index was selected as

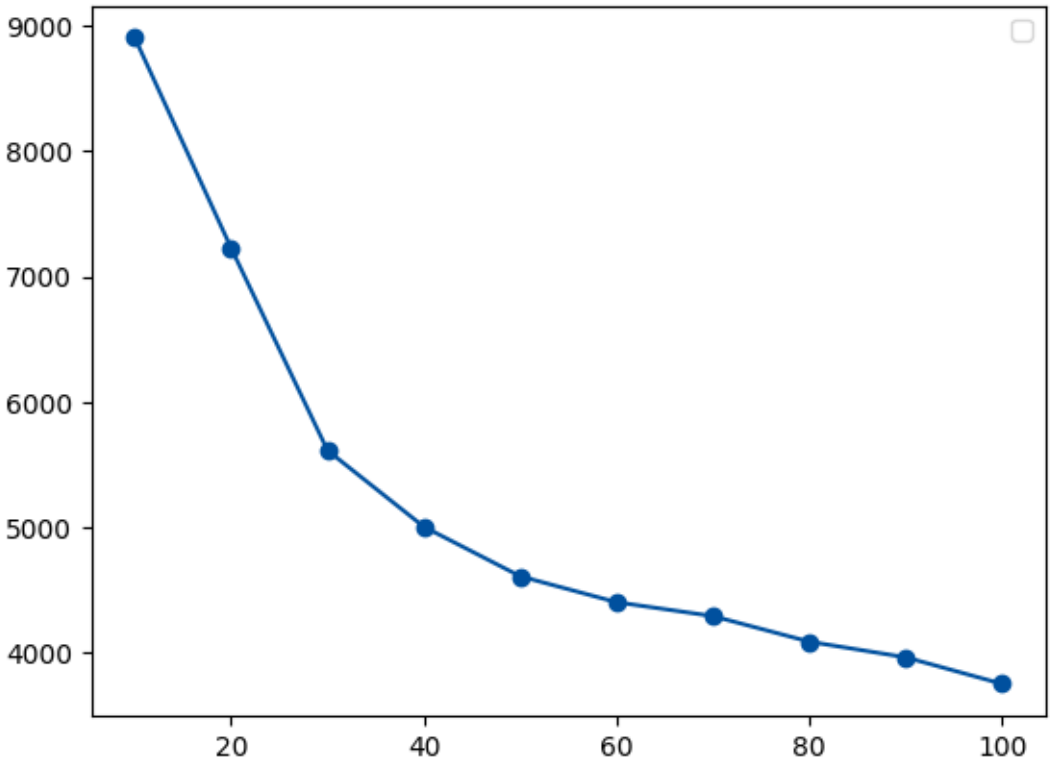


Figure 6.1.: Finding the optimal number of clusters by using the Elbow plot, with $w_{single} = w_{set} = 0.45, w_{outcome} = 0.1$

the optimal weights for our clustering algorithm.

Results

When initially using the CH index to find the optimal value for both clusters and weights, it was observed that the CH index reached a peak at k=2 and then declined significantly and stabilized at a low level. This peak at k=2 did not seem to be representative or appropriate given the large dataset size exceeding 30,000 patients. Furthermore, the subsequent stability of the CH index provided limited information about the optimal number of clusters.

Changing the strategy and moving on to calculating the WCSS for each k, the plot provided more valuable insights. For most of the set of weights, the elbow suggested an optimal value of $k = 30$. The plot for weights $w_{single} = w_{set} = 0.45, w_{outcome} = 0.1$ is shown in Figure 6.1.

Another interesting finding in the elbow plot was the observed higher value of CH index when the single values were assigned greater weight than the set values. This suggested a lack of significant similarity between two sets of diagnosis codes. To further

Weights			Mean CH index
Single	**Set**	**Outcome**	
0.5	0.5	0.0	2479
0.45	0.45	0.1	2622
0.4	0.4	0.2	2675
0.4	0.5	0.1	2606
0.35	0.45	0.2	2631
0.3	0.5	0.2	2694
0.5	**0.4**	**0.1**	**2790**
0.45	0.35	0.2	2508
0.5	0.3	0.2	2648

Table 6.1.: Mean CH index for different sets of weights when k = 30

examining the optimal values for the weights, the CH index was calculated for three iterations with $k = 30$ for each set of weights. The mean value of the three iterations for each set of weights are shown in Table 6.1, and suggest an optimal weighting of $w_{single} = 0.5, w_{set} = 0.4, w_{outcome} = 0.1$.

The further steps will utilize the clusters formed using these optimal weights and the optimal value of $k = 30$.

6.2.4. Analyse the Cluster Results

Experimental Setup

After finding the optimal values of both the weights and the number of clusters, the next step was to conduct an analysis of the cluster results formed with this optimization. This involved a review of the characteristics of the clusters formed by the SASCA. For each cluster, the demographics of the patients (i.e. age and sex) will be examined, as well as the distribution of single values describing the history and the most common ICD codes within each cluster. As the similarity between the set ICD codes is based on the hierarchical nature of the codes, it is not expected to achieve clusters with a significant amount of only one diagnose. Hence, it makes sense to investigate the codes further. This is done by producing a heatmap of the ICD chapters represented in each of the clusters. To also facilitate a visual overview of the numerical features, a heatmap for the mean value of each feature in each cluster will be produced.

Results

The optimization process revealed a total of 30 clusters, each varying in size. The smallest cluster, cluster 30, consists of only 8 patients, while the biggest, cluster 1, consists of 8281 patients. Each of the clusters has at least one value that deviates significantly from the average. In an attempt of explaining how each feature in each cluster differs from the average, each value is given a label of either very low (VL), low (L), average (A), high (H) or very high (VH). The value range corresponding to each label is given in Table 6.2. These values are used as a basis for the overall summary, provided in Table 6.3. This table outlines the key characteristics of each cluster, including the size of the cluster, average age of patients, ratio male, scaled values of the numerical features, and the top three most common ICD codes.

Feature	**VL**	**L**	**A**	**H**	**VH**
1- No. of episodes	0-10	10-20	20-30	30-40	40+
2- Total duration	0-12d	12-25d	25-35d	35-50d	50+
3- No. of ICU stays	0-0.6	0.6-1.2	1.2-1.8	1.8-2.4	2.4+
4- Total ICU duration	0-20h	20h-1d 15h	1d 15h-2d 10h	2d 10h-3d	3d+
5- Duration since last episode	0-60d	60-120d	120-180d	180-210d	210d+
6- Duration last episode	0-15h	15h-1d 5h	1d 5h-2d 1h	2d 1h-2d 15h	2d 15h+

Table 6.2.: Values for the numerical features for each label

No.	**Size**	**Av. age**	**Amount male**	**Num. features**						**ICD codes**
				1	**2**	**3**	**4**	**5**	**6**	
1	8281	61.0	93.2%	H	H	H	H	A	A	1. Z491 (13%), 2. C61 (3%), 3. Z501 (2%)
2	1693	29.0	1.42%	VL	VL	VL	VL	A	VL	1. None (40%), 2. Z113 (2%), 3. Z640 (2%)

3	1596	75.0	5.14%	A	H	A	A	H	VH	1. J449 (8%), 2. Z491 (5%), 3. J159 (4%)
4	418	69.0	100.0%	H	H	H	H	VL	VH	1. C189 (14%), 2. C61 (6%), 3. C20 (6%)
5	515	44.0	100.0%	L	A	A	VH	VH	L	1. C910 (5%), 2. R298 (5%), 3. R522 (5%)
6	374	66.0	100.0%	VL	L	L	A	VH	A	1. K805 (5%), 2. K802 (5%), 3. K800 (5%)
7	191	35.0	0.0%	VH	VH	VH	L	VH	L	1. F603 (19%), 2. Z032 (7%), 3. F431 (5%)
8	718	69.0	18.52%	H	A	A	A	A	A	1. H903 (12%), 2. H353 (11%), 3. H905 (5%)
9	277	75.0	3.25%	A	H	A	A	A	VH	1. L400 (17%), 2. L208 (5%), 3. Z478 (3%)
10	761	78.0	90.67%	A	A	A	H	A	H	1. H353 (8%), 2. H258 (5%), 3. H401 (3%)
11	167	46.0	94.61%	A	L	A	L	VH	VL	1. L400 (16%), 2. C910 (11%), 3. L309 (10%)
12	360	48.0	88.89%	L	A	A	VH	VH	A	1. Z478 (4%), 2. D610 (3%), 3. C61 (3%)
13	1043	65.0	0.86%	A	A	A	L	A	A	1. M058 (6%), 2. M161 (5%), 3. M171 (4%)

14	758	66.0	19.39%	VH	VH	VH	H	L	H	1. Z491 (16%), 2. C509 (4%), 3. C539 (3%)
15	1903	22.0	3.99%	L	L	L	A	H	A	1. Z349 (10%), 2. Z358 (10%), 3. Z491 (5%)
16	636	63.0	95.44%	H	H	A	H	A	A	1. E109 (7%), 2. E119 (7%), 3. H360 (4%)
17	863	48.0	5.91%	A	L	L	VL	VH	A	1. Z491 (6%), 2. N10 (4%), 3. N185 (3%)
18	472	48.0	84.53%	VH	VH	A	A	H	A	1. F200 (30%), 2. F900 (4%), 3. Z032 (3%)
19	392	73.0	22.96%	L	L	L	L	VH	H	1. G35 (14%), 2. G20 (6%), 3. G301 (6%)
20	1440	64.0	6.67%	A	L	L	L	VH	A	1. Z491 (4%), 2. R298 (3%), 3. R42 (3%)
21	2311	65.0	12.42%	VH	H	H	L	L	A	1. C509 (14%), 2. C349 (6%), 3. C189 (4%)
22	956	77.0	2.2%	L	A	A	H	A	H	1. Z491 (5%), 2. I702 (4%), 3. I890 (3%)
23	2231	61.0	14.43%	H	A	A	A	A	A	1. Z491 (6%), 2. K509 (3%), 3. R104 (2%)
24	173	76.0	6.94%	A	H	H	H	A	H	1. D509 (4%), 2. D500 (2%), 3. M45 (2%)

25	1350	74.0	97.19%	A	H	H	A	A	H	1. C61 (8%), 2. N40 (4%), 3. C679 (4%)
26	313	49.0	3.83%	VH	H	H	VH	L	A	1. E668 (13%), 2. Z491 (5%), 3. E109 (4%)
27	1077	60.0	95.91%	A	A	A	H	A	H	1. J449 (16%), 2. C349 (3%), 3. J459 (3%)
28	1060	64.0	95.28%	A	L	L	L	H	A	1. M45 (5%), 2. C61 (2%), 3. M161 (2%)
29	3357	45.0	83.8%	VL	VL	VL	VL	L	VL	1. None (44%), 2. B24 (2%), 3. Z113 (1%)
30	8	6.0	25.0%	VL	L	H	VH	A	VL	1. S065 (26%), 2. S062 (14%), 3. Z033 (14%)

Table 6.3.: General overview of all of the 30 clusters

This table is only meant to serve a general overview of how the clusters vary, and it is a challenging task to find the most interesting clusters from this overview. These will rather be analysed in terms of outcome in the next step.

Moving further in the examination of the clustering results, a heatmap for the ICD chapter in each cluster was produced, as can be shown in Figure 6.2. Finally, the heatmap of each of the demographics and numerical features can be shown in Figure 6.3. Note that the two heatmaps need to be interpreted differently. The one for the ICD chapters represent the proportion of patients with the occurrence of the given ICD chapter within each of the clusters. A value of 1 represent that all patients in the cluster have at least one code within the given ICD chapter. The heatmap for the numerical features represent the normalized means within each feature, and the values for the same feature are related to each other. Each column in this heatmap will have minimum one cell of 1 representing the cluster with maximal mean value.

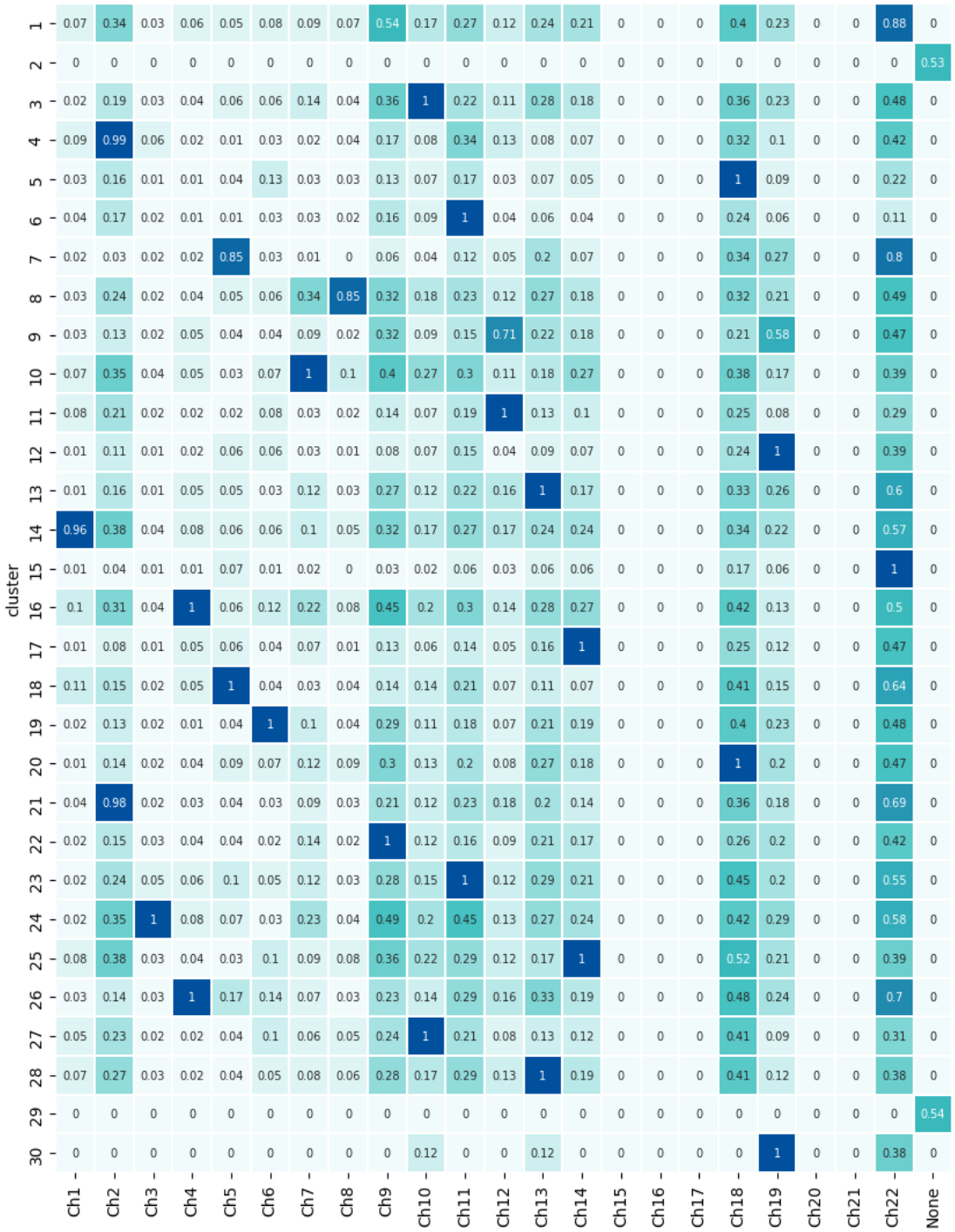


Figure 6.2.: Heatmap for ICD chapters

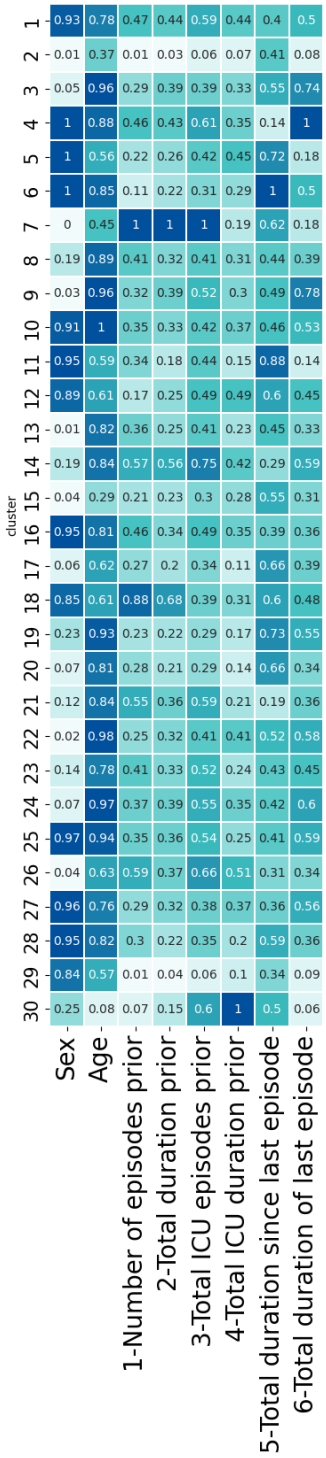


Figure 6.3.: Heatmap for mean values of each feature

6.2.5. Relate Cluster Results to Patient Outcomes

Experimental Setup

In the fourth and last phase of the experimental plan, the aim was to link the patient clustering results with their respective outcome. After the clusters have been defined and characterized, each cluster was examined in terms of the associated outcomes, namely, the result of the blood culture and the total duration of hospital visits within 60 days post-suspected episode. This analysis was conducted in a similar way as before, where the characteristics for each cluster were examined separately, as well as examining the distribution of the different outcome groups. Interesting findings from these clusters will be further analyzed with the results from the last step, to relate the patient history to the different outcomes.

The results of this last phase provide insights into how the clusters differs in terms of their blood culture result and subsequent hospital stay.

Results

The overall outcomes including both blood culture results and post hospital stay were analyzed for each of the 30 clusters. A summary of the outcomes associated with each cluster is presented in Table 6.4. This table shows the average total duration of the subsequent hospital stay, the amount of positive blood cultures, and the distribution of each outcome group.

			Outcome distribution			
Cluster no.	**Average post duration**	**Amount pos. BC**	**NS**	**NL**	**PS**	**PL**
1	16.0d, 1.0h	5.17%	70.1%	24.73%	3.9%	1.27%
2	11.0d, 13.0h	10.63%	64.8%	24.57%	6.26%	4.37%
3	10.0d, 7.0h	3.32%	69.55%	27.13%	2.32%	1.0%
4	25.0d, 2.0h	36.12%	35.65%	28.23%	26.56%	9.57%
5	18.0d, 4.0h	16.89%	53.59%	29.51%	11.07%	5.83%
6	13.0d, 15.0h	20.32%	49.2%	30.48%	16.04%	4.28%
7	6.0d, 22.0h	2.09%	74.35%	23.56%	1.57%	0.52%
8	14.0d, 5.0h	5.57%	70.89%	23.54%	4.87%	0.7%
9	16.0d, 14.0h	15.52%	58.12%	26.35%	10.83%	4.69%
10	17.0d, 18.0h	20.37%	55.58%	24.05%	15.24%	5.12%

11	8.0d, 12.0h	2.99%	74.25%	22.75%	2.4%	0.6%
12	22.0d, 2.0h	3.89%	74.17%	21.94%	3.06%	0.83%
13	14.0d, 5.0h	19.56%	55.8%	24.64%	12.37%	7.19%
14	14.0d, 5.0h	14.51%	57.78%	27.7%	10.03%	4.49%
15	15.0d, 13.0h	11.88%	73.62%	14.5%	7.62%	4.26%
16	14.0d, 4.0h	5.82%	67.61%	26.57%	3.93%	1.89%
17	6.0d, 23.0h	4.75%	69.99%	25.26%	3.01%	1.74%
18	16.0d, 1.0h	6.57%	67.16%	26.27%	4.66%	1.91%
19	11.0d, 0.0h	15.56%	59.44%	25.0%	10.2%	5.36%
20	8.0d, 21.0h	2.92%	68.54%	28.54%	2.01%	0.9%
21	13.0d, 19.0h	5.06%	63.31%	31.63%	3.46%	1.6%
22	18.0d, 6.0h	20.92%	48.74%	30.33%	12.97%	7.95%
23	11.0d, 8.0h	3.45%	67.64%	28.91%	2.38%	1.08%
24	14.0d, 17.0h	4.05%	62.43%	33.53%	2.31%	1.73%
25	14.0d, 10.0h	24.07%	52.22%	23.7%	17.26%	6.81%
26	16.0d, 3.0h	14.06%	64.54%	21.41%	8.63%	5.43%
27	10.0d, 14.0h	3.25%	69.64%	27.11%	2.14%	1.11%
28	11.0d, 5.0h	3.58%	67.26%	29.15%	2.36%	1.23%
29	10.0d, 2.0h	5.36%	70.21%	24.43%	3.57%	1.79%
30	12.0d, 13.0h	12.5%	62.5%	25.0%	0.0%	12.5%

Table 6.4.: Summary of outcome values for each of the 30 clusters

Looking at the overview, there are some clusters that stand out having a significant high or low value for positive blood cultures. Among the clusters with a significant low amount of positives, being less than or equal to 4%, are cluster no. 3, 7, 11, 12, 17, 20, 23, 24, 27, 28. These clusters will hereafter be referred to as the negative clusters. The positive clusters, with amount of positives being greater than or equal to 19.5% include 4, 6, 10, 13, 22, 25. The negative and positive clusters with their values are summarized and compared to the values in the whole dataset in Table 6.5 and Table 6.6, respectively. Details of these clusters will also be presented. In these descriptions, there will be several referrals to ICD chapter numbers. While each mention chapter will be accompanied by

its relevant description, readers are advised to visit Appendix C for a comprehensive mapping between the chapter number, the corresponding code range and the description.

Details of Negative Clusters

Cluster 3: Comprised primarily of older women, this group stands out for its longer duration of last episode. All patients in this group have been diagnosed with a diagnose in chapter 10, *diseases of the respiratory system*.

Cluster 7: Notable for its younger female demographic with very high rates of prior medical history. Yet, these patients have shorter ICU stays, averaging 9,5h per ICU stay compared to the overall average being more than 33h. The patients in the cluster also have shorter durations of last episodes. Diagnoses included are mainly from chapters 5 and 21, corresponding to *mental, behavioral and neurodevelopmental disorders* and *factors influencing health status and contact with health services*, respectively. The cluster has a notable 16% of patients being diagnosed with borderline personality disorder (F603).

Cluster 11: Characterized by young men with an average number of prior episodes but shorter durations, with an average of less than 6h per episode compared to the average being almost 31h. These patients have had a long time since their last episode, and their last episode was brief. All of the patients in the cluster has at least one diagnose in chapter 12, covering *Diseases of the skin and subcutaneous tissue*. The patients typically end with a shorter following hospital stay.

Cluster 12: Mainly men around the age of 48. The cluster is characterized by a low number of prior episodes, but each episode's duration is extended, averaging 48h. This is true for ICU stays as well, where the average time per ICU stay is 50h. All patients have at least one diagnosis within chapter 19, describing *injury, poisoning and certain other consequences of external causes*. This cluster is particularly interesting because of the extended subsequent hospital stay, averaging 22 days, being the second longest of all clusters, and longest among the negatives.

Cluster 20: Mainly women with short prior stays in the GMU and ICU. These patients have a long period since their last episode and typically conclude with a short hospital stay. The patients have at least one diagnose in chapter 18, *symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified*

Cluster 23: The most mixed cluster in terms of sex, but with a female majority. Patients here have a higher number of GMU stays, but shorter duration per stay compared to the overall average. All have a diagnose in chapter 11, *diseases of the digestive system*.

Cluster 27: Mostly average men make up this cluster. They have slightly longer ICU stays, shorter durations since their last episode, and longer durations of the last

	All	Clusters							
		3	**7**	**11**	**12**	**20**	**23**	**27**	**28**
Size	35,694	1,596	191	167	360	1,440	2,231	1,077	1,060
Age	58	75	35	46	48	64	61	60	64
Amount male	52.5%	5.14%	0%	94.61%	88.89%	6.67%	14.43%	95.91%	95.28%
No. episodes	25	22	75	26	13	21	31	22	23
Total duration	32d	39d	102d	18d	26d	21d	34d	33d	23d
No. ICU stays	1.43	1.31	3.36	1.48	1.64	0.98	1.74	1.29	1.18
Total ICU duration	2d 0h	2d 6h	1d 8h	1d1h	3d 10h	0d 23h	1d 16h	2d 13h	1d 8h
Duration since last episode	153d	190d	216d	306d	209d	228d	148d	126d	206d
Duration last episode	1d 15h	2d 21h	0d 17h	0d 13h	1d 18h	1d 7h	1d 18h	2d 4h	1d 9h
Duration post 60 days	14d	10d	7d	9d	22d	9d	11d	11d	11d
Amount pos. BC	8.45%	3.32%	2.09%	2.99%	3.89%	2.92%	3.45%	3.25%	3.58%

Table 6.5.: Characteristics of the negative clusters

episode. Diagnoses are all within chapter 10, which makes this group similar to Cluster 3 except for the sex.

Cluster 28: Men with an average GMU history, fewer and a bit shorter ICU stays, and a long period since the last episode. All patients have a diagnose in chapter 13; *diseases of the musculoskeletal system and connective tissue*

Details of Positive Clusters

Cluster 4: This cluster only consists of men, with an average age of 69 years old. All prior history values are higher than average, the gap since last episode is only 48d old and the duration of the last episode is almost 2,4 times more than the average. The absolute majority of the patients have been diagnosed with a diagnose in chapter 2, *neoplasms* and 14% have a history of C189, indicating colon cancer. This cluster is definitely the group with highest positive rate, exceeding 36%. It is also the group with the longest hospitalization of the 60 days following the suspected episode, with 25 days.

Cluster 6: Another cluster of older men, but with fewer prior episodes. This group is notable for its significant average gap of 347 days since the last episode. All patients have a diagnosis in chapter 11, *diseases of the digestive system,* with as much as 59% diagnosed with some form of cholelithiasis, i.e gallstones. Despite average total duration post-blood culture, the number of positive results is more than double the typical rate.

Cluster 10: Characterized by the oldest cluster of men, the medical history of this group is mainly average. However, all patients share diagnoses within chapter 7, indicating *diseases of the eye and adnexa*, and is the only cluster with this specific characterisation. The patients have a slightly extended hospital stay, and a positive rate of above 20%.

Cluster 13: The first positive cluster with a majority of women, and the women cluster with the highest positive rate, being almost 30%. The patients' history is largely average, with somewhat low total ICU duration. All have diagnoses within chapter 13, i.e *Diseases of the musculoskeletal system and connective tissue.*

Cluster 22: This cluster, with an average age of 77, is the oldest women's group. The patients' history is average, but there are high numbers for features related to the last episode, both the duration since last episode and duration of last episode. The most common ICD chapter is number 9, *diseases of the circulatory system*, where all patients had at least one diagnose from this chapter.

Cluster 25: This last group is yet another a typical male cluster. The history is average, and the group has slightly higher numbers for total duration, number of ICU stays, and last episode duration. The most frequent ICD chapter is number 14, indicating

	All	Clusters					
		4	6	10	13	22	25
Size	35,694	418	374	761	1,043	956	1,350
Age	58	69	66	78	65	77	74
Amount male	52.5%	100%	100%	90.67%	0.86%	2.2%	97.19%
No. episodes	25	34	8	26	27	19	26
Total duration	32d	44d	23d	34d	26d	33d	37d
No. ICU stays	1.43	2.04	1.04	1.4	1.39	1.38	1.82
Total ICU duration	2d 0h	2d 11h	2d 0h	2d 13h	1d 13h	2d 20h	1d 18h
Duration since last episode	153d	48d	347d	159d	156d	179d	142d
Duration last episode	1d 15h	3d 21h	1d 23h	2d 1h	1d 7h	2d 6h	2d 7h
Duration post 60 days	14d	25d	14d	18d	14d	18d	14d
Amount pos. BC	8.45%	36.12%	20.32%	20.37%	29.56%	20.92%	24.07%

Table 6.6.: Characteristics of the positive clusters

diseases of the genitourinary system. Post-blood culture duration is average, while the positive rate is at 24.07%.

6.2.6. Validate Methodology and Results With Clinicians

Experimental Setup

In order to validate the methods utilized and clinical results achieved, the findings were presented at a meeting with the CoSem group. This presentation took place on the 5th of June, with the audience comprising clinical professionals from St. Olavs Hospital who are researching various aspects of sepsis and bloodstream infection. The aim of the presentation was to share the results derived from our study, and initiate a dialogue regarding these findings from a clinical perspective.

The presentation began by presenting the overall research goal, as stated in Section 1.2, followed by a description of the dataset using descriptive statistics and distribution of ICD codes. Subsequently, the specifics of the SASCA was presented, including the chosen similarity measures, the optimal weights and number of clusters used.

After describing the data and laying out the methodology, the results were presented. The focus of this part was to discuss specifics of the clusters categorized as either negative or positive, as detailed in the previous section. Each of theses clusters were presented with their mean age, amount of males, mean number of episodes, total duration, number of ICU stays, total ICU duration, duration since last episode, duration of last episode, duration of stay post 60 days and the amount of positive blood cultures. For the sake of comparison, the same parameters for the entire dataset were included.

In addition to these numerical features, a heatmap to represent the ICD chapters within each of the interesting clusters were also included. After this presentation, time was set aside for discussion of the methods and results.

Results

The clinicians who were present had interesting inputs and points to discuss after the presentation. Their feedback and discussions offered crucial insights, highlighting the importance of earlier clinician involvement in the process. They provided intriguing questions and suggestions that will be expanded upon in Chapter 8.

One point of interest was the decision to analyze only the last suspected episode. Although this was done to maximize the historical data, a suggestion was made to do a separate analysis and consider the first suspected episode, which might yield additional insights. Other points raised included suggestions to base the history on specific, interesting episodes. These interesting episodes include either only the number of suspected episodes, or the number of episodes with an actual confirmed BSI.

Moving to the cluster results, the clinicians confirmed the higher average age of infected patients. However, the male predominance in positive clusters was odd. This highlighted the importance of also including the sizes of each cluster in the presentation, as the women clusters could include more patients in total. When these numbers were found,

and we sat that the finding of positive clusters with more men still existed, the clinicians related this to the suspicion of men having a worse immune system in general.

Regarding the heatmap of the ICD chapters, the clinicians found that these helped visualize the group separation, but the chapters themselves did not provide much information. The specific most frequent diagnose was presented for the clusters where this frequency was significant, which was more interesting for the clinicians. This emphasize the importance of clustering based on diagnoses, as SASCA does, instead of only the chapter as other typically existing algorithm would have done.

The specific cluster that was discussed the most was the one with the largest amount of positive patients, namely cluster 4. The cluster represent old men with an extended and recent prior hospitalization, and a history of cancer. The fact that a history of cancer indicate an increased risk was known for the clinicians, so this could explain the high positive rate. The clinicians discussed the fact that cancer patients are at risk both because of the surgeries done for these patients, and because of their worse immune system, and that critical ill patients could easier get bacteria in the bloodstream.

However, cluster 4 also had some strange and unexpected findings. The first thing pointed out was the amount of males, and that no other positive cluster represent the same findings for women. There are no known increased risks related to male patients with cancer compared to females, so it would be expected to find the same for a female cluster. Another unexpected finding was the high frequency of colon cancer, as this specific diagnose is not known to be more present for men. It would be more expected to find diagnoses within C60-C63 range in this cluster, as they represent malignant neoplasms of male genital organs.

The presence of colon cancer is odd in general, as this type is not associated with any increased risks compared to other cancer diagnosis. The clinicians wondered if it could be related to the surgery of removing the colon, which could be further investigated by looking back on the procedures documented for these patients in the original dataset. It was also discussed if colon cancer might have an increased risk of specific bacteria. Hence, the pathogen found in the blood culture for these specific patients could also be investigated by looking back to the original dataset. This suspicion also made a suggestion of including the specific pathogen in the clustering, in stead of using a binary representation of presence as done in this experiment.

The other clusters were not discussed in the same extent as cluster 4. Of the most interesting other findings discussed was the fact that the women clusters did not include any significant relations to diagnoses, and that the women cluster with highest positive rate actually had pretty average history. Further details into this were not addressed.

The meeting was concluded by that most of the findings need to be further investigated and researched to be made use of. The clusters provide as a good starting point for looking into specific combinations of demographics, historical features and diagnoses. The findings related to cancer and age are known, and the high amount of men could emphasize the suspicion of them having a worse immune system.

7. Evaluation

This chapter aims to evaluate the results presented in the previous section. This includes an evaluation of the data selection in Section 7.1, before evaluating both the preliminary experiments, in Section 7.2, and SASCA in Section 7.3. The section continues with an evaluation of the clinical results in Section 7.4 and concludes with an evaluation of the methodology in Section 7.5.

7.1. Data Selection

This section will evaluate the data selection process, highlighting its strengths and limitations, and considering its impact on the study's findings and interpretations.

The selection appears to be effective and describing, as only one patients was excluded due to the absence of any ordered blood cultures and thus, not meeting the criteria of the study being a patient with suspicion of a BSI. However, some adjustment could have been done when selecting the episode where the blood culture was ordered. Currently, this episode only forms part of the post-history, but it might be more appropriate to split it at the blood culture date. This way, the part of the history before the blood culture would have contributed to the prior history, while the second part contributes the post history. This would allow a more precise representation of how the episode relate to the history.

During the clinical validation conducted with clinicians from the CoSem group, the clinicians discussed the decision to analyze only the last suspected episode. Initially this choice was made to maximize the historical data; however, it was suggested that including analysis of the first suspected episode could provide additional interesting insights.

Looking further on the features within the resulting clusters, the values varies significantly, suggesting a robust descriptive approach. However, the relationships of these features to the specific outcome of BSI remains a bit more unclear.

Despite efforts to handle outliers through normalization, many numerical features demonstrated a dense distribution around the means. This suggests that outliers may not have been effectively addressed and could be a consideration for future data handling approaches.

While the data selection process generally was effective, some modifications could enhance the depth of analysis and accuracy of results in future research. This includes adjusting the selection criteria for the suspected episode, both the cutoff for prior and post history, and using the first rather than the last suspected episode. Additionally, it would have been beneficial with a better outlier handling.

7.2. Preliminary Experiments

The preliminary experiments carried out in the early stages of the study provided valuable insights. Even though the experiments were not fully implemented, they proved to discover potential challenges, limitations and requirements that informed SASCA's development. Significant issues addressed in these initial experiments were applicable to the main experiments, while non-recurrent aspects offered guidance for future research direction.

One key observation from these experiments was the challenge related to implementing less-known algorithms. Such algorithms, in this case derived from research papers, tend to have less documentation and less intuitive code choices compared to well-known algorithms like k-means and k-prototype. For instance, the reasoning for selecting the specific edges in the DDSCA appeared arbitrary, which led to complications during implementation.

While the selected algorithms corresponded well with the complex data structure in this study, adjusting the data to facilitate the implementation of a well-documented, well-known algorithm could have been beneficial. This approach could have yielded preliminary results to serve as a comparative measure for SASCA's results. With the current approach, the absence of such comparative results presents a less robust foundation for SASCA's evaluation.

7.3. SASCA

Evaluating SASCA compared to other methods presents a challenge due to the study's limited time frame and its iterative, experimental nature. However, it is important to remember that the overall aim of the study is not to engineer the most efficient or most accurate clustering algorithm, but rather the concept of applying clustering algorithms to reveal potential relationships between various features of medical history. Hence, the evaluation of SASCA will focus on how an which features change across the different clusters.

As explained in step two of the main experiment, the elbow method suggested $k = 30$ for most of the weights, and the highest CH index was achieved for the values $w_{single} = 0.5, w_{set} = 0.4$ and $w_{outcome} = 0.1$. The highest CH index was however found for $k = 2$, but the large amount of patients make $k = 30$ a more suitable number. The utilization of the combination of these two metrics, both WCSS and the CH index, strengthens the resulted optimized values, as the elbow plot suggested k=30 for the weights and these weights gave the highest CH index when k=30.

With the given data of 35,694 patients, it is expected that clustering with $k = 30$ should yield groups with a decent amount of patients in each. However, it is evident that the clusters vary a lot in size, from n=8 in cluster 30, representing only 0,02% of the patients, and n=8281 representing in cluster 1, representing 23%. This variation in size will influence the CH Index and WCSS metrics, as a more nuanced cluster will yield a lower WCSS.

Despite the varying sizes, the algorithm captured significant characteristics in each cluster. By evaluating how the clusters vary in values for each feature, the separation and cohesion can be evaluated, as well as it will provide a clue for the feature importance. Being the only categorical value with almost even distribution, the sex feature is well differentiated across the clusters with several clusters consisting of only patients with the same sex. Considering the age distribution skewed to older individuals, this feature is also well distributed. The youngest group except for the small cluster no. 30, averaged 22 years, while the oldest averaged 78.

Using the scale defined for very low to very high values for the numerical features, we can see that the algorithms finds more significant values for number of episodes prior and duration since last episodes, with 9 clusters having either very high or very low values for both of the features. The values for total duration differs the least, with only 5 clusters with a very high or very low value. Looking at these features for the clusters with significant high number of positive outcomes (amount positive $\geq 19.5\%$), two of the six clusters had either very high or very low value for duration since last episode, one had very low value for number of episodes, and one had very high value for duration last episode. The negative clusters (amount positive $\leq 4\%$) had a very high number of value at least once for each of the features, while the duration since last episode was again the features with the most significant values, where four of the clusters had a very high value. This indicate that the algorithm quite well separate patients with different histories.

Regarding the ICD codes, there are clear clusters with a significant amount of the different chapters. Each of the chapters except 21 have a frequency of more than 65% in maximum two of the clusters, and each chapter except 15, 16, 17, 20 and 22 have at least one cluster with 85% or more in frequency. This indicates a both a good separation and a good cohesion.

Moving over to the last features, representing the outcome, the separation and cohesion is not that evident. When only looking at the distribution of the outcome groups in each cluster, it is clear that the largest portion of patients are categorized as NS in all clusters. With the skewed distribution of outcomes in the selected dataset, this is expected. It is easier to see the difference when looking at the two features separately, where the post duration range from 6d and 22h to as much as 25d and 2h, and the amount of positive range from 2.09% to 36.12%. Considering the overall average being 14d post stay and 8.45% positive rate, the range of the values indicate a somewhat good separation and cohesion after all.

7.4. Clinical Results

The last part of the evaluation include the clinical results, where the findings will be evaluated to both the related work and expert clinical knowledge through a discussion with clinicians as a part of the CoSem group. The evaluation is again based on presentation and discussion conducted 6th of June, where the results related to the clusters categorized as negative or positive were presented to clinicians in the research group.

The clear differences in sex for each cluster is quite interesting, where the cluster with

the biggest mix consist of 14.43% men. This separation is even clearer in the positive clusters, where less than 10% women was included in the most mixed cluster. It is important to remember that this variable is categorical, as the only categorical value in the single values, which contributes a clearer difference when using euclidean distance.

Looking further in to the positive clusters, one obvious finding was the higher average age in each cluster, in addition to including cluster 10 with the highest average age of all clusters. Another interesting finding was the sex ratio, where four out of six clusters were characterized by mainly being male. Also the number of patients within these cluster are higher than the number of female patients, being 2903 out of 4902.

The key finding in the cluster was that older male patients with an extended prior hospital stay and cancer diagnosis carry an increased risk of infection. This finding aligns with both existing research and clinical knowledge. The overall finding of all positive clusters showing a higher mean age further confirms known risk factors, and is consistent with existing knowledge. These observations strengthen the credibility of the results and confirms the methodological approach used in the study.

However, the other nuanced and more specific findings in the clusters do not align particularly with the current understanding, and should be further investigated. The male clusters with either colon cancer or cholelithiasis as the most frequent diagnose were specifically odd, as the diagnoses are not more frequent in men than in women. Since one specific diagnose rarely suggest an increased risk for only one sex, it would have been expected to find this in a mixed cluster with respect to sex, or in a cluster for each of the sexes, if the diagnose indicates an increased risk. This suggest that the number of clusters are too low, not capturing the females with the same diagnose, or that the finding is random.

As the clustering method was able to find the already known risk factors in some clusters, the feasibility of the method is somewhat confirmed. However, some of the key findings within the clusters are not supported by existing knowledge, which suggest some random clustering. These findings should be further investigated to evaluate the feasibility of the clustering.

7.5. Methodology

This section evaluates the implementation of the Design Science Research (DSR) methodology in this study. Despite the iterative and exploratory nature of the study, which made DSR a fitting choice, the tight timeline and extensive scope presented challenges. Some preliminary literature review was conducted during the previous semester, however the largest part of the research that proved to be relevant was done during this spring semester. This main time frame of one semester constrained the number of completed experimental cycles. The adaptation of DSR for this study allowed for experimentation without requiring the completion of each cycle, which was particularly appropriate given the limitations.

The work conducted in the fall semester occurred before accessing the data, resulting in much of this early work becoming irrelevant once the scope was refined upon data

access. This underlines the need for early data access, allowing for better familiarity with the data and a clearer understanding of the research possibilities.

With the addressed time constraints, the attention allocated to each step of the research approach — data familiarization, domain knowledge acquisition, literature review, and method implementation and evaluation — was limited. The implementation phase often felt wasted as considerable time was dedicated to experiments later deemed unsuitable for the study's objectives. Despite these challenges, the trials were at the heart of the DSR approach, providing valuable lessons for the main experiment.

In conclusion, regardless of moments being overwhelmed, the adapted DSR approach was suited for the research. Lessons learned from each design cycle was valuable, contributing significantly to the final outcome.

8. Discussion

The results from Chapter 6 will now be discussed in light of the research questions. This chapter will discuss each of the research questions in Section 8.1, Section 8.2 and Section 8.3, before concluding with the limitations in Section 8.4.

8.1. Research Question 1: Relevant Features

RQ1: What are the relevant features to be used to describe a patient's medical history in the context of clustering?

When implementing well-known clustering algorithms from established libraries, the evaluation of feature importance is often included and a straightforward task. This is however not the case when implementing an own made algorithm. The following discussion of the relevant features will hence focus on how the features distinguish between the clusters, as well as including aspects addressed by the clinicians in the validation done during the presentation 5th of June. The relevance to the patient's outcome will not be addressed here, but rather discussed in a later section.

The number of episodes prior feature seems to be highly correlated to the total duration prior, as each significant value of this feature is also always present for the number of episodes, and the distance between the scaled value of the two features is at maximum one. This indicates that there might not be necessary with both the number of episodes prior and the total duration. The total duration is also the feature with the least number of significant findings, indicating that the number of episodes should be sufficient and the only feature describing the prior history of GMU stays.

Looking at the total duration since last episodes, there is no such obvious relation to the other values. However, it could be observed a possible inverse relationship between the duration since the last episode and the duration of the last episode. For every instance where the duration since the last episode was categorized as very high, the duration of the corresponding last episode was almost always average or lower. In contrast, whenever the duration since the last episode was very low, the duration of the last episode tended to be average or above. This is not as consistent as the prior correlation, with for instance exceptions in cluster 3, 19 and 29. With this exceptions present, one should be careful to assume that this relation is applicable, and it is expected that the features provide different aspects to the history.

The values for the ICU stays seem correlated, but not to the same extent as the GMU stays. For instance cluster 7 shows that there could be significant difference in the number of ICU stays, here being very high, and the total ICU duration, being low. The features, especially the total number of ICU stays, seems to some extent relate to the

GMU features. Cluster 30 is the only cluster where the difference in the scaled value of number of GMU stays and ICU stays differ significantly. Because of the small amount of patient in cluster 30, being only 8, this exception is not representative for the entire dataset. Hence, it is naturally to believe that the feature describing the number of ICU stays is redundant together with the number of episodes in general.

The results from the validation with clinicians also provide valuable insights to the chosen features. It was suggested to only count the episodes where there were a suspicion of BSIs, or only the episodes with a confirmed infection. This is interesting for this particularly case where the goal of the clustering was to investigate these kind of patients, but is not necessarily interesting in the general case of describing medical history. This might indicate that the research question is too general for this specific case, as further discussed in Section 8.4.

During the same validation, the clinicians emphasized the clinical meaning of using each particular ICD code and not only the ICD chapter. This indicate that the list of ICD codes also should be included as a feature describing the history.

8.2. Research Question 2: Application of Clustering to Differentiate Patients

RQ2: How can the application of clustering help differentiate patients with varying outcomes in suspected bloodstream infection cases, considering their medical history?

Initially, this discussion will cover aspects related to clustering medical data in general, focusing on the results from the preliminary experiments. Following this, the specifics and decisions related to the main experiment, SASCA, will be discussed. Finally, the research question will be correlated to the specific results from the clustering of patients suspected of having BSIs. In this discussion, only the results with significant findings for either negative or positive outcomes, as presented in table Table 6.5 and Table 6.6, will be addressed.

8.2.1. Application of Clustering Algorithms on Medical Data

A key factor when clustering medical history is the structure of patient histories. As each patient's history is unique in both content and length, with differing numbers of episodes and diagnoses, finding a standard representation for all patients is difficult. A specific set of features need to be chosen for such a description. Even though the specific features chosen in this study are discussed in the previous chapter, the challenges related to this, in the context of clustering medical data, will be addressed here.

The large volume of information for each hospital visit can be overwhelming, making it crucial to consider the curse of dimensionality when choosing the features to use (Bellman, 1966). This curse emerges when an excess of features are chosen, resulting in the algorithm struggling to find similar patients. Thus, summarizing each patient's history in a descriptive and compact way is essential.

However, knowing what to describe in each specific context can be challenging, and it is necessary to make assumptions even before the experiments begin. For instance, with clinicians expressing a wish to count the number of suspected episodes retrospectively, it is assumed that this number is related to the outcome of the suspected episode. This holds true for all features chosen in this research as well, where the outcome is assumed to be associated with the number and duration of stays, as well as details related to recent visits.

Taking into account the diagnosis codes given to each patient, the dimensionality increases even more. With over 1400 different ICD codes in our dataset, it is clear that each code cannot be represented as a feature. Zhong et al. discuss how existing clustering algorithms struggle to cluster patient data where both single and set values are present (Zhong et al., 2020, 2021). There is no distance measure that capture both parts of the features, introducing the need of finding another approach.

One alternative could have been to convert each of the ICD codes into corresponding ICD chapters, using a binary representation and k-prototype for clustering. However, with each ICD chapter consisting of up to 204 different codes, much valuable information would be lost with this approach. It was emphasized during the results presentation to clinicians on June 5th that ICD chapters themselves are not clinically interesting. This reinforces that such an approach would not be applicable, and underscores the need for a custom or specifically designed clustering algorithm for this kind of medical data clustering.

Keeping all the diagnosis codes for each patient presents another challenge when measuring the similarity between patients. In the DDSCA algorithm implemented in the preliminary experiment, this similarity was measured through a weighted edit distance to ensure the order was considered. However, with the number of different diagnosis codes for each patient in our dataset, the complexity became far too high compared to the dataset used in the original paper (Zhong et al., 2021). When choosing the distance measure, a balance must be struck between functionality and complexity.

After choosing the descriptive features, the resulting data typically consists of a mix of categorical and numerical data, which poses a challenge when calculating the distance between two patients. This is particularly evident when looking at the sex distribution in the results of this research, which is far more distinct than other features. This specific case will be further discussed in the succeeding sections.

To categorize all variables, the algorithms used in the preliminary experiments employ discretization. However, this method omits the valuable information an outlier can provide. In the context of clustering medical data, these outliers might be just as important. Extreme cases of either very short history or very long history could provide specific indicators of either outcome, and should be considered.

8.2.2. Implementation and Choices in SASCA

In moving to the specific decisions made during the implementation of SASCA, it is crucial to discuss the foundations of the clustering algorithm, namely the distance metrics.

The resulting clusters rely heavily on the measurement of similarity based on these metrics, and their suitability is hence critical.

Starting with single-value features, the SASCA implementation uses Euclidean distance. Although Euclidean distance is generally a good choice for numerical data, its application becomes questionable with categorical features present such as sex. The constant distance of 1 between different sexes and 0 for the same sex will influence the overall distance measure, as evidenced in the cluster analysis where patients of each sex are highly distinguishable.

Addressing the issue of normalization and outliers, very high or very low values in the single features result in most normalized values falling within a narrow range. While this allows the identification of outliers, it may hide more subtle variations within the dataset. The choice of normalization was made based on the dissatisfaction with discretizing done in the preliminary experiments. In retrospect, it could have been interesting to test also this approach with SASCA to examine its impact on the clustering result.

The challenge of defining a meaningful similarity measure for patients' diagnostic histories, expressed in ICD codes, was another significant aspect of this study. The chosen approach leveraged the hierarchical nature of ICD codes to calculate pairwise similarities, by identifying the code in the first list with the minimum distance to each code in the second list. The process of developing and refining this measure was a substantial and time-consuming part of the work. After completing the measure, it was realised that this measure should have been subject of early discussions with clinicians to ensure its clinical validity and relevance.

Towards the end of this study, a potentially promising alternative method for measuring similarity between ICD codes was identified, which involved constructing a bipartite graph of two ICD lists (Gottlieb et al., 2013). In this method, the edges represent the similarity between each code, and the total distance is defined by the maximal matching. Unfortunately, due to time constraints, this approach could not be implemented within the scope of this study. However, its potential to offer a more nuanced understanding of the similarity between patients' diagnostic histories requires further exploration in future research.

The clusters in SASCA are formed based on the weighted combination of the mentioned measures. With the weights, one can optimize and change the relative importance of the different features. In these cases it is important that each similarity measure return the same distribution of values, which is ensured here where each distance return a value between 0 and 1. This somewhat ensures an equal contribution, even though the nature of the measures are different.

With the chosen optimized weights, the set values were not as much weighted as the single values. This could be due to the similarity measure of set values, and that the chosen approach made many pairwise lists of codes similar in distance, making the distance not as significant as for the single values. Even though this finding shows that each patient often has a unique history of codes that is challenging to compare, it also emphasize the need for a suitable distance measure to capture the relevant similarity.

8.2.3. Differentiating Patients With SASCA

The last part of the discussion for research question 2 covers the specific resulting clusters achieved when using SASCA. As the research question aims to distinguish patients with different outcomes, it is interesting to start by looking at the distribution of the four groups defined in Chapter 5 in each of the clusters. The cluster with the largest proportion of the NS group is cluster 7, which is defined as a significant negative cluster. For the NL group, the highest amount is in cluster 24, also being among the ones categorized as negative. Both the PS and the PL groups are most present in cluster 4, being the one with by far highest amount of positive blood cultures. Considering the relative high amount of clusters, it is unexpected that both PS and PL have the highest amount in the same cluster. This could mean two things; either that the algorithm struggle to distinguish these groups, or that there are no clear difference in the different groups regarding their histories. Either of the two indicate that the results in this research is more interesting to examine with the attributes separate rather than the groups defined.

The further discussion will hence focus on the clusters with significant findings for either negative or positive outcomes, as presented in Table 6.5 and Table 6.6. The two groups of clusters will be referred to as the negative clusters and the positive clusters, respectively, and a cluster belonging to the first group will be denoted as a negative cluster, while a cluster belonging to the second group will be denoted as a positive cluster.

At the first glance, there is a clear difference across the different groups in the mean age. All of the positive clusters have a higher mean age than in the entire dataset, indicating that elder patients have a higher risk of infection. This finding is also confirmed by both the literature and the clinicians.

However, it is important to note the high mean age of cluster 3, categorized as negative. Looking into the specifics of the historical features, this cluster seems very similar to the positive cluster 22. Even though the values in each of the clusters differ to a certain extent, it is not significant enough to differentiate two patients. Adding the ICD codes in the analysis make a somewhat clearer difference, but it is not enough to make a conclusion. All patients in cluster 3 have been diagnosed with a code in chapter 10, while all patients in cluster 22 has a diagnose in chapter 9. It is however important to note that 36% of the patients in cluster 3 also has a diagnose within the same chapter as cluster 22, making the difference small. This indicate that there is not a significant finding related to the two clusters, and one should be careful making assumptions related to the risk of these patients.

All of the remaining negative clusters have a lower age than all of the positive clusters. Remembering that the data selection chose the last suspected episode make this finding not surprising, and it would have been interesting to see the differences if the selection rather chose the first episode. The further discussion will focus on the other features contributing to each cluster, but it is important to keep the age difference in mind. The discussion will be based on the positive clusters, and see how they differentiate from the negative.

Examining the features of the positive cluster 4, there is no clear similar negative cluster. This cluster is significant in many ways, being the positive cluster with the most

prior hospitalization, shortest gap since last episode and the longest post duration with highest positive rate. Looking at the ICD codes, cluster 21 is quite similar with the presence of chapter 2 indicating cancer. Regarding the sex rate it is naturally to believe that cluster 21 is the women with the same kind of history as the men in cluster 4. This similar cluster has a positive rate of only 5.06%, which alone could suggest that women with cancer has a lower risk of being infected with BSI than men with cancer. Even though Pittet et al. suggest that male patients with cancer have an increased risk, the existing knowledge does not include such obvious differences between the sexes (Pittet et al., 1997).

However, looking further into the heatmap, there are quite a few clusters with an increased amount of patients diagnosed with a code in chapter 2. This make it natural to suspect that the other women with a cancer diagnose and positive blood culture are distributed into different groups, which indicates that the clustering is not fulfilling the goal of the research.

Moving to cluster number 6, one could in one way relate it to cluster number 12. These are the two clusters with the lowest number of prior hospitalization for each of the two cluster groups. Including the ICD codes however, make the difference clearer. Codes from chapter 11, being the 100% in cluster 6, is only present for 15% of the patients in cluster 12. All patients in cluster 12 are diagnosed with a code in chapter 19, which is only present in 6% of the patients in cluster 6. This indicate a clear difference between the two groups, making the algorithm suitable for differentiating some groups.

The positive cluster 10 is quite similar regarding the historical features to the negative cluster 27, and the other positive cluster 22. The last mentioned has a lower number of prior episodes, but the other features are quite similar. All of them have average values for the other features except for total ICU duration and total duration of last episode, being high in all three. 100% of the patients in cluster 10 have a diagnose in chapter 7, 40% in chapter 9 and 27% in chapter 10. These values for cluster 22 are 14%, 100% and 12% respectively, while being 6%, 5% and 100% for cluster 27. The negative cluster stand out with the small amount of patients with a diagnose from chapter 7 and 9, which indicate a separation between the positive and negative clusters.

So far in the discussion, the ability for the algorithm to differentiate the groups have been proved to vary. This is also true for the two remaining positive clusters. There is no other negative cluster to be categorized as similar to cluster 13, indicating a good differentiating, while cluster 25 could again be related to the negative cluster 3. Examining the heatmap of the ICD codes for the two cluster 25 and 3, the most significant chapter is different across the two. However, both clusters have quite a high rate of several present chapters. Hence, the differentiating might not be as good for these clusters.

By examining these positive clusters and their similar patients group, the ability of SASCA to differentiate patient with different groups have been tested. Some of the positive patient groups are clearly different than all of the negative groups, including cluster 4 and 13 with no significant similar history as the other negatives. The clusters 6, 10, 22 and 25 on the other hand, could be related to some of the negative clusters regarding their numerical features. Most of them are different regarding the ICD diagnoses, but

it is important to remember that the ICD chapters themselves do not provide enough information. It would be interesting to spend more time examining the actual history of the patients in each of the groups by investigating the original dataset, but this would take a significant amount of time. Due to the limited time frame in this project, this is not achievable.

8.3. Research Question 3: Clinical Utility and Potential

RQ3: What is the clinical utility and potential of clustering methods in revealing relationships between relevant features of a patient's medical history and patient outcomes in suspected bloodstream infection cases?

In addressing this specific question, the potential relationships revealed in the specific case will firstly be explored, before proceeding to discuss the clinical utility of such features and findings.

The most significant finding identified was the cluster with 36% positive rate. This cluster was largely populated by older men who had a history of at least one cancer diagnosis, with colon cancer being the most prevalent. Other features characterizing this cluster include a higher rate of prior history, an unusual long duration of the last episode, and a very short interval since the last episode.

Another notable correlation is the prevalence of diagnoses related to gallstones in cluster 6, where 59% of patients had one variant of this diagnosis. This cluster primarily consisted of men with a low count of prior history and a considerable gap since their last episode. Approximately 20% of these patients ended up with a positive blood culture, more than double the average rate.

Most patients diagnosed within chapter 7 were grouped into cluster 10, exhibiting a positive rate of 20.37%, again more than double the average. Mark that this means that almost all patients with a diagnose from this chapter is in this particular cluster, and not separated across two or more different. A prior diagnosis within this chapter could, therefore, potentially indicate a higher risk.

Interestingly, no significant findings related to women emerged, as the histories for these clusters were average and the notable ICD chapters were common across other groups as well. This might suggest that women have a lower risk of infection.

During the analysis, it is important to remember that there will always be patients with the same diagnosis who have been placed in a different cluster. Even if the number of patients with diagnoses within a specific chapter is not significant, there will be exceptions. This applies to other features as well, and thus, clustering should not be viewed as conclusive.

Hence, despite the promising findings, these results should not be considered standalone risk factors in a clinical context. The clinical validation process highlighted several findings considered odd. However, these resultant clusters can be used as a basis for further analysis. With access to the original data, patients grouped in each cluster can be further investigated. For instance, while one cluster may include a significant number of a specific diagnosis and an overall increased positive rate, it does not necessarily imply that the

positive patients are those with that specific diagnosis. Consequently, findings like these can serve as starting points for further analysis and must be validated by examining the actual patients in the original dataset.

Even with the further analysis, it is important to remember that the usage of such findings should be as a guidance and a support, and not as a conclusion for treatment or replacing clinicians reasoning. Patients associated with a low risk should not be excluded from consideration for a blood culture. Instead, those associated with an increased risk could be subject to extra precautions, especially concerning catheter insertions, as discussed in Section 2.1.

The utility of clustering can be further enhanced when clinicians are involved in designing the selection criteria. During the results presentation to the CoSem group, it was suggested that the history of episodes with either a confirmed or suspected BSI should be considered, and rather consider the first and not last suspected episode for each patient. To maximize the utility of the results, both the implementation and validation should be conducted in close collaboration with clinicians.

A significant factor making these clustering methods suitable in a clinical context is their transparency. As detailed in Section 2.2, providing explicit reasoning for the results generated by clustering increases their explainability, unlike models that predict a label without providing further reasoning.

8.4. Limitations

Given the complexity of the data and the task, coupled with the limited time frame during which most of the work was conducted primarily within one semester, there are several notable limitations that need to be addressed. This section aims to elucidate these limitations.

The limitations related to the data in general, including its complexity and reliability, have been previously elaborated in Chapter 5 and will not be repeated in detail in this section. The complexity of the data necessitated a considerable amount of time for familiarization, consequently constraining the time available for the implementation and for conducting experiments. Additionally, the major parts of the research was conducted during one semester, since the work done during the preparatory project was later deemed to be irrelevant. This limit the feasibility of such a complex task and the iterative way of conducting a research. As other algorithms were not fully implemented and tested, the results from SASCA could not be compared.

One of the major limitations was the lack of consultation with clinicians during the decision-making process, which later proved to have had a profound impact on data selection. This underlines the crucial importance of interdisciplinary collaboration.

Regarding the overall research goal and questions, there are also considerable limitations. The first research question, which broadly addresses a general description of a patient's medical history, may be too expansive for the scope of this study. A more specific focus on features directly related to the context to be researched, BSIs in this specific case, might have been beneficial. Additionally, the exploratory nature of the research does not

necessarily provide concrete answers, but rather raises further questions. The potential relationships found could be due to random connections, providing answers to questions we did not have. A more hypothesis-driven research approach could have potentially led to more definitive conclusions.

One last limitation to address is also related to the exploratory nature of the research, as well as the limited time. Even though the study provide several interesting aspects, neither the introduced novel algorithm nor the produced clusters have been fully evaluated. In order to make use of any of these, they need to be further investigated, as elaborated upon in Section 9.3.

9. Conclusion and Future Work

This chapter aims to conclude the work done in this research. Section 9.1 summarizes the overall goal, research method and results, before concluding how the findings answer the research questions and goal. These findings provide several contributions in the field, which will be addressed in Section 9.2. The last section, Section 9.3 will further propose future work that builds upon the findings from this project.

9.1. Conclusion

The main aim of this study has been to explore the application of clustering algorithms for grouping patients suspected of having a bloodstream infection, as conveyed in the research goal defined in Section 1.2. The other part of the goal includes finding how and which features of the medical history relate to patient outcomes. The research seeks to answer the goal by applying a Design Science Research approach for an iterative and exploratory nature.

The selection of features to describe the history was mainly based on results from a literature review, conducted as an adopted approach of the approach proposed by Kofod-Petersen (Kofod-Petersen, 2012). The features cover values for prior history at the GMU, including number of prior hospitalizations and duration of these episodes, as well as the number of stays and duration related to the GMU stays. They also include values related to the most recent stay, both the duration of the last episode and the duration since this episode happened. Lastly, the history is further described by the list of primary ICD codes given during each episode.

Further in this research, these features were used, together with a feature describing the outcome, as input when exploring different clustering algorithms. The exploration started with yet another literature review, ending with two relevant methods applied in the preliminary experiments, namely MASPC and DDSCA (Zhong et al., 2020, 2021). During the implementation of both algorithms, limitations and challenges were faced. These insights guided the development of a novel approach for clustering medical data, SASCA. This algorithm was made as an adoption of DDSCA, with an alternative approach for calculating the distances between each pair of patients.

The optimized implementation of SASCA resulted in 30 different clusters, 8 of which where categorized as negative and 6 as positive. These clusters revealed some relationships already known by the clinicians in the CoSem group, and some findings that were considered significantly odd. This emphasize the need for additionally analysis and the fact that clustering medical data can not be used as a tool alone, but as a starting point for investigation of the patients considered similar.

The findings in the research answer all of the associated questions, however to varying extents. Regarding the first question, it is concluded that features describing the medical history of a patient to be used in a clustering context largely depend on the goal of the clustering. For the specific context in this research it was suggested to only count episodes with suspected or confirmed BSI, which may not apply to another context. However the features that ended up being utilized in this research serve as good basis. Looking at the resulting cluster, it can be concluded that the number of episodes at the GMU and the total ICU duration should be sufficient to describe the overall history, and the attributes for the more recent stays should consist to describe the more recent condition. The list of prior primary ICD codes contribute with the actual diagnosis and should be included in its entirety. The clinical evaluation shows that only the ICD chapters would not be sufficient, as they do not provide valuable information in a clinical context.

When it comes to the second research question, this study shows that the application of clustering methods, particularly the novel approach SASCA, can be used to differentiate patients. Looking at both the demographics, numerical features and ICD codes we can see clear differences in each group's history. However, with the current weights utilized in the optimal clustering with SASCA, the outcome in the terms of the four groups defined in Chapter 5 are not particularly differentiated in the clusters. On the other hand there are clear differences in both the number of patients with a confirmed BSI and the total duration during the 60 days following the suspicion, indicating that clustering can be applied to differentiate patients with varying outcomes.

The last research question aims to investigate the clinical utility and potential of the resulting clusters. This study prove that the clustering in a clinical context has potential, but should mainly be used as a starting point for further analysis. The clustering with SASCA formed groups that revealed both expected and unexpected results. Especially the findings that do not confirm existing knowledge should be further investigated, as the findings either could be revolutionary in a clinical context or only happened by chance. To maximize the clinical utility of clustering methods in a context like this it is crucial with a well-defined interdisciplinary collaboration.

9.2. Contributions

The contributions of this research involve several aspects of the utilization of machine learning, specifically clustering algorithms, within healthcare. First and foremost, this research introduces a novel approach to cluster medical data. With inspiration from the existing DDSCA approach, SASCA cluster medical data with both single and set values in an efficient way. This new algorithm suggests that tailored approaches could be beneficial in the field of patient data clustering. Additionally, the research further evaluates the potential of the two algorithms MASPC and DDSCA on a complex dataset with different objectives than the ones the algorithms were designed for.

Another significant area of exploration is the evaluation of the clinical utility of clustering. The findings suggests that clustering serves well as an initial step for further analysis. The results underscores the need for a well-defined interdisciplinary collaboration

to maximize the potential benefits of these techniques.

Through an in-depth literature review, subsequent feature analysis, and discussion with clinicians, this research also identifies a set of features that are effective in describing medical data for the purpose of patient clustering. In particular, it provides features to be used in the context of analyzing BSIs. Highlighting the importance of using ICD codes rather than just the chapters for clinical analysis, this study emphasizes the role of precise ICD coding. The motivation for correct coding should extend beyond billing purposes to include research considerations.

9.3. Future Work

As this research is one of the first exploring the application of clustering methods on clinical health data in Norway, several potentials for future work have been discovered. This section aims to present some of these, and includes exploring potential in the features describing the medical history, the potential of SASCA and other adjustments that can contribute further in the field of clustering medical data.

9.3.1. Explore and Compare Other Algorithms

Due to the limited time for this research, only three potential algorithms were explored, where two of them only ended up being partly implemented and hence did not produce any results. This make the basis for comparing the results from SASCA weak, including both the computational and clinical aspects of the results. It would have been interesting to see how SASCA performed compared to DDSCA with a portion of the dataset, as the complexity of DDSCA was too high when using the complete data. Another approach could have been to add BSI as a diagnose for each patient with a positive blood culture, and use MASPC to cluster these patients and find related diagnoses.

The algorithms explored in this research was however limited by the literature review conducted, and could be influenced by the subjective judgements of papers concluded as relevant. There could be several other algorithms out there worth trying, which further could serve as a comparison basis for SASCA.

9.3.2. Evaluate SASCA and Feature Importance

To further evaluate the performance and reliability of SASCA, the algorithm should be executed iteratively. These iterations should include both the same dataset with the same features, the same dataset with varying features and varying dataset. This approach will not only contribute to the evaluation of the algorithm in general, but also further investigate the importance of the different features.

9.3.3. Explore Different Distance Measures

By using SASCA as a basis one could do several interesting changes to see how it affect the results. One change that could be particularly interesting is the way the pairwise

distance between the lists of ICD codes are computed. As mentioned in the discussion of research question 2 in Section 8.2, one potential method discovered was the one introduced by (Gottlieb et al., 2013). This involves using a bipartite graph for the two lists, where the edges represent the similarity between each code and the total distance is defined by the maximal matching. Other methods for distance should also be explored and consulted with clinicians.

9.3.4. Explore the Potential of the Dataset

The complexity of the dataset covering the demographics and visits of 35,694 patients introduce many potentials for alternative feature selection. One first potential change, introduced by the clinicians in the CoSem group, is to rather count the history prior to the first episode and not the last. Another suggestion that appeared during the same meeting was to only count the suspected or confirmed episodes. As these changes could be interesting for this particular context, several other attributes could also be explored for a general context. For a more general approach, the column *hastegradkode* (English: urgency code) in the table ***nimesaktivitet*** could be utilized. Additionally, the procedure codes could provide valuable information not captured by the diagnosis codes, and can be handled in the same way as the diagnosis codes in the developed algorithm. These codes can be found in *ncsp* and *ncmp* columns, still as a part of ***nimesaktivitet***. Lastly, one could use more of the test results from ***nsml*** and ***trfl*** to describe the prior history.

9.3.5. Explore the Revealed Relationships

A last suggestion for further research include investigating the relationships revealed in the clustering. By diving into the original dataset, preferably utilizing a database tool like DBeaver (DBeaver Corp and contributors, 2023), one could look at the specific details of the patients grouped together. Does the provided data actually include more men patients with a diagnose of cancer ending with a positive blood culture, or is the women patients with cancer just distributed in the different cluster? The same question can be applied to each of the significant findings elaborated on in Section 8.3. By investigating these further, one could either confirm or reject the potential findings in the resulting clustering.

Bibliography

Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

Yae Jee Baek, Young Ah Kim, Dokyun Kim, Jong Hee Shin, Young Uh, Kyeong Seob Shin, Jeong Hwan Shin, Seok Hoon Jeong, Geun Woo Lee, Eun Ji Lee, et al. Risk factors for extended-spectrum-β-lactamase-producing escherichia coli in community-onset bloodstream infection: impact on long-term care hospitals in korea. *Ann Lab Med*, 41(5):455–462, 2021.

Ziv Bar-Joseph, David K Gifford, and Tommi S Jaakkola. Fast optimal leaf ordering for hierarchical clustering. *Bioinformatics*, 17(suppl_1):S22–S29, 2001.

David W Bates, Lee Goldman, and Thomas H Lee. Contaminant blood cultures and resource utilization: the true consequences of false-positive results. *Jama*, 265(3): 365–369, 1991.

Richard Bellman. Dynamic programming. *Science*, 153(3731):34–37, 1966.

Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

Centers for Disease Control and Prevention. Bloodstream infection event (central line-associated bloodstream infection and non-central line-associated bloodstream infection). *Device-associated Module BSI*, pages 1–48, 2022.

Deborah Cook, Adrienne Randolph, Phillip Kernerman, Cynthia Cupido, Derek King, Clara Soukup, and Christian Brun-Buisson. Central venous catheter replacement strategies: a systematic review of the literature. *Critical care medicine*, 25(8):1417–1424, 1997.

CoSem. Cosem seminar. 2023.

DBeaver Corp and contributors. Dbeaver: Free universal database tool, 2023. URL https://dbeaver.io. Version 23.0.1 [Software].

Direktoratet for e-helse medisinske kodeverk. Finnkode - direktoratet for e-helse medisinske kodeverk. https://finnkode.ehelse.no/#icd10/0/1/0/-1. (Accessed on 05/23/2023).

Dayana Fram, Meiry Fernanda Pinto Okuno, Mônica Taminato, Vinicius Ponzio, Silvia Regina Manfredi, Cibele Grothe, Angélica Belasco, Ricardo Sesso, and Dulce Barbosa. Risk factors for bloodstream infection in patients at a brazilian hemodialysis center: a case–control study. *BMC infectious diseases*, 15(1):1–9, 2015.

Jose Garnacho-Montero, Teresa Aldabó-Pallás, Mercedes Palomar-Martínez, Jordi Vallés, Benito Almirante, Rafael Garcés, Fabrio Grill, Miquel Pujol, Cristina Arenas-Giménez, Eduard Mesalles, et al. Risk factors and prognosis of catheter-related bloodstream infection in critically ill patients: a multicenter study. *Intensive care medicine*, 34(12): 2185–2193, 2008.

Teofilo F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985. ISSN 0304-3975. doi:https://doi.org/10.1016/0304-3975(85)90224-5. URL https://www.sciencedirect.com/science/article/pii/0304397585902245.

M Goto and MN Al-Hasan. Overall burden of bloodstream infection and nosocomial bloodstream infection in north america and europe. *Clinical Microbiology and Infection*, 19(6):501–509, 2013.

Assaf Gottlieb, Gideon Y Stein, Eytan Ruppin, Russ B Altman, and Roded Sharan. A method for inferring medical diagnoses from patient similarities. *BMC medicine*, 11 (1):1–10, 2013.

Gösta Grahne and Jianfei Zhu. High performance mining of maximal frequent itemsets. In *6th International workshop on high performance data mining*, volume 16, page 34, 2003.

Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

Alan R Hevner. A three cycle view of design science research. *Scandinavian journal of information systems*, 19(2):4, 2007.

Thomas L Higgins, Daniel Teres, Wayne Copes, Brian Nathanson, Maureen Stark, and Andrew Kramer. Updated mortality probability model (mpm-iii). *Chest*, 128(4):348S, 2005.

HUNT Cloud. Hunt cloud - ntnu. https://www.ntnu.edu/mh/huntcloud. (Accessed on 05/14/2023).

William R Jarvis. Selected aspects of the socioeconomic impact of nosocomial infections: morbidity, mortality, cost, and prevention. *Infection Control & Hospital Epidemiology*, 17(8):552–557, 1996.

Anders Kofod-Petersen. How to do a structured literature review in computer science. *Ver. 0.1*, 1, 2012.

Kevin B Laupland and Deirdre L Church. Population-based epidemiology and microbiology of community-onset bloodstream infections. *Clinical microbiology reviews*, 27(4): 647–664, 2014.

Kevin B Laupland, DB Gregson, DL Church, T Ross, and JDD Pitout. Incidence, risk factors and outcomes of escherichia coli bloodstream infections in a large canadian region. *Clinical microbiology and infection*, 14(11):1041–1047, 2008a.

Kevin B Laupland, Terry Ross, and Daniel B Gregson. Staphylococcus aureus bloodstream infections: risk factors, outcomes, and the influence of methicillin resistance in calgary, canada, 2000–2006. *The Journal of infectious diseases*, 198(3):336–343, 2008b.

Jean-Roger Le Gall, Stanley Lemeshow, and Fabienne Saulnier. A new simplified acute physiology score (saps ii) based on a european/north american multicenter study. *Jama*, 270(24):2957–2963, 1993.

National Research Ethics Committees. Regional committees for medical and health research ethics. https://www.forskningsetikk.no/en/about-us/our-committees-and-commission/rek/, 12 2019. (Accessed on 05/14/2023).

NTNU. Computational sepsis mining and modelling - ntnu. https://www.ntnu.edu/cosem#/view/publications, a. (Accessed on 05/23/2023).

NTNU. Hunt - helseundersøkelsen i trøndelag - ntnu. https://www.ntnu.no/hunt, b. (Accessed on 05/23/2023).

Kai-Chih Pai, Min-Shian Wang, Yun-Feng Chen, Chien-Hao Tseng, Po-Yu Liu, Lun-Chi Chen, Ruey-Kai Sheu, and Chieh-Liang Wu. An artificial intelligence approach to bloodstream infections prediction. *Journal of clinical medicine*, 10(13):2901, 2021.

Harsha V Patil, Virendra C Patil, MN Ramteerthkar, and RD Kulkarni. Central venous catheter-related bloodstream infections in the intensive care unit. *Indian Journal of Critical Care Medicine: Peer-reviewed, Official Publication of Indian Society of Critical Care Medicine*, 15(4):213, 2011.

Ken Peffers, Tuure Tuunanen, Marcus A Rothenberger, and Samir Chatterjee. A design science research methodology for information systems research. *Journal of management information systems*, 24(3):45–77, 2007.

Nilay Peker, Natacha Couto, Bhanu Sinha, and John W Rossen. Diagnosis of bloodstream infections from positive blood cultures and directly from blood samples: recent developments in molecular approaches. *Clinical Microbiology and Infection*, 24(9): 944–955, 2018.

Didier Pittet, Ning Li, Robert F Woolson, and Richard P Wenzel. Microbiological factors influencing the outcome of nosocomial bloodstream infections: a 6-year validated, population-based model. *Clinical infectious diseases*, 24(6):1068–1078, 1997.

Sebastian Raschka. Mlxtend: Providing machine learning and data science utilities and extensions to python's scientific computing stack. *Journal of open source software*, 3 (24):638, 2018.

Carla Sancho-Mestre, David Vivas-Consuelo, Luis Alvis-Estrada, Martin Romero, Ruth Usó-Talamantes, and Vicent Caballer-Tarazona. Pharmaceutical cost and multimorbidity with type 2 diabetes mellitus using electronic health record data. *BMC Health Services Research*, 16:1–8, 2016.

Morten Schmidt, Jacob Bonde Jacobsen, Timothy L Lash, Hans Erik Bøtker, and Henrik Toft Sørensen. 25 year trends in first time hospitalisation for acute myocardial infarction, subsequent short and long term mortality, and the prognostic impact of sex and comorbidity: a danish nationwide cohort study. *Bmj*, 344, 2012.

St. Olavs hospital. Helseplattformen. https://stolav.no/helseplattformen. (Accessed on 05/23/2023).

David Sánchez, Montserrat Batet, and David Isern. Ontology-based information content computation. *Knowledge-Based Systems*, 24(2):297–303, 2011. ISSN 0950-7051. doi:https://doi.org/10.1016/j.knosys.2010.10.001. URL https://www.sciencedirect.com/science/article/pii/S0950705110001619.

Barbier F Tabah A Bassetti M. Timsit JF, Ruppé E. Bloodstream infections in critically ill patients: an expert statement - pmc. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7223992/#:~:text=Bloodstream%20infection%20(BSI)%20is%20defined,that%20is%2C%20without%20identified%20origin., 2020. (Accessed on 11/02/2022).

Tone: Tønjum. blodkultur - store medisinske leksikon. https://sml.snl.no/blodkultur. (Accessed on 04/15/2023).

Universitetssykhuset Nord-Norge. Laboratoriehåndbok7: Blodkultur. https://labhandbok.unn.no/mikrobiologi/blodkultur-article1561-821.html. (Accessed on 04/15/2023).

Claudio Viscoli. Bloodstream infections: the peak of the iceberg, 2016.

Visual Studio Code. Visual studio code, 2023. URL https://code.visualstudio.com/. Version 1.77.3 [Software].

Bjørn Waagsbø. Sepsis, 2022. URL https://legehandboka.no/handboken/kliniske-kapitler/infeksjoner/tilstander-og-sykdommer/bakteriesykdommer/sepsis.

World Health Organization. International classification of diseases and related health problems (icd). https://www.who.int/standards/classifications/classification-of-diseases. (Accessed on 04/09/2023).

Jionglin Wu, Jason Roy, and Walter F Stewart. Prediction modeling using ehr data: challenges, strategies, and a comparison of machine learning approaches. *Medical care*, pages S106–S113, 2010.

Haodi Zhong, Grigorios Loukides, and Robert Gwadera. Clustering datasets with demographics and diagnosis codes. *Journal of biomedical informatics*, 102:103360, 2020.

Haodi Zhong, Grigorios Loukides, and Solon P Pissis. Clustering demographics and sequences of diagnosis codes. *IEEE Journal of Biomedical and Health Informatics*, 26 (5):2351–2359, 2021.

Jack E Zimmerman, Andrew A Kramer, Douglas S McNair, and Fern M Malila. Acute physiology and chronic health evaluation (apache) iv: hospital mortality assessment for today's critically ill patients. *Critical care medicine*, 34(5):1297–1310, 2006.

Yazeed Zoabi, Orli Kehat, Dan Lahav, Ahuva Weiss-Meilik, Amos Adler, and Noam Shomron. Predicting bloodstream infection outcome using machine learning. *Scientific reports*, 11(1):1–11, 2021.

A. PostgreSQL Query for Data Selection

```
select
    prior_episodes.ppid,
    prior_episodes.no_episodes_prior,
    prior_episodes.total_duration_prior,
    prior_episodes.total_icu_count,
    prior_episodes.total_icu_duration,
    coalesce(last_episodes.dur_since_last_ep,0) as dur_since_last_ep,
    coalesce(last_episodes.duration_last_ep,0) as duration_last_ep,
    prior_episodes.icd_codes,
    prior_episodes.last_bc,
    replace(trim(prior_episodes.micr_prt_name), ',', ' ') as
    micr_prt_name,
    post_episodes.total_duration_post60d,
    p.fødtår as birthyear,
    p.kjønn as sex
from
    (
  select
      distinct blood_cultures.ppid,
      count(distinct episodes.hashid) as no_episodes_prior,
      coalesce(sum(episodes.total_duration),0) as total_duration_prior,
      coalesce(string_agg(episodes.icd_codes, ';'), '') as icd_codes,
      coalesce(sum(episodes.icu_count),0) as total_icu_count,
      coalesce(sum(episodes.total_icu_duration),0) as total_icu_duration,
      max(blood_cultures.last_bc) as last_bc,
      coalesce(string_agg(distinct blood_cultures.micr_prt_name, ';'), ''
    ) as micr_prt_name
  from
      (
    select
        coalesce(n.hashid, 'ICU_'||a.aninoppholdstart) as hashid,
        coalesce(max(n.ppid), max(a.ppid)) as ppid,
        coalesce(max(n.inndatotid), max(a.aninoppholdstart)) as
    inndatotid,
        coalesce(extract (epoch
    from
        max(n.duration)), 0) as total_duration,
        string_agg(distinct n.pdxkoder, ';') as icd_codes,
        sum(distinct a.duration) as total_icu_duration,
        count(distinct a.duration) as icu_count
    from
        aninopphold a
```

```
40      full join nimesaktivitet n on
41          a.ppid = n.ppid
42        and a.aninoppholdstart >= n.inndatotid
43        and a.aninoppholdstart <= n.utdatotid
44      group by
45          coalesce(n.hashid, 'ICU_'||a.aninoppholdstart)
46        )
47      episodes
48    right join (
49      select
50          distinct nsml.ppid,
51          max(date_req) as last_bc,
52          string_agg(distinct trim(micr_prt_name), ';') as micr_prt_name
53      from
54          nsml
55      right join
56        (
57        select
58            ppid,
59            max(date_req) as last_bc
60        from
61            nsml
62        where
63            matr_desc like '%Blod%'
64        group by
65            ppid
66        ) n2 on
67          nsml.ppid = n2.ppid
68        and nsml.date_req = n2.last_bc
69      where
70          matr_desc like '%Blod%'
71      group by
72          nsml.ppid
73      )
74    blood_cultures on
75        episodes.ppid = blood_cultures.ppid
76      and episodes.inndatotid <= blood_cultures.last_bc
77    group by
78        blood_cultures.ppid
79    order by
80        blood_cultures.ppid
81    ) prior_episodes
82  join (
83    select
84        distinct blood_cultures.ppid,
85        coalesce(sum(episodes.total_duration),0) as total_duration_post60d
86    from
87        (
88  select
89          coalesce(n.hashid, 'ICU_'||a.aninoppholdstart) as hashid,
90          coalesce(max(n.ppid), max(a.ppid)) as ppid,
91          coalesce(max(n.inndatotid), max(a.aninoppholdstart)) as
      inndatotid,
```

```
92          coalesce(max(n.utdatotid), max(a.aninoppholdslutt)) as utdatotid,
93          coalesce(extract (epoch
94      from
95          max(n.duration)), 0) as total_duration
96      from
97          aninopphold a
98      full join nimesaktivitet n on
99          a.ppid = n.ppid
100       and a.aninoppholdstart >= n.inndatotid
101       and a.aninoppholdstart <= n.utdatotid
102     group by
103         coalesce(n.hashid, 'ICU_'||a.aninoppholdstart)
104     )
105   episodes
106   right join (
107     select
108         distinct nsml.ppid,
109         max(date_req) as last_bc,
110         string_agg(distinct trim(micr_prt_name), ';') as micr_prt_name
111     from
112         nsml
113     right join
114     (
115       select
116           ppid,
117           max(date_req) as last_bc
118       from
119           nsml
120       where
121           matr_desc like '%Blod%'
122       group by
123           ppid
124     ) n2 on
125         nsml.ppid = n2.ppid
126       and nsml.date_req = n2.last_bc
127     where
128         matr_desc like '%Blod%'
129     group by
130         nsml.ppid)
131   blood_cultures on
132       episodes.ppid = blood_cultures.ppid
133     and episodes.utdatotid > last_bc
134     and episodes.inndatotid <= last_bc + interval '60 days'
135   group by
136       blood_cultures.ppid
137   order by
138       blood_cultures.ppid
139   ) post_episodes on
140     prior_episodes.ppid = post_episodes.ppid
141 left join (
142   select
143       e.ppid,
144       coalesce(max(e.duration),0) as duration_last_ep,
```

```
145       coalesce(extract(epoch
146   from
147       max(last_bc - last_episode_out_date)),0) as dur_since_last_ep
148   from
149       (
150     select
151         coalesce(n.ppid, a2.ppid) as ppid,
152         coalesce (max(utdatotid), max(a2.aninoppholdslutt)) as
      last_episode_out_date,
153         max(last_bc) as last_bc
154     from
155         nimesaktivitet n full join aninopphold a2 on n.ppid = a2.ppid
156     join (
157       select
158           distinct nsml.ppid,
159           max(date_req) as last_bc,
160           string_agg(distinct trim(micr_prt_name), ';') as micr_prt_name
161       from
162           nsml
163       right join
164         (
165         select
166             ppid,
167             max(date_req) as last_bc
168         from
169             nsml
170         where
171             matr_desc like '%Blod%'
172         group by
173             ppid
174         ) n2 on
175           nsml.ppid = n2.ppid
176         and nsml.date_req = n2.last_bc
177       where
178           matr_desc like '%Blod%'
179       group by
180           nsml.ppid
181       ) bc1 on
182         n.ppid = bc1.ppid
183       and n.utdatotid < bc1.last_bc
184     group by
185         coalesce (n.ppid, a2.ppid)
186     ) last_episode
187    join ( select coalesce(n.ppid, a3.ppid) as ppid, coalesce(utdatotid,
       a3.aninoppholdslutt) as utdatotid, coalesce(extract (epoch from n.
       duration), a3.duration) as duration from
188       nimesaktivitet n full join aninopphold a3 on n.ppid = a3.ppid) e
189     on
190       last_episode.ppid = e.ppid
191     and last_episode.last_episode_out_date = e.utdatotid
192   group by
193       e.ppid
194   ) last_episodes on
```

```
195     prior_episodes.ppid = last_episodes.ppid
196 join pasient p on
197     prior_episodes.ppid = p.ppid
```

B. SASCA Implementation in Python

This appendix include the code implementing SASCA in Python, utilized to find the clusters discussed in this research. Note that the code is heavily inspired by the code for DDSCA given by (Zhong et al., 2021). The changes from the original corresponds to the changes in the algorithm, including the distance measurements for both single and set values.

```
import numpy as np
import pandas as pd
import networkx as nx
import pickle as pkl
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
from matplotlib import pyplot as plt
from scipy.spatial import distance
import copy
import random
import math

from collections import Counter
import h5py


def create_ICD_tree():
    """ Create the networkx graph representation of the ICD-10 hierarchy.
    """
    with open('../data/icd10hier.txt', 'r') as f:
        config = f.read().splitlines()
        icd_tree = []
        for ele in config:
            line = ele.split(' ')
            icd_tree.append(line)
    cutted_icd_tree = []
    for i in icd_tree:
        cutted_icd_tree.append(i[1:])
    icd_tree_inform = sorted([list(item) for item in set(
        tuple(row) for row in cutted_icd_tree)], key=lambda x: (x[0]))
    icd_DiGraph = nx.DiGraph()
    for i in icd_tree_inform:
        icd_DiGraph.add_edges_from([(i[3], i[2]), (i[2], i[1]), (i[1], i
    [0])])

    return icd_DiGraph
```

```
35
36
37 def get_all_unique_icds_most_frequent(icd_codes_list, top_n=200):
38     """
39     Return all unique icd code found in the icd_codes_list that is
       included in the top_n frequent codes
40
41     Args:
42         icd_codes_list (nested array): A nested list representing the
       list of icd codes given to each patient
43         top_n (int, optional): The number of frequent icd codes to
       include. Defaults to 200.
44
45     Returns:
46         top_n_codes: The top n most frequent codes
47     """
48     tree = create_ICD_tree()
49     flattened_list = [code[:3]
50                       for icd_codes in icd_codes_list for code in
       icd_codes if code[:3] in tree]
51     code_counts = Counter(flattened_list)
52     sorted_codes = sorted(code_counts.items(),
53                           key=lambda x: x[1], reverse=True)
54     top_n_codes = [code for code, count in sorted_codes[:top_n]]
55
56     return top_n_codes
57
58
59 def information_content(node, tree):
60     """Calculate the information content for a given node in the given
       tree
61
62     Args:
63         node (networkx node): The current node to calculate th
       information content for
64         tree (networkx graph): Networkx graph representing the icd
       structure
65
66     Returns:
67         float: the calculated information content for the node
68     """
69     all_leaves = [node for node in tree.nodes() if tree.out_degree(node)
       == 0]
70     L = len(all_leaves)
71     leaves = len([node for node in nx.descendants(
72         tree, node) if node in all_leaves])
73     ancestors = len(nx.ancestors(tree, node))
74     t = (leaves/(ancestors + 1)) + 1
75     n = L + 1
76     return -math.log(t/n)
77
78
79 def jiang_conrath(node1, node2, tree):
```

```
    """Calulate the jiang conrath distance between two nodes from the tree

    Args:
        node1 (networkx node): The first node to calculate distance between
        node2 (networkx node): The second node to calculate distance between
        tree (networkx graph): Networkx graph representing the icd structure

    Returns:
        float: The jiang conrath distance between the two nodes given the tree
    """
    lca = list(nx.algorithms.tree_all_pairs_lowest_common_ancestor(
        tree, root='A00-Z99', pairs=[(node1, node2)]))[0][1]
    return information_content(node1, tree) + information_content(node2, tree) - 2*information_content(lca, tree)


def create_icd_jc_distance_matrix(all_unique_icds_in_tree, tree):
    """Create the distance matrix representing the jiang conrath distance between each frequent icd code in the tree

    Args:
        all_unique_icds_in_tree (array): Representing all the frequent icd codes that are present in the tree
        tree (networkx graph): Networkx graph representing the icd structure

    Returns:
        numpy.ndarray : Representing the distance matrix for each ICD code
    """
    distance_matrix = np.zeros(
        [len(all_unique_icds_in_tree), len(all_unique_icds_in_tree)])
    max_distance = jiang_conrath('A00', 'U04', tree)
    l = len(all_unique_icds_in_tree)
    for i in range(l):
        code1 = all_unique_icds_in_tree[i]
        for j in range(i, l):
            code2 = all_unique_icds_in_tree[j]
            norm_distance = jiang_conrath(code1, code2, tree) / max_distance
            distance_matrix[i][j] = distance_matrix[j][i] = norm_distance
    return distance_matrix


def find_distance(icd_list1, icd_list2, icd_distance_matrix, all_unique_icds):
    """Calculate the distance between two lists of icd codes, based on the given distance matrix and all unique frequent icd codes
```

```
120
121     Args:
122         icd_list1 (array): List of diagnose codes for the first patient
123         icd_list2 (array): List of diagnose codes for the second patient
124         icd_distance_matrix (numpy.ndarray): Distance matrix for each ICD
     code
125         all_unique_icds (array): All unique frequent icd codes in the
    tree
126
127     Returns:
128         float: The distance between the two lists, a valiue between 0 and
     1
129     """
130     icd_list1 = [code[:3]
131                  for code in icd_list1 if code[:3] in all_unique_icds]
132     icd_list2 = [code[:3]
133                  for code in icd_list2 if code[:3] in all_unique_icds]
134     if (len(icd_list1) == 0 and len(icd_list2) == 0):
135         distance = 0
136     elif len(icd_list1) == 0 or len(icd_list2) == 0:
137         distance = 1
138     else:
139         dis1 = sum(min(icd_distance_matrix[all_unique_icds.index(
140             code1[:3])][all_unique_icds.index(code2[:3])] for code2 in
    icd_list2) for code1 in icd_list1)
141         dis2 = sum(min(icd_distance_matrix[all_unique_icds.index(
142             code1[:3])][all_unique_icds.index(code2[:3])] for code1 in
    icd_list1) for code2 in icd_list2)
143         distance = 0.5 * ((dis1/len(icd_list1)) + (dis2/len(icd_list2)))
144     return distance
145
146
147 def get_outcome(row):
148     """Group the patient row to the corresponding outcome
149
150     Args:
151         row (pandas row): Row representing the values for a patient
152
153     Returns:
154         int: A number between 0-3 represetning the outcome
155     """
156     if pd.isna(row['micr_prt_name']):
157         if row['total_duration_post60d'] == 0:
158             return 0  # No BC and short stay
159         return 1  # No bc but long stay
160     if row['total_duration_post60d'] == 0:
161         return 2  # BC and short stay
162     return 3  # BC and long stay
163
164
165 def make_patient_history_df():
166     """Make the dataframe for the history of a patient from the csv-
    tables
```

```
167
168     Returns:
169         pandas.DataFrame: Dataframe represetning the history of each
    patient in the csv
170     """
171     patient_history = pd.read_csv('../data/patient_history.csv')
172     patient_history['last_bc'] = pd.to_datetime(patient_history['last_bc'
    ])
173     patient_history['age'] = patient_history.apply(
174         lambda row: row['last_bc'].year - row['birthyear'], axis=1)
175     patient_history['sex'] = patient_history['sex'].replace(
176         {'Mann': 0, 'Kvinne': 1})
177     patient_history = patient_history.drop(['birthyear', 'last_bc'], axis
    =1)
178     patient_history['icd_codes'] = patient_history['icd_codes'].fillna("
    ")
179     patient_history['icd_codes'] = patient_history['icd_codes'].apply(
180         lambda x: x.split(';'))
181
182     patient_history = patient_history.drop(patient_history.tail(1).index)
183
184     return patient_history
185
186
187 def preprocess_patient_history(patient_history, standardize=True,
    normalize=False):
188     """Preprocess the given dataframe
189
190     Args:
191         history (pandas.DataFrame): Describing the history of each
    patient
192     Returns:
193         pandas.DataFrame: The preprocessed dataframe
194     """
195
196     patient_history['total_duration_post60d'] = StandardScaler(
197     ).fit_transform(patient_history[['total_duration_post60d']])
198     bins = sorted(
199         [0] + list(patient_history['total_duration_post60d'].quantile([0,
     1.0]).values))
200     patient_history['total_duration_post60d'] = pd.cut(
201         round(patient_history['total_duration_post60d'], 3), bins=bins,
    labels=[0, 1])
202
203     patient_history['outcome'] = patient_history.apply(
204         lambda x: get_outcome(x), axis=1)
205     patient_history.drop(
206         ['total_duration_post60d', 'micr_prt_name'], axis=1, inplace=True
    )
207
208     num_cols = ['no_episodes_prior',  'total_duration_prior', '
    total_icu_count',
```

```
209                 'total_icu_duration', 'dur_since_last_ep',  '
    duration_last_ep', 'age']
210     patient_history[num_cols] = MinMaxScaler(
211     ).fit_transform(patient_history[num_cols])
212     return patient_history
213
214
215 def create_single_distance_matrix(single_vals_wo_outcome):
216     """Create the distance matrix representing the distance between each
    patient' single values, saved as a hdf54 file
217
218     Args:
219         single_vals_wo_outcome (numpy.ndarray): Nested list representing
    the single values for each patient
220     """
221     n = len(single_vals_wo_outcome)
222     batch_size = 10000
223     num_batches = (n+batch_size-1) // batch_size
224
225     min_val = float("inf")
226     max_val = float("-inf")
227
228     with h5py.File('single_distance_matrix.hdf5', "w") as hdf5_file:
229         single_distance_matrix = hdf5_file.create_dataset(
230             "single_distance_matrix", (n, n),  dtype=np.float64)
231
232         for i in range(num_batches):
233             start_i = i * batch_size
234             end_i = min((i+1) * batch_size, n)
235
236             for j in range(i, num_batches):
237                 start_j = j*batch_size
238                 end_j = min((j+1)*batch_size, n)
239
240                 batch_distances = distance.cdist(
241                     single_vals_wo_outcome[start_i:end_i],
    single_vals_wo_outcome[start_j:end_j], metric='euclidean')
242
243                 min_val = min(min_val, np.min(batch_distances))
244                 max_val = max(max_val, np.max(batch_distances))
245
246                 if i == j:
247                     single_distance_matrix[start_i:end_i,
248                                            start_j:end_j] =
    batch_distances
249                 else:
250                     single_distance_matrix[start_i:end_i,
251                                            start_j:end_j] =
    batch_distances
252                     single_distance_matrix[start_j:end_j,
253                                            start_i:end_i] =
    batch_distances.T
254         for i in range(num_batches):
```

```
            start_i = i * batch_size
            end_i = min((i + 1) * batch_size, n)

            for j in range(i, num_batches):
                start_j = j * batch_size
                end_j = min((j + 1) * batch_size, n)

                # Normalize the current batch
                single_distance_matrix[start_i:end_i, start_j:end_j] = (
                    single_distance_matrix[start_i:end_i, start_j:end_j]
    - min_val) / (max_val - min_val)

                if i != j:
                    # Normalize the symmetric batch
                    single_distance_matrix[start_j:end_j, start_i:end_i]
    = (
                        single_distance_matrix[start_j:end_j, start_i:
    end_i] - min_val) / (max_val - min_val)


def p_product_distance(selected_center):
    """Calculate the distance between each patient and the selected
    center. The distance is given by the d_jc for single and d

    Args:
        selected_center (int): The index of the current selected center

    Returns:
        array: The total distance vector representing the d_SASCA
    distance between each patient and the current seleted center
    """    """
    with h5py.File("single_distance_matrix_normalized.hdf5", "r") as
    hdf5_file:
        single_distance_matrix = hdf5_file["single_distance_matrix"]
        single_vector = [single_distance_matrix[i]
                         [selected_center] for i in range(n)]
    icd_code_vector = [find_distance(icd_codes_list[i], icd_codes_list[
    selected_center],
                                     icd_distance_matrix, all_unique_icds
    ) for i in range(n)]
    outcome_vector = [abs(single_val_to_np[i][-1] -
                          single_val_to_np[selected_center][-1])/3 for i
    in range(n)]
    p_product_vector = (w_single * (np.array(single_vector) ** 2) + w_set
     * (
        np.array(icd_code_vector)**2) + w_outcome * (np.array(
    outcome_vector)**2)) ** 0.5
    return list(p_product_vector)


def retFarthestPoint(all_centers, points_index, distance_matrix):
    """Find the next center that is furthest away from the already
    selected centers
```

```
296
297     Args:
298         all_centers (array): List of all already selected centers
299         points_index (array): All remaining patient indices
300         distance_matrix (numpy.ndarray): Current distances between each
    of the already selected centers and the patients.
301
302     Returns:
303         int: index of the point farthest away from all centers
304     """
305     distance_rows = copy.deepcopy(np.array(distance_matrix))
306     min_values = np.amin(distance_rows, axis=0)
307     indexes = np.concatenate(np.argwhere(min_values == np.max(min_values)
    )).ravel(
308     ).tolist()
309     indexes_filtered = [x for x in indexes if x not in all_centers]
310     index = random.choice(indexes_filtered)
311     return index
312
313
314 def get_cluster(centers, distance_matrix, n):
315     """Assign each record to a clusted based on the selected centers and
    the distance matrix with the  distance between each center and the
    patients.
316     Each patient is assigned to the cluster with the closest center.
317
318     Args:
319         centers (array): List of all selected centers
320         distance_matrix (numpy.ndarray): Distance matrix representing
    distance between the centers and the patients
321         n (int): The number of patients
322
323     Returns:
324         numpy.ndarray: A nested list representing the clusters, where the
     ints in the first list represent the record ids assigned to the first
     cluster
325     """
326     index_cluster = np.argmin(np.array(distance_matrix), axis=0)
327     wcss = [0] * len(centers)
328     bcss = [0] * len(centers)
329     all_clusters = [[] for i in centers]
330     k = len(centers)
331     average_distances = np.mean(distance_matrix, axis=0)
332     global_center_index = np.argmin(average_distances)
333
334     for patient_i, cluster_i in enumerate(index_cluster):
335         all_clusters[cluster_i].append(patient_i)
336         wcss[cluster_i] += (distance_matrix[cluster_i][patient_i]**2)
337
338     global_center_distances = [(distance_matrix[i]
339                                 [global_center_index]**2) for i in range(
    len(centers))]
340     for i, cluster in enumerate(all_clusters):
```

```
341         n_k = len(cluster)
342         bcss[i] = global_center_distances[i] * n_k
343     ch = (sum(bcss)/sum(wcss)) * ((n-k)/(k-1))
344     return all_clusters, ch, sum(wcss)
345 
346 
347 if __name__ == '__main__':
348     """ SASCA """
349     tree = create_ICD_tree()
350     patient_history = make_patient_history_df()
351     patient_history = preprocess_patient_history(patient_history)
352 
353     single_val_cols = [
354         col for col in patient_history.columns if col not in ['ppid', '
    icd_codes']]
355     single_val_to_np = patient_history[single_val_cols].to_numpy()
356     icd_codes_list = patient_history['icd_codes'].to_numpy()
357 
358     all_unique_icds, p = get_all_unique_icds_most_frequent(
359         icd_codes_list, top_n=200)
360     # icd_jc_distance_matrix = create_icd_jc_distance_matrix(
    all_unique_icds, tree)
361     with open('temp/icd_jc_distance_matrix_top_200.pkl', 'rb') as fp:
362         icd_distance_matrix = pkl.load(fp)
363 
364     n = len(single_val_to_np)
365     w_single = 0.5
366     w_set = 0.4
367     w_outcome = 0.1
368 
369     points_index = list(range(n))
370     random.shuffle(points_index)
371     start = points_index.pop()
372 
373     first_distance = p_product_distance(start)
374     min_k = 2
375     total_k = 30
376     k_values = range(2, total_k+1)
377     centers = [start]
378     # Start SASCA
379     # Taking first random center
380     # Point removed so we don't loop between the two farthest points
381     distance_matrix = [first_distance]
382     for k in k_values:
383         print("Finding a new center (no. {}) and calculating the distance
     matrix for this center".format(k))
384         # Finding which point is furthest from all already chosen centers
    , the distance from this point to the closest center, and the index of
     it.
385         farthest_point = retFarthestPoint(
386             centers, points_index, distance_matrix)
387         # Find next center as the point furthest away
388         centers.append(farthest_point)
```

```
389         points_index.remove(farthest_point)
390         distance_matrix.append(p_product_distance(centers[-1]))
391     clusters, ch, wcss = get_cluster(centers, distance_matrix, n)
```

Listing B.1: SASCA implementation in Python

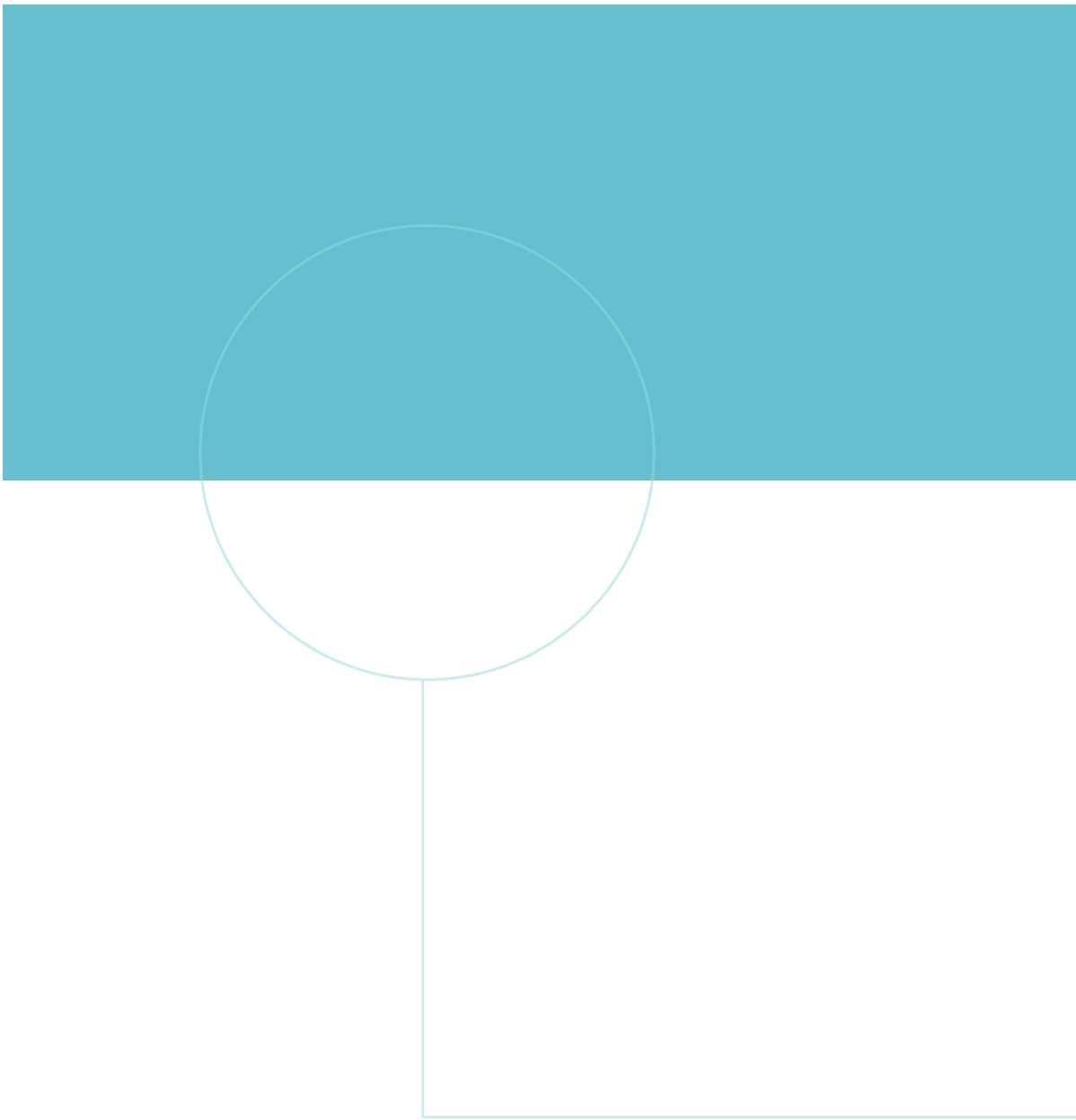
C. ICD Mapping

C.1. ICD Chapter Mapping to Description

Chapter number	Range of ICD codes	Description
Chapter 1	A00-B99	Certain infectious and parasitic diseases
Chapter 2	C00-D49	Neoplasms
Chapter 3	D50-D89	Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism
Chapter 4	E00-E90	Endocrine, nutritional and metabolic diseases
Chapter 5	F00-F99	Mental, Behavioral and Neurodevelopmental disorders
Chapter 6	G00-G99	Diseases of the nervous system
Chapter 7	H00-H59	Diseases of the eye and adnexa
Chapter 8	H60-H95	Diseases of the ear and mastoid process
Chapter 9	I00-I99	Diseases of the circulatory system
Chapter 10	J00-J99	Diseases of the respiratory system
Chapter 11	K00-K93	Diseases of the digestive system
Chapter 12	L00-L99	Diseases of the skin and subcutaneous tissue
Chapter 13	M00-M99	Diseases of the musculoskeletal system and connective tissue
Chapter 14	N00-N99	Diseases of the genitourinary system
Chapter 15	O00-O99	Pregnancy, childbirth and the puerperium
Chapter 16	P00-P96	Certain conditions originating in the perinatal period

Chapter 17	Q00-Q99	Congenital malformations, deformations and chromosomal abnormalities
Chapter 18	R00-R99	Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified
Chapter 19	S00-T98	Injury, poisoning and certain other consequences of external causes
Chapter 20	V0n-Y98	External causes of morbidity
Chapter 21	Z00-Z99	Factors influencing health status and contact with health services
Chapter 22	U00-U85	Codes for special purposes

Table C.1.: List of mappings from ICD chapters to their code range and description



NTNU

Norwegian University of
Science and Technology