UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

# MACHINE LEARNING TO DEVELOP A MODEL THAT WILL PREDICT EARLY IMPENDING SEPSIS IN NEUROSURGICAL PATIENTS

Diploma Thesis

**Evgenios Vlachos**

**Supervisor:** Michael Vassilakopoulos

September 2022

UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

# MACHINE LEARNING TO DEVELOP A MODEL THAT WILL PREDICT EARLY IMPENDING SEPSIS IN NEUROSURGICAL PATIENTS

Diploma Thesis

**Evgenios Vlachos**

**Supervisor:** Michael Vassilakopoulos

September 2022

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

# ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ ΓΙΑ ΚΑΤΑΣΚΕΥΗ ΜΟΝΤΕΛΟΥ ΠΟΥ ΘΑ ΠΡΟΒΛΕΠΕΙ ΠΡΩΙΜΑ ΕΠΕΡΧΜΟΜΕΝΗ ΣΗΨΗ ΣΕ ΝΕΥΡΟΧΕΙΡΟΥΡΓΙΚΟΥΣ ΑΣΘΕΝΕΙΣ

Διπλωματική Εργασία

**Ευγένιος Βλάχος**

**Επιβλέπων:** Μιχαήλ Βασιλακόπουλος

Σεπτέμβριος 2022

Approved by the Examination Committee:

Supervisor **Michael Vassilakopoulos**
Professor, Department of Electrical and Computer Engineering, University of Thessaly

Member **George Giannakopoulos**
Grade B/Researcher (Specialized Functional Scientist), NCSR "Demokritos"

Member **Aspasia Daskalopoulou**
Assistant Professor, Department of Electrical and Computer Engineering, University of Thessaly

# Acknowledgements

I would like to thank Mr. Vasilakopoulos for all the knowledge that provided me throughout my student years, and also his help for the preparation of my diploma thesis. In addition, I would like to express my undivided gratitude to Mr. Giannakopoulos and NCSR DEMOKRITOS for the contribution and transmission of knowledge, and also the University Hospital of Heraklion for providing and explaining the data. Without the help of Mr. Christos Tsitsipanis and Mr. Aris Salapatas the completion of my thesis would have been impossible. Finally I would like to thank my parents for the support and understanding they showed me during my student years.

# DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

«Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I also declare that the results of the work have not been used to obtain another degree. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism».

The declarant

Evgenios Vlachos

Diploma Thesis

**MACHINE LEARNING TO DEVELOP A MODEL THAT WILL PREDICT EARLY IMPENDING SEPSIS IN NEUROSURGICAL PATIENTS**

**Evgenios Vlachos**

# Abstract

Sepsis is currently defined as a “life-threatening organ dysfunction caused by a dysregulated host response to infection”. The early detection and prediction of sepsis is a challenging task, with significant potential gains regarding the lives of patients and — as such — should be researched comprehensively. The main goal of this study is to take anonymised and appropriately processed data in order to detect infections which imply future probability for sepsis. In that way, medical practitioners may have the opportunity to treat patients appropriately in a proactive manner. Feature selection techniques were applied in order to define the most important features to feed machine learning models and maximize the performance of the prediction as a binary classification problem. We also aim to highlight the relation of specific clinical input features to the prediction outcome, possibly contributing to an improved, data-driven understanding of this multi-factorial dysfunction. Early findings indicating promising classification performance, with different machine learning algorithms, but also based on appropriate feature engineering, building upon features with a time-sensitive aspect (i.e. features representing different samplings in different positions in time).

**Keywords:**

Prediction, Infection, Sepsis, Machine Learning

Διπλωματική Εργασία

**ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ ΓΙΑ ΚΑΤΑΣΚΕΥΗ ΜΟΝΤΕΛΟΥ ΠΟΥ ΘΑ ΠΡΟΒΛΕΠΕΙ ΠΡΩΙΜΑ ΕΠΕΡΧΜΟΜΕΝΗ ΣΗΨΗ ΣΕ ΝΕΥΡΟΧΕΙΡΟΥΡΓΙΚΟΥΣ ΑΣΘΕΝΕΙΣ**

**Ευγένιος Βλάχος**

# Περίληψη

Η σήψη ορίζεται ως “μια απειλητική για τη ζωή δυσλειτουργία οργάνου που προκαλείται από μια απορρυθμισμένη απόκριση του ξενιστή στη μόλυνση”. Η έγκαιρη ανίχνευση και πρόβλεψη της σήψης είναι ένα δύσκολο έργο, με σημαντικά πιθανά οφέλη σχετικά με τη ζωή του ασθενούς και — ως εκ τούτου — θα πρέπει να ερευνηθεί εκτενώς. Ο κύριος στόχος αυτής της εργασίας είναι η λήψη ανώνυμων και κατάλληλα επεξεργασμένων δεδομένων προκειμένου να ανιχνευθούν λοιμώξεις που υποδηλώνουν μελλοντική πιθανότητα για σήψη. Με αυτόν τον τρόπο, το ιατρικό προσωπικό μπορεί να έχει την ευκαιρία να θεραπεύσει κατάλληλα τον ασθενή με προληπτικό τρόπο. Εφαρμόστηκαν τεχνικές feature selection προκειμένου να καθοριστούν τα πιο σημαντικά χαρακτηριστικά για την τροφοδοσία μοντέλων μηχανικής μάθησης και τη μεγιστοποίηση της απόδοσης της πρόβλεψης που ορίστηκε ως πρόβλημα δυαδικής ταξινόμησης. Επιπλέον, στοχεύουμε να τονίσουμε τη σχέση συγκεκριμένων κλινικών χαρακτηριστικών εισόδου με το αποτέλεσμα πρόβλεψης, συμβάλλοντας πιθανώς σε μια βελτιωμένη κατανόηση αυτής της πολυπαραγοντικής δυσλειτουργίας βάσει δεδομένων. Πρώιμα ευρήματα που υποδεικνύουν πολλά υποσχόμενη απόδοση ταξινόμησης, με διαφορετικούς αλγόριθμους μηχανικής μάθησης, αλλά και με βάση την κατάλληλη “φύση” χαρακτηριστικών, βασισμένη σε χαρακτηριστικά με μια πτυχή ευαίσθητη στο χρόνο (δηλαδή χαρακτηριστικά που αντιπροσωπεύουν διαφορετικές δειγματοληψίες σε διαφορετικές θέσεις στο χρόνο).

**Λέξεις-κλειδιά:**

Πρόβλεψη, Λοίμωξη, Σήψη, Μηχανική Μάθηση

# Table of contents

# List of figures

# List of tables

# Abbreviations

| | |
|---|---|
| WBC | White Blood Cell |
| NEU | Neutrophil |
| CRP | C-reactive protein |
| HCT | Hematocrit |
| PLT | Platelets |
| GLU | Glucose |
| UR | Urine |
| CR | Creatinine |
| SBP | Systolic Blood Pressure |
| DBP | Diastolic Blood Pressure |
| PUL | Pulse pressure |
| FEV | Fever |
| SGOT | Serum Glutamic-Oxaloacetic Transaminase |
| SGPT | Serum Glutamic Pyruvic Transaminase |
| gGT | gamma-Glutamyl Transferase |
| INR | International Normalized Ratio( Prothrombin ) |
| APTT | Activated Partial Thromboplastin Clotting Time |
| K | Potassium |
| NA | Natrium |
| nlp | Natural language processing |
| DT | Decision Tree |
| GB | Gradient Boosting |
| LR | Logistic Regression |
| ML | Machine Learning |

# Chapter 1

# Introduction

Sepsis, currently defined as a "life-threatening organ dysfunction caused by a dysregulated host response to infection" [1], is one of the main factors that contribute to death and morbidity every year, and more than 700,000 patients in the US are affected by it [2]. Every year, more than six million people die from sepsis worldwide, although many of these deaths could be avoided if sepsis was identified early. In the ICU, one-third of patients pass away within 48 hours [3].

As mortality increases by 7.6% for each hour that treatment is delayed[2], early detection of sepsis improves patient outcomes either by better developing a plan to treat patient or by administering appropriate antibiotics which decreases mortality.

Many sepsis symptoms are unrecognizable and may be caused by many other clinical reasons which may affect and harm patients. Moreover, treatments for sepsis are costly, representing a cost close to $ 24 billion for the US health care system. In that way, early detection of sepsis will contribute to the reduction of those expenses, but at the same time it remains one of the most challenging problems in medicine[4].

## 1.1 Motivation

There has been a lot of research on the construction of different models, using machine learning techniques, which aimed to the on-time prediction of sepsis and septic shock[5],[4], but most of them have not been applied in practice. This is due to the heterogeneity of sepsis, with each person reacting differently and showing a variety of symptoms[6]. Also numerous existing studies perform poorly and occasionally demand time-consuming test outcomes.

Despite the effort of many researchers to develop solutions for the restriction of that phenomenon, there is a lack of specific effective therapies that work for each patient. That happens due to the complicated and inexplicable influences in host's immune system. Another reason is that the improvement of the results is not achieved because the treatment is independent on the results of each research.

In most surveys, researchers are focused on the use of numerous risk scores, which in some measure can explain some characteristics of the situation of patient and his mortality risk levels[7],[2], but they can not adequately capture the heterogeneity of sepsis and its results in a specific patient. Also, a large amount of sepsis symptoms may be caused by many other clinical factors. Moreover, the majority of many surveys used the same feature set and their findings are limited only to a specific set of data that has specific characteristics in specific patients for a certain period of time.

## 1.2 Contribution

The main goal of this study, is to find new not tested before features that are strongly related to infection in order to prevent sepsis. In that way, it is highly likely that a large amount of patients will have greater chances of survival. Also, to predict infection on time in order to avoid possible future sepsis. Based on the above, our contributions are as follows:

- We first collect an appropriate dataset of clinical data, following all ethical and deontological requirements, to form a feature rich description of each case. This dataset was provided by the General University Hospital of Heraklion.
- We then examine 4 different learning algorithms and their performance in the prediction of infection given patient data.
- We also experiment on:
    1. different representations of the input feature space, especially regarding time-related features.
    2. applying appropriate correlation analysis techniques for identifying the association between all types of variables of our dataset.
    3. different imputation methods in pre-processing.

4. finding the most important features of our input set performing feature selection techniques.
5. adding the most insignificant variables of each category in a category which represents a set of characteristics.

Also, in many surveys the available data come from hospitals that failed to record multiple measurements, which results in the uncertain performance of the model. In our study, the data is diligently recorded and correspond to reality. Furthermore, it is — to the best of our knowledge — one of few studies that examine a significant number novel (i.e. previously unused) features.

## 1.3 Related Work

There have been many studies according to the detection of sepsis by using machine learning algorithms and by building deep learning architectures. In most of them researchers tried to use conventional algorithms like linear regression, Naive Bayes [8], XGBOOST [9] etc. In many cases, the results were encouraging, but in reality they were not applicable. This was happening due to the fact that they were predicted sepsis few hours before on set. Almost all researchers were not interested in handling missing values, which in many cases affects the final result. There are some comparative studies for building classifiers for the early detection of sepsis before on-set which indicate the performance of each classifier[10]. One of the most efficient algorithms that was applied in many studies was XGBOOST achieving high levels of accuracy. A very promising study, was about a LSTM recurrent neural network that was fed by data that were pre-processed by a gaussian multitask process[11]. They predicted sepsis 4 hours before on-set using 5 vital features and 29 features that represent laboratory values. Also, they compared their results with the same LSTM model without the multitask gaussian process. Their proposed method outperforms the overly simplistic clinical scores and it is better than the LSTM without the gaussian pre-process[11].

Additionally, a different study has shown that combining NLP features with physiological data from an electronic health record can improve classification performance than just NLP or physiological characteristics alone. Those nlp features come from clinical notes of medicals and are capable of highlighting early observations that can be very important for patients clinical condition and disease's evolution. They achieved improved prediction performance, fewer false alarms and sufficient time for early intervention[12].

In the majority of studies, researchers use or develop risk scores which display patient's clinical condition or their probability of developing sepsis. Risk scores such as MEWS, EWS, SIRS, qsofa, SOFA [7] provide some limited useful information but, they can not capture the heterogeneity of sepsis and the specific symptoms of each patient because they are based on specific measures in pre-defined intervals. On the other hand, most of the studies use data that come from within the ICU, while one would hope to utilise data collected before the ICU and reduce the chances of ICU-hospitalization.

Researchers faced the early detection of sepsis as a classification, regression, clustering problem and tried to approach it by using numerous methods[13],[14]. In the majority of studies related to the early detection and prediction of sepsis, the constructed ML models use limited clinical parameters, such as demographic features and features like vital signs and lactate levels[15],[5],[10]. Nevertheless, there may be other factors that cause infection and in the future, sepsis.

In our work we try to extend the list of input features taken into account when describing the patient instance, in order to examine whether this added information can improve the algorithms predictive capacity. We also examine two different cases of pre-processing of the time-sensitive features, trying to better integrate the time-related knowledge in the machine learning models. By using this approach we try to summarize the information from our large set of characteristics to reduce computational cost and get more solid results that can be used in practice.

## 1.4 Structure of Paper

The remainder of the thesis is organized as follows. We begin by a short review of the related work in Section 1.3. We overview our method in Chapter 4, describing the data gathered and the analysis pipeline including the feature selection process. We proceed with an experimental evaluation to examine the predictive capacity of the models we examine for our given setting, in Section 3. We close the paper with work limitations (Section 5.2), the conclusion and suggestions for related work (Section 6.2) to improve our findings. Also, in 4.2.5 we review the list of parameters that took into consideration for feeding grid search method to give us the optimal parameters of our applied algorithms.

# Chapter 2

# Background

In this part of our analysis, we are going to explain some important terms of our work. Hence, the reader will have the opportunity to understand the basic terms and follow the topic and the applied methods. We organize this section by following the applied steps in our code. So, we first begin by explaining some medical definitions concerning sepsis and septic shock. Then we are going to define the applied imputation methods for handling missing values and after the feature selection methods that underlying the importance of the columns in our dataset. Moreover, we are going to give a detailed overview of the applied algorithms and the evaluation measures for testing them. And finally, we will explain the importance of clustering algorithms for adding more information in our datset, their definition and also grid search and correlation analysis.

## 2.1 Sepsis and septic shock

There have been a lot of attempts for defining sepsis and septic shock in recent years, but all of them could not define them correctly.

- Sepsis: Sepsis is now defined as a “life-threatening organ dysfunction caused by a dysregulated host response to infection”.

- Septic Shock: Septic shock can be thought of as a subtype of sepsis in which very severe metabolic, cellular, and circulatory abnormalities are associated with a significantly higher risk of death than sepsis alone. Compared to sepsis, septic shock carries a substantially higher risk of death.

- Organ Dysfunction: Organ dysfunction can be defined as an increase of 2 points or more in the Sequential Organ Failure Assessment (SOFA) score, which is associated with a hospital mortality rate of at least 10%.

## 2.2 Imputation methods

Missing data is a common issue in many clinical studies and they must be treated appropriately. There are many methods that can handle missing data, reducing bias and increasing the performance of the applied machine learning models. We chose to apply 2 different techniques which are kNN Imputation and Multiple Imputation.

- kNN imputation: The kNN approach is widely used in many domains of machine learning and has been extended to handle missing values of a dataset. It is the appropriate challenge when we do not have prior knowledge about the distribution of data. Picking a distance metric like euclidean distance or Minkowski Distance, the algorithm finds K closest neighbors of the incomplete instance and replaces the missing value with the mean or mode of the neighbors. The mode is used for handling missing categorical values. It is an efficient strategy for large datasets, but it struggles when a large percentage of data is missing. One challenging issue is to find the optimal number of neighbors, which is may a computationally costly action.
- Multiple Imputation: Multiple imputation is a widely known technique for handling missing values. Unlike single imputed methods, multiple imputation takes into consideration the uncertainty related to the imputed values. It produces maximally likely values, thus their results do not reflect the distribution of the underlying data [16] . Goal of multiple imputation is to create several versions of the missing value. All missing values are imputed n-times to represent the uncertainty of possible values that are to be imputed. Finally, the n-times values then analyzed to obtain unique combined estimations [17] .

## 2.3 Computer Science Knowledge

Computer science's field, machine learning employs statistical techniques to provide programs the ability to draw on past knowledge in order to improve performance or to make accurate predictions. Machine Learning systems are often classified to four major categories:

1. Supervised Learning : The learner uses a collection of examples with labels as training data. and provides forecasts. This is a common case involving classification and regression.
2. Unsupervised Learning: Unlabeled training data are given to the learner, who then makes predictions. It can be challenging to analyze since there are typically no annotated examples available to evaluate the effectiveness of a learner.
3. Semi-Supervised Learning Both labeled and unlabeled samples are provided to the learner who creates forecasts using data.
4. Reinforcement Learning. We use machine learning for many applications in our daily life, such as checking if an email is spam or not, or detecting if a transaction is fraud.

Apart from the theoretical definition of machine learning, it is necessary to explain some important definitions that are related with machine learning and with the training of the applied models in a specific dataset.

- Classification: The method for determining the class of a set of data points is called classification. Targets, labels, and categories all serve to describe classes. An approximate mapping function from input variables to discrete output is the purpose of categorization variables.
- Deep learning: The backpropagation algorithm is used in deep learning to determine and suggest changes to the parameters that a machine should make in order to calculate the representation in each layer using the results and representation from the previous layer. Deep Convolutional neural networks have greatly improved numerous fields, also Recurrent neural networks have an incredible contribution on sequential data such as voice and audio, as well as image processing, video, speech recognition, and audio at the same time. A neural network with three or more layers that can make predictions is said to be deep learning architecture. These neural networks attempt to replicate the actions of the the capacity of the human brain to learn from massive volumes of data. While a neural network with a single layer can approximate forecast events with poor accuracy, deep learning can optimize and improve accuracy by utilizing additional hidden layers. Applications that use deep learning can significantly

increase automation by carrying out mental and physical tasks without the need for a human. Many modern technology, common goods and services, as well as upcoming technologies, are powered by deep learning. Also, deep learning algorithms are used extensively used to determine the most important features and distinguish each class from the other.

- Cross validation: Is a method that is used to train a specific machine learning model based on several train-test splits. In that way, we can understand better the behavior of a model on non-seen before data.
- Leave One Out Cross Validation: It is a method to evaluate the performance of a specific machine learning model without splitting our dataset into training and testing. It is a very useful technique for small datasets, while it is computationally costly for large datasets[18].Starting with all observations except for one, it divides the dataset into training and testing sets. We calculate the mean squared error of the predicted value and the value that was left out of training after the model has been trained using the training set. Then we repeat this process as the number of observations and calculate the average mean squared error for each iteration. The advantage of Leave One Out Cross Validation is that offers unbiased results, but it could be computationally expensive.
- Training set: The sample of data that is used for fitting and training our machine learning model. The applied model is trained and learn patterns on this data.
- Testing set: The testing set is used for evaluating a machine learning algorithm. Moreover, is the set of data which provide an unbiased evaluation of a final model fit on the training set.

## 2.4 Scaling

We need to standarize our data, because in many columns there is a significant difference between their values, hence after the training of our predictive models we might ended up facing overfitting. There are many ways of standarizing our data. Variables that are measured at various scales may not all equally contribute to the model fitting, creating a bias outcome. We examined two different approaches. The first was applying standard scaler, and the second to apply Min Max Scaler.

- Standard Scaling: It is appropriate when the distribution of the features follows normal distribution. It substracts the mean of the column from the value of the feature in a specific record and devides it by the standard deviation of the feature[19]. One disadvantage of standard scaler is that it is sensitive to outliers.

- Min Max Scaling: The Min Max Scaler will transform all records of a specific feature between 0 and 1. It scales values within a specified range without changing the shape of the original distribution [20]. $X_{new} = \frac{X_i - \min(X)}{\max(X) - \min(X)}$, where X is a specific feature of the dataset.

## 2.5 Feature Selection

- Chi square: We were able to comprehend the link between our category variables with the use of the chi-square test. A high Chi-Square value suggests that the independence between the two variables hypothesis is false. The feature can be chosen for model training when the Chi-Square value rises because it depends more on the answer.

  $x_c^2 = \sum \frac{O_i - E_i}{E_i}$, where c=degrees of freedom, O= observed values, E= expected value.

## 2.6 Algorithms

In this section we analyze 4 machine learning algorithms we applied in our pre-processed data to test their performance in the prediction of infection. Firstly, we decided to apply XGboost in order to test its efficiency due to its high performance in many studies. Also, we tested Gradient Boosting, Logistic Regression and Decision Trees. All these 4 methods are describing below.

- XGBoost: XGBoost(eXtreme Gradient Boosting) algorithm is not only popular for predicting sepsis, as it can be applied to numerous tasks with efficient results. One advantage of this algorithm is that it can work with missing data, so it does not require feature engineering. Also, XGBoost is more appropriate to be applied in small datasets than neural network architectures , [21].Decision trees are trained sequentially on the training set and in order to improve the objective function, a new decision tree is added in each iteration [22].

$L^t = \sum_{i=1}^{n}[l(y_i, \hat{y_i}^{t-1} + f_t(x_i))] + \Omega(f_t)$, where l is the loss term, $\Omega$ is the regularization term,$y_i$ is the actual value of the ith instance ,$\hat{y_i}$ is the prediction we make of the ith instance ,$f_t$ is the function of tree and n is the number of instances in the training set .

$\Omega(f_t) = \gamma T_t + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$, where $\Omega(f_t)$ is applied to penalize the model to avoid overfitting, $\gamma$ and $\lambda$ are hyperparameters,T is the total number of leaves in the tree and w is the weight of each leaf.

By taking the second order Taylor approximation we conclude to the following equation: $L^t \cong \sum_{i=1}^{n}[l(y_i, \hat{y_i}^{t-1} + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i))] + \Omega(f_t)$ Where

$g_i = \partial_{\hat{y_i}^{t-1}}\, l(y_i, \hat{y_i}^{t-1})$

is the first order gradient statistic of the loss function.

and $h_i = \partial^2_{\hat{y_i}^{t-1}}\, l(y_i, \hat{y_i}^{t-1})$ is the second order gradient statistic of the loss function.

The latter two equations, respectively, offer the ideal leaf weight in a leaf node j and the related optimal value of the objective function for a fixed tree structure, where $I_j$ denotes the instance set of leaf j.The scoring function L(q) is used for measuring the quality of the tree structure. $w_j^* = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}$

$L(q) = -\frac{1}{2} \sum_{j=1}^{T} \frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} + \gamma T$

- Logistic Regression:

  Logistic regression is frequently used to assess the probability that a certain instance belongs to a certain class. The Logistic Regression technique employs a linear equation with independent or explanatory elements to forecast a response value. It’s a slightly unique form of linear regression. This variable is categorical in nature. The log of odds serves as the dependent variable. The model predicts that the instance belongs to that class if the probability estimation is greater than 50%; otherwise, it does not. Consequently, it’s a binomial classifier. Linear Regression Equation:

$$z = \beta 0 + \beta 1 X1 + \beta 2 X2 + \ldots\ldots + \beta n X n$$

  Where, The dependent variable is z, and the explanatory variables are $\mathrm{x1}, \mathrm{x2}\ldots$ and $\mathrm{Xn}$. **Sigmoid Function:** This projected response value (z) is then translated into a probability value between 0 and 1. To transfer expected values to probability values, we employ the sigmoid function.

$$\phi(z) = \frac{1}{1 + e^{-z}}$$

Apply Sigmoid function on linear regression: $p = \frac{1}{1+e^{-(\beta 0+\beta 1X1+\beta 2X2....\beta nXn)}}$ Properties of Logistic Regression:

- In logistic regression, the dependent variable follows the Bernoulli Distribution. It necessitates that the observations be unrelated to one another. As a result, the findings should not be based on repeated measurements.
- The independent variables should not have a high degree of correlation.
- The sample sizes determine the success of the Logistic Regression model. To attain great accuracy, a large sample size is usually required.
- Maximum likelihood is used for estimation.

The modeling hypothesis that maximizes the likelihood function can be found by maximizing

$$\text{sum i to } n \log(P(yi \mid xi; p))$$

(p is our logistic regression model)

- Decision Trees: A decision tree is a procedure which includes separations from the root to achieve a boolean outcome in the leaves. Decision trees are powerful machine learning techniques that is used in various domains like image processing and to identify patterns. The tree is consisted of the root, the intermediate nodes and the leaves that include the final outcome [23]. One important measure while constructing a decision tree is the entropy, which only lies between 0 and 1. The closer it is to zero the better.

  Entropy = $\sum_{i=1}^{c} P^i log2^{P_i}$, where $P_i$ is the ratio of the sample number of the subset , and i is the ith attribute value

  Another metric is information gain which is the amount of information carries each node before splitting them. It is the exactly opposite of entropy, and the closer is to 1 the better for our final result.

  Information gain = $\sum_{v \in V(A)} \frac{|S_v|}{|S|} Entropy(S_v)$, Where the range of attribute A is V(A) and $S_v$ is a subset of set S equal to the attribute value of attribute v.

  Steps of Decision Tree ID3 algorithm:

  - It starts with the original set S as the root node.

  - On each iteration of the process, the algorithm computes the entropy and the information gain of each unused attribute.
  - After these calculations, it selects the attributes with the largest information gain or the smallest entropy.
  - Then, the selected attribute splits the S set, producing a subset of the data.
  - The algorithm visits each remaining subset considering only attributes that never visited before until it reaches to the leaf nodes to take the prediction.

- Gradient Boosting: Is a supervised machine learning technique which combines decision trees using a gradient boosting method. Due to its ability to comprehend complicated patterns, it has found widespread use in numerous fields, including credit risk assessment and transportation crash prediction, electrical circuit forecast and defect prognosis. It employs least squares, then only one replacing the function minimization problem with parameter optimization based on the original criterion and attaining excellent performance [24], [25].

  Goal of gradient boosting trees is to minimize the expected value of the loss function which is constructed using the predicted and the actual value by creating a weak learner $h_t(x_i; a)$ iteratively that points in the the negative gradient direction [26].

  Gradient boosting procedure follows the next 7 steps:

  - Firstly, we have to specify our loss function. A common loss function is the square root which is defined below:

    $L(y_i, f(x_i)) = \frac{1}{2}(y_i - f(x_i))^2$
  - Step 2: The algorithm begins by initializing the weak learner

    $f_0(x) = argmin \sum_{i=1}^{n} L(y_i, \gamma)$
  - Step 3: . Since L is selected to be the square loss function, $f_0(x)$ becomes$f_0(x) = \frac{\sum_{i=1}^{n} y_i}{n}$
  - Step 4: Then it computes the gradient with respect to the predicted outcome $r_i m = -\frac{\partial L(y_i, f(x_i))}{df(x_i)}$, where m=1 to M and i is the index of the observations.
  - Step 5: After that, the algorithm calculates the $\gamma_m$to solve the optimization problem $\gamma_m = argmin \sum_{i=1}^{n} L(y_i, f_{m-1}(x_i) + \gamma f_m(x_i))$

- Step 6: By solving the above euation we get $\gamma_m = \frac{\sum_{i=1}^{n} h_m(x_i)[y_i - f_{m-1}(x_i)]}{\sum_{i=1}^{n} h_m(x_i^2)}$
- Step 7: Lastly, we update the f function $f_m(x) = f_{m-1}(x) + \gamma_m h_m(x)$

## 2.7 Grid Search

Most of machine learning models will not manage to achieve optimal performance due to the incorrect selection of the hyperparameters. So, in order to optimize our applied alagorithms is is essential to perform hyperparameter-tuning, because inappropriate parameters lead to inefficient results. We can not know in advance the optimal parameters of our applied model, so without the searching methods we had to try all the combinations manually, which may take long time [27]. GridSearchCV is a method that is used along machine learning algorithms to perform hyperparameter-tuning. Grid search optimizes the parameters of the machine learning model using cross validation which splits the data into k subsets. The k-1 subsets are used as training sets and the other one subset is used as an evaluating set. In that way many combinations are produced and the most accurate one is selected to train our algorithm. Although, grid search is not appropriate to apply when we deal with large amount of parameters [28]. All parameters values are passed in a dictionary and grid search tries all different combinations using the cross validation with defined cv. At the same time an evaluating tool is applied and the combination with the best score is chosen. Moreover, we pass some necessary parameters in the grid search method when we call it.

- estimator: it is our pre-defined model that we want to optimize its parameters.
- params_grid: It includes the set of the parameters we want to check in a dictionary form.
- scoring: The method we want to use to evaluate the grid search findings.
- cv: Cross validation value, which defines the number of attempts to try for specific hyperparameters.
- verbose: When we set it to 1 we could have a detailed view of the of the applied combination.
- n_jobs: When we set it equal to 1, grid search will use all the available cores of our machine.

## 2.8 Evaluations Tools

- Accuracy: Represents the rate of correct classifications. In a binary classification problem.

  Accuracy = $\frac{TruePositive+TrueNegative}{TruePositive+FalsePositive+FalseNegative+TrueNegative}$. In a general classification problem, Accuracy = $\frac{Numberofcorrectpredictions}{Totalnumberofpredictionsmade}$

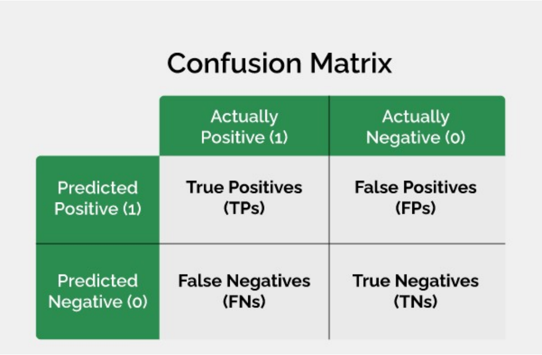
- Specificity: The percentage of negative cases which predicted as negative, Specificity $= \frac{TN}{TN+FP}$

- Sensitivity-Recall: The proportion of positive cases that were predicted as being positive., $recall = \frac{TP}{TP+FN}$

- Log Loss: Log Loss is one of the most important measures for evaluating the performance of a classification problem. The Log Loss tells us how close is the prediction probability to the actual label which is one or zero for a binary classification problem. The smaller the Log Loss is, the more efficient is the outcome of a machine learning algorithm concerning the prediction. Our goal is to minimize that metric to take optimal results. The perfect model has Log Loss equal to zero. It is the appropriate metric when the output of our model is the probability of the appearance of a binary result.

  LogLoss = $\frac{-1}{N} \sum_{i=1}^{N} \sum_{j=1}^{M} x_{ij} * log(p_{ij})$

- Confusion Matrix: The confusion matrix is a matrix of the model predictions against the ground-truth labels, and it is an important tool for evaluating classification performance. Instances in a predicted class are represented by the rows of the confusion matrix, while the occurrences in an actual class are represented by the columns. The diagonal values of the matrix represent the right predictions for various classes, while on the other hand the off-diagonal elements represent misclassified examples. Also it is an important tool for summarizing useful information about the performance of our dataset and identify some interesting patterns.



Confusion Matrix

| | Actually Positive (1) | Actually Negative (0) |
|---|---|---|
| Predicted Positive (1) | True Positives (TPs) | False Positives (FPs) |
| Predicted Negative (0) | False Negatives (FNs) | True Negatives (TNs) |

Figure 2.1: Confusion Matrix representation.

## 2.9 Clustering

- K-means: This algorithm's objective is to group n observations into K clusters, where each observation will be grouped with the cluster that has the closest mean. One of the most straightforward and well-liked unsupervised machine learning algorithms for clustering data is K-means clustering. Assuming we have the input data points $x_1, x_2, x_3, , x_n$ and value the number of clusters K. We follow the below steps:
  - We select K points at random or the first K from the input data to serve as our initial centroids.
  - Next, we calculate the Euclidean distance between each dataset point and the identified K points (cluster centroids).
  - The distance we discovered in step ii is then used to assign each data point to the closest centroid.
  - We next take the average of the points in each cluster group to determine the new centroid.
  - Finally, we repeat the proccess of steps 2 to 4 for a fixed number of iteration or until the centroids will remain the same. Euclidean Distance between two points in space: d(p,q) = $\sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}$

Assigning each point to the nearest cluster: If each cluster centroid is denoted by $c_i$, then each data point x is assigned to a cluster based on arg min dist$(c_i, x)^2 c_i \in C$ where dist() is the euclidean distance Finding the new centroid from the clustered group of points: $c_i = \frac{1}{|S_i|} \sum_{x_i \in S_i} x_i$

$S_i$ is the set of all points assigned to the ith cluster.

- Agglomerative Hierarchical Clustering: In this algorithm, we begin by assuming each data point as a subcluster. We set a metric to calculate the distance between all pairs of subclusters at each step and keep merging the nearest two subclusters in each step. We repeat the same process until there is only one cluster left in the system[29]. In our data we used single Linkage measurement which is the distance between closest elements in clusters, $D(c_1, c_2) = minD(x_1, x_2)$, where $x_1 \in c_1, x_2 \in c_2$.

  Steps of Agglomerative Clustering:

  - We assign each data point as a single cluster.
  - We define what distance measure we are going to use(e.g euclidean distance) and after that we calculate the distance matrix.
  - Determine the linkage measurements to merge the clusters.
  - Update the distance matrix.
  - We repeat the same process until every data point become one cluster.

  Linkage measurements:

  - Single Linkage : Distance between closest elements in clusters, $D(c_1, c_2) = minD(x_1, x_2)$, where $x_1 \in c_1, x_2 \in c_2$
  - Complete Linkage: Distance between farthest elements in clusters, $D(c_1, c_2) = maxD(x_1, x_2)$, where $x_1 \in c_1, x_2 \in c_2$ Average Linkage: Average of all pairwise distances
  - $D(c_1, c_2) = \frac{1}{c_1}\frac{1}{c_2}\sum_{x_1 \in c_1}\sum_{x_2 \in c_2} D(x_1, x_2)$

  Centroids: Distance between centroids (means) of two clusters,

  $D(c_1, c_2) = D((\frac{1}{c_1}\sum_{x \in c_1} \overrightarrow{x}), (\frac{1}{c_2}\sum_{x \in c_2} \overrightarrow{x}))$

## 2.10 Correlation Analysis

Correlation analysis is a way of finding relationships between the features of a dataset and how strong this relationship is. We can have negative correlation, positive correlation and no correlation. There are many ways of finding the correlation between two features and depends on the nature of the data. Pearson's approach measures the relationship between linearly related variables which are normally distributed. Additionally, Spearman's rank correlation and Kendall's rank correlation are metrics that gauge the degree of dependence between two characteristics and variables, respectively. The Spearman's rank correlation is the most appropriate correlation matrix in our situation to comprehend our data since it does not make any assumptions about the distribution of the data and is the best correlation analysis when the variables are measured on at least ordinal scales. These methods were applied only for measuring correlations between numerical features. For measuring the correlation between numerical and categorical and categorical with categorical features we applied two different techniques. For the first case, we applied ANOVA test in 95% confidence interval to see if the mean of our numeric variable changes with different values of the categorical variable. That doesn't give us a correlation, but it tells us if there's a relationship. For the second case we performed chi squared test in 95% confidence interval. The values of the contingency table will be determined by the assumption that two variables are independent. These variables have to be evenly distributed. After that, we measure the distance from uniform the actual values are. The probability that the null hypothesis is correct is known as the P-Value. P-Value<0.05 is the sole case where the assumption is accepted (H0). If P-Value less than 0.05 indicates that the two characteristics under consideration are connected.

### 2.10.1 Pearson Correlation Coefficient

Pearson's correlation measures the linear relationship between 2 variables. It is a statistical measure that is used in many domains, for example in data analysis, classification, regression, clustering or biological and finance analysis. Also, the direction of the linear relationship can be seen from the sign of the correlation. It can be considered as the covariance of the two variables divided by the product of their standard deviations.

$r_{xy} = \frac{\sum(x_i-\bar{x})\sum(y_i-\bar{y})}{\sqrt{\sum(x_i-\bar{x})^2}\sqrt{\sum(y_i-\bar{y})^2}}$

Where $\bar{x}$ and $\bar{y}$ indicate the mean value of x and y respectively. The range of $r_{xy}$ is from -1 to 1. A value close to -1 or 1 indicates correlation between x and, while correlation close or equal to 0 indicates no linear correlation.

### 2.10.2 Spearman Rank Correlation

A measure of relationship between two variables that are at least ordinal is the Spearman rank correlation. It assesses both the strength and the direction of the link between two variables. We use Spearman rank correlation for data which failed the requirements of Pearson's correlation coefficient and are ordinal or continuous data. It does not measure the linear relationship between two variables, but the strength of the monotonic association between them. Monotonic relationship means that when the one variable increases, the other variables also increases or decreases

$r_{xy} = 1 - \frac{6\sum d_i^2}{n(n^2-1)}$ , where $d_i$ is the pair distance of the ith element of the two variables and n is the total number of instances.

Spearman rank correlation is appropriate to use when we deal with ranked data with an observed monotonic relationship and we want to conclude if two variables are correlated.

### 2.10.3 Kendall Rank Correlation

As Spearman's rank correlation, Kendall's rank correlation measures the monotonic association between two ranked variables and the direction of their relationship. It is also an alternative of Pearson's correlation and it requires ordinal or continuous data. It is more appropriate the Spearman when we deal with few instances with many tied ranks.

$\tau = \frac{n_c - n_d}{\sqrt{(n_0-n_1)(n_0-n_2)}}$

$n_0 = \frac{n(n-1)}{2}$, where n is the number of instances,

$n_c$ is the number of paired observation that has their difference has the same sign,

$n_d =$ is the number of paired observation that has their difference has the opposite sign,

$n_1 = \frac{\sum t_j(t_j-1)}{2}$, where $t_j$ is the number of x values tied in the jth value,

$n_2 = \frac{\sum u_k(u_k-1)}{2}$, where $u_k$ is the number of y values tied in the kth value.

### 2.10.4 Chi square

The chi square test is used to determine if two categorical variables are independent comparing two hypothesis tests. When doing a null hypothesis test, we presume that the alternative hypothesis, we assume that there is no correlation between the two variables while for the alternative hypothesis there is a strong correlation between the two variables, hence we can forecast the other variable's value based on the first. To apply chi square test, we calculate p_value and if it is less or equal than a significant limit, we can safely conclude that there is sufficient relationship between the two categorical variables.

However, chi square test is very sensitive the the total sample of instances and in that way, and it is highly likely that trivial relationship between variables may appear to be statistically significant. Also, this test does not measure the strength of a relationship or the direction of their association, but only tells us if the two variables are related to each other.

### 2.10.5 Anova

We use Anova test for measuring the correlation between a numerical and a categorical variable. It measures if there is a significant difference between the mean of each numerical value for every categorical value. We conclude that if the p_value is less or equal to a significant limit, the variables are correlated. If p_value is greater than 0.05 we accept the null hypothesis and the two variables are not correlated.

# Chapter 3

# Data

In this chapter, we are going to explain our data in detail, which we gained access from the University Hospital of Heraklion. Also, we are going to explain each medical measurement, how we pre-processed them to use it properly, some changes that we performed and finally some graphs in order to understand and find some interesting patterns that are exported from our data.

## 3.1 Data Collection

We received appropriately processed and anonymised patient data from the General University Hospital of Heraklion for the purposes of the study. The dataset is consisted of a combination of demographic features and many medical measurements such as white blood cells, systolic blood pressure and C-reactive protein (CRP). Another important features is patient's mortality, which is removed from our used feature list. Our dependent value is related with the probability a patient has infection or not. This was a first collection of data. In the coming months we expect the addition of numerous of data to confirm our findings. In the paper we were limited to this small amount of data.

We note that each medical measurement is repeated within a period of 5 days. Each day of a medical measurement is represented as an independent column in the dataset. In addition, concerning the non-infected patients, 5-day measurements for each laboratory feature are used in a consecutive random 5-day period. The fifth day of each laboratory measurement represents the day in which the patient was sent for culture in order to confirm their infection.

Our dataset consists of 47 records with 242 features in each. There are several missing values (2150 such problems exist in the data), which were dealt with using two different imputation approaches, as described in the following paragraphs. Of these 47 records, 28 are patients which have an infection, while the other 19 form our control group, i.e. do not have an infection, rendering our dataset a binary classification dataset. We stress that our dataset contains a large number of features, hence, it includes the majority of features that have been used in other researches that exist in literature[1].

## 3.2 Data Description

- Sex: Gender of each patient that was admitted to the hospital.
- Age: The age of each patient.
- Days of Hospitalization: Total number of days that each patient spent within hospital.
- Hospitalization in ICU: A binary column that informs if the patient was admitted within ICU or not.
- Total Days in ICU: This feature is about both infected or non-infected patients and is the total number o their hospitalization in the ICU.
- ICD 10: It is the International Classification of Diseases and is the international standard for defining and reporting diseases and health conditions.
- Disease: Is an abnormal condition that affects the human body and can cause many symptoms like pain or dysfunction and sometimes death in specific occasions. This column summarizes patient's diseases in a list.
- Surgery: What type of surgery or surgeries was performed in each patient to face their illnesses.
- Comorbidities: It occurs when the same person suffers from two or more illnesses at the same time. So for each patient there is a list that summarizes each illnesses.
- Identified Microbe: The microbe that was identified after the culture that taken from each patient that suffered from infection.

[1] As future work we plan to further study the intricacies of using different subsets of features, taking into account other existing datasets.

- IDENTIFICATION CULTURES OF THE MICROBES: What type of culture was performed to identify the microbe that causes infection.

- Antibiotic: What specific antibiotic was administered in an infected patient.

- Duration of Taking Antibiotic Therapy: Total days of antibiotic administration in a specific patient.

- ROUTE OF RECEIVING ANTIBIOTIC TREATMENT: From which part of the body the administration of the antibiotic was made.

- WBC:Is a type of blood contained in the human body and plays an important role for fighting against bacteria and infections. An adult human contains approximately 4000 to 10,000 WBC. Some symptoms of WBC disorder are: chronic infections, weight loss, and weakness [30].

- NEU: Are a type of WBC and lead the immune's system response as they constitute the 55-70% of them. Normal neutrophil levels for an adult range from 4,500 to 11,000/mm$^3$.

- CRP: A c-reactive protein test measures the level of c-reactive protein (CRP) in the blood and is produced by the liver.It is sent into the bloodstream in response to an inflammation. A CRP test may be done to monitor conditions that can cause inflammation. High value of CRP maybe means possible inflammation.

- HCT: By measuring HCT(Hematocrit) we can see the proportion of red blood cells, which carry oxygen throughout our body, in our blood and in that way to help your our doctor to make a proper diagnosis or monitor our response to a specific treatment. Normal results vary, but in general they are for male 40.7% to 50.3% and for women 36.1% to 44.3%.

- PLT: Platelets are cells that help the clot of the blood. When we observe few platelets there may be a sign of cancer, infections or other health problems. Too many platelets may mean a risk for blood clots or stroke. Normal platelet level range is from 150,000 to 450,000.

- GLU: GLU(Glucose) is a type of sugar and is our body's main source of energy. Insulin helps to move glucose from our bloodstream into our cells. It is necessary to check our levels of glucose, because too much or too little glucose can be a sign of a serious medical condition. High Glucose levels may be a sign of Diabetes. Glucose level less than 140 mg/dL (7.8 mmol/L) is considered normal.

- UR: Urine along with creatinine is used to evaluate kidney function. If we observe not properly work of kidney urine levels increase. On the other hand, if there is severe liver disease, urine levels decrease. Normal urine values from 10 to 50 mg/dL.

- CR: Creatinine is a reliable indicator to show the proper work of the kidney. High Creatinine levels signifies impaired kidney function or kidney disease. Normal Creatinine levels for a healthy women vary from 88 to 128 mL/min and for a healthy male from 97 to 137 mL/min.

- SBP: The force that propels blood through the circulatory system is known as blood pressure, and because tissues are pushed to receive nutrients and oxygen, it is crucial for organs. Systolic blood pressure, or SBP, is used to determine blood pressure and with DBP (Diastolic Blood Pressure) they define blood pressure. Systolic blood pressure allows us to gauge the force that each time the heart beats, it places on the artery walls. A healthy person's blood pressure ranges from 90 to 120 mmHg.

- DBP: It is the measure of the pressure that the heart exerts while rests between beats. Normal systolic blood pressure for a healthy person is between approximately 60 and 80 mmHg.

- PUL: Pulse pressure is the difference between the SBP and DBP of someone's blood pressure. With that measure, we can observe health problems before developing symptoms and is an indicator of diseases. Also, PUL can indicate decreased cardiac output. A normal PUL for a healthy person varies between 40 and 60 mmHg, for example if SBP = 120 and DBP = 80, then PUL = 120 - 80 = 40mmHg

- FEV: Fever is technically the temperature of a person. We observe high values of fever when someone has a disease and its normal value varies from 36.4°C to 37.2°C.

- SGOT: The serum glutamic-oxaloacetic transaminase is a measure of the one of two liver enzymes which is increased when liver damage or liver disease exists. It evaluates how much of the liver enzyme is in the blood. The normal values for SGOT vary between 5 to 40 per liter of serum.

- SGPT: The Serum glutamic pyruvic transaminase is an enzyme that is found in liver and heart cells. When a liver damage appears, SGPT is released into blood increasing its value. SGPt levels are also increased due to some medications. The normal values for SGPT vary between 5 to 56 per liter of blood serum.

- gGt: The gamma-glutamyl transferase (GGT) measure estimates the level of the enzyme GGT in the blood. The purpose of this test is to monitor diseases of the liver or bile ducts, but it can not diagnose the exact cause of liver disease. Hence, it is usually implemented along with other tests. Normal gGT values for healthy adults are between 5 to 40 U/L.

- INR: It is a measure for evaluating the total time for the blood to clot. It helps the blood to remain its consistency and is a protein produced by the liver. In healthy people INR is approximately equal to 1.1 or below.

- APTT: Activated Partial Thromboplastin Clotting Time is a measure for checking the work of clotting factors, but it can used with other tests at the same time. It is one of several blood coagulation tests. A normal APTT range is around 21 to 35 seconds, but the results may vary due to the equipment and the used methods.

- K: Potassium test is used for detecting or diagnose kidney diseases which may appear due to high potassium levels. Normal potassium values are between 3.6 to 5.2 millimoles per liter (mmol/L). When low values of potassium are observed and especially less than 2.5 mmol/L, rapid medical actions are required.

- NA: Natrium is a type of electrolyte, which help to control the amount of fluid and the balance of acids and bases in our body. Natrium can help the nerves and muscles to work properly. A normal Natrium range is between 135 and 145 milliequivalents per liter (mEq/L).

- MORTALITY:A column that informs if a patient died within his admission in the ICU or in the hospital.

- Infection: A binary column that represent if a patient is infected or not. 0 means that is not infected and 1 that is infected.

## 3.3 Excel modifications

Before going to the pre-processing stage, we performed some changes to our dataset in our excel file to enable us to process our data more efficiently. We observed that there were many rare categories in our categorical columns which after their encoding will not contribute to formation of the result. Hence, we replaced those values with the word other to form a category that contains all those rare categories. In that way we could see the other category prevails from the most frequent categories.

Moreover, we added 4 new columns to concentrate the information from the Comorbidities, Microbes, Diseases and antibiotics columns. Each new column contains the total number of Comorbidities, Microbes, Diseases and antibiotics of each record. We believed that with that method we would avoid the creation of a sparse matrix due to the transformation of all categorical columns. To test our idea, we ran our experiments with these two formulations.

Finally, we added our target value which called Infection in order to predict a binary outcome for our patients. Our first dataset contained only patients with infection, while the second contained only non-infected patients. For that reason we knew which patient had infection or not.

.

## 3.4 Graphical explanation of data

At this point, we considered that it is necessary for the better understanding of our the data, to proceed with the construction of figures that will highlight relationships between the data. By this procedure, patterns are projected through that cannot be perceived with naked eye. We used python libraries such as matplotlib and seaborn with their functionalities.

Through this process, we can understand in an easier way our available data and use them in an efficient way when we move on the training of machine learning algorithms. Data visualization is necessary for understanding both small and large datasets and can solve significant difficulties.

To begin with, we chose to summarize mortality of the patients based on their gender in the figure 3.1. We observe that there are 22 out of 27 males and 16 out of 20 females that survived. Due to the importance of the column days of hospitalization, we decided to concentrate on this feature and how it acts with other important features. Figures 3.2 and 3.3 are related with the days of hospitalization for patients under and over 40 years old. We wanted to see how age may be connected with the their admission in hospital. The amount of patients we are dealing with is mainly older patients, but we can see longer days of hospitalization in people over 40 with the average value being higher than those under 40. In the next figure(figure 3.4) we can see the distribution of days of hospitalization based on the amount of patients that died or not. We can easily see that there are many patients that spent many days within hospital but finally they did not manage to survive. Although, there are some patients that spent many days hospitalized but in the end they survived. In figure 3.5 the amount of patient that died or not based on if they had infection or not. Only one patient died without being infected. The next plot(figure 3.6) indicates the amount of patients that died or not if we take into consideration their admission within ICU. The symbol - indicates that we were not aware of the patient admission or not within ICU. The figure 3.7 shows the the mean PLT level. The mean value of the mean PLT level is around 127248 with a standard deviation of 154935. Figure 3.8 indicates the average PLT levels based on the days of hospitalization in a scatter plot. We can observe that there are many patients with values beyond normal values. As Figure 3.8, Figure 3.9 shows the mean Ggt levels based on the days of hospitalization. We can see that the large amount of patients had stable ggt values with only some outliers that exceed normal values.

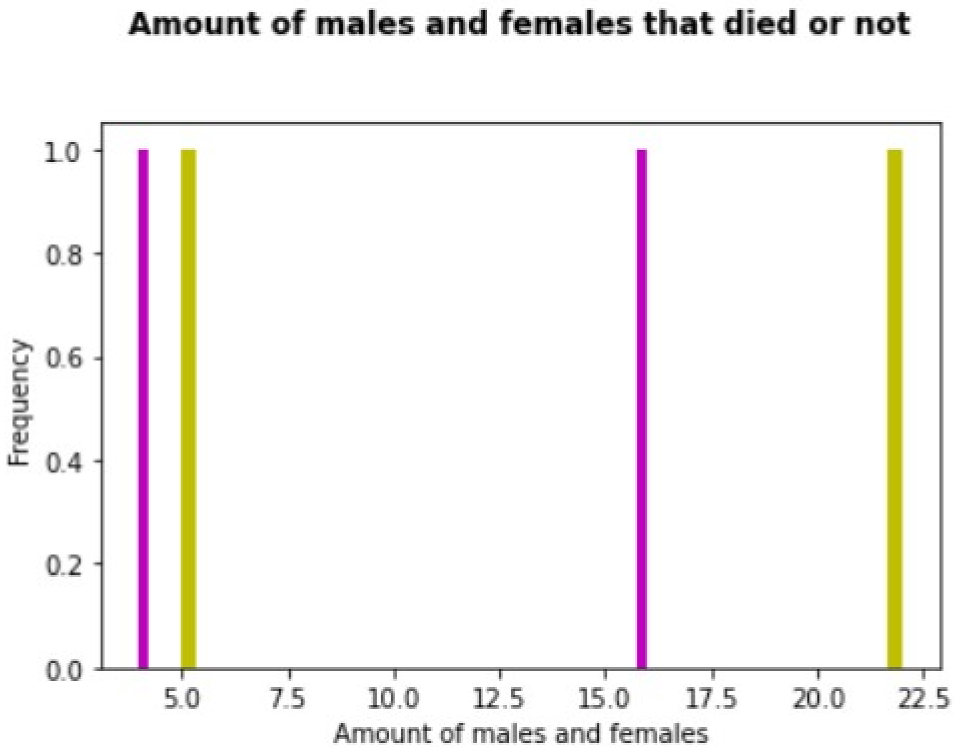


Figure 3.1: Graphical description of mortality between males and females in our dataset.

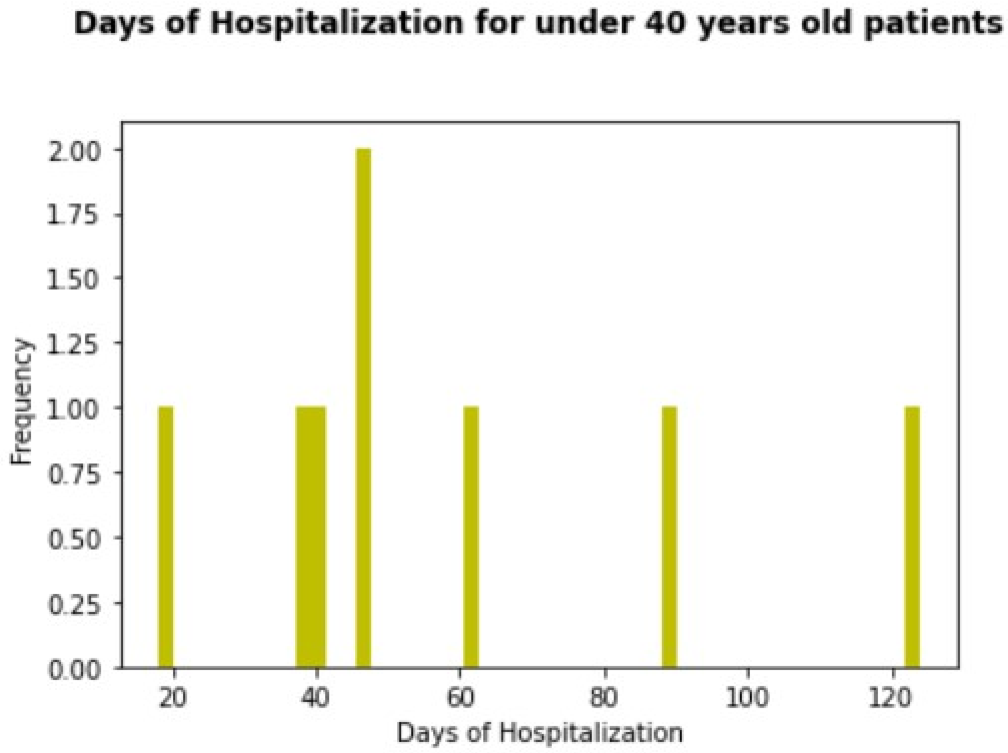


Figure 3.2: Days of hospitalization for under 40 years old patients.

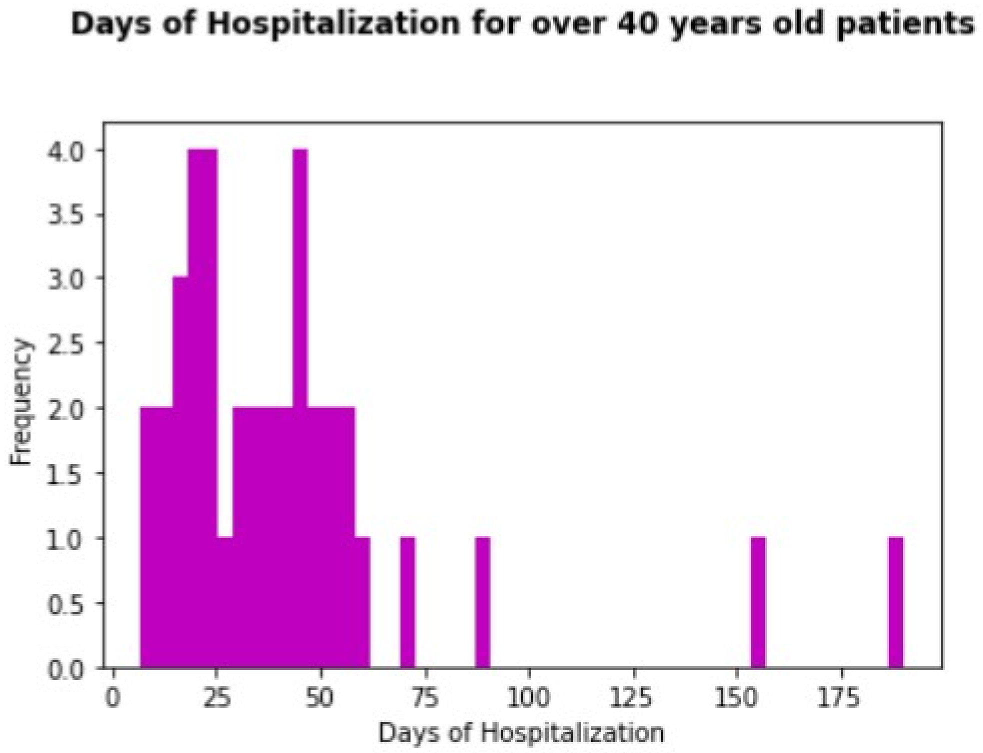


Figure 3.3: Days of hospitalization for over 40 years old patients.

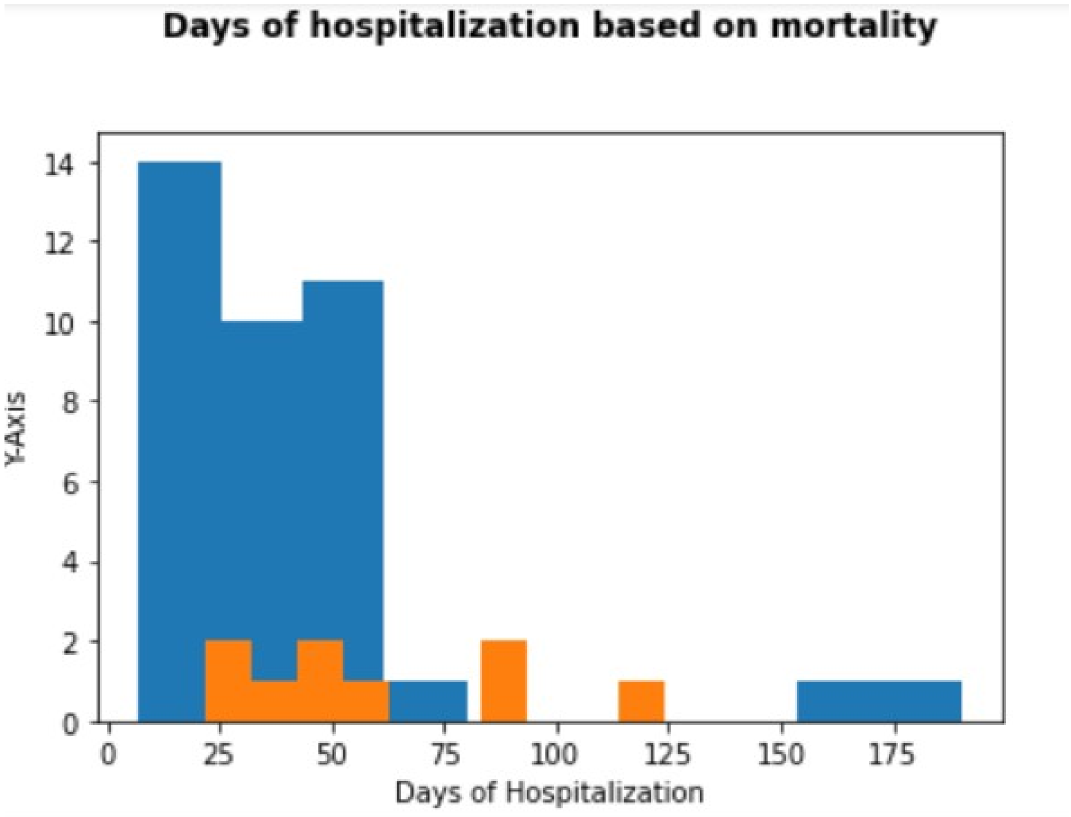


Figure 3.4: Days of hospitalization based on mortality of each patient.

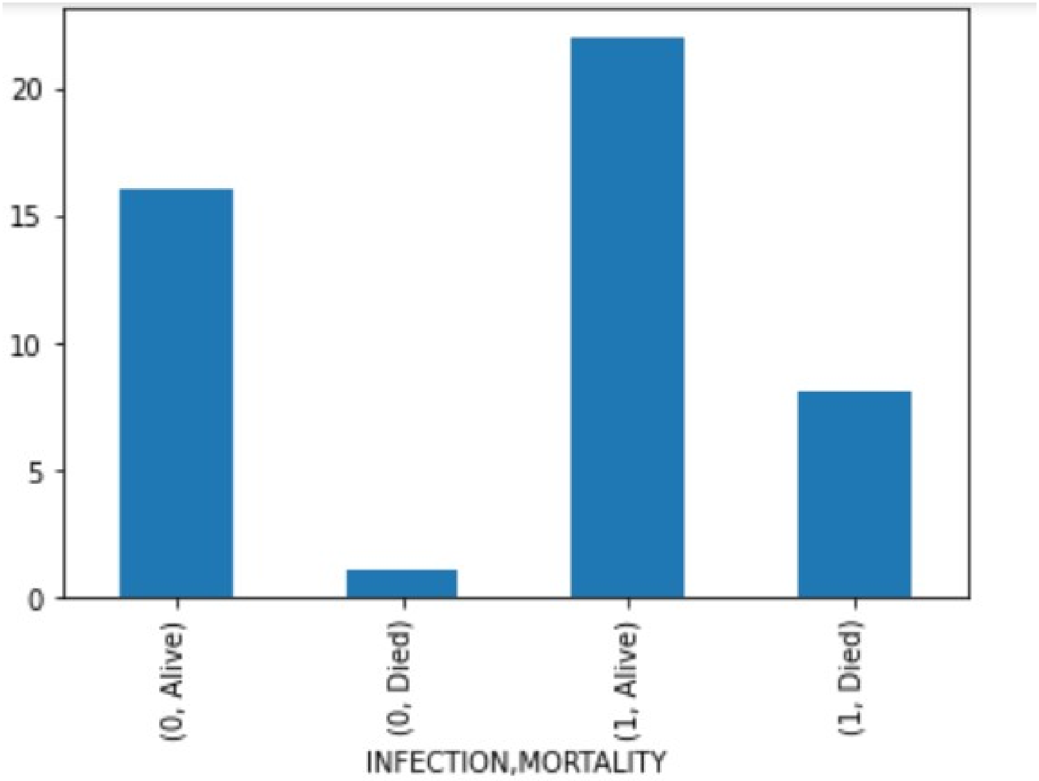


Figure 3.5: Graphical representation of how many patients died or not based on infection.

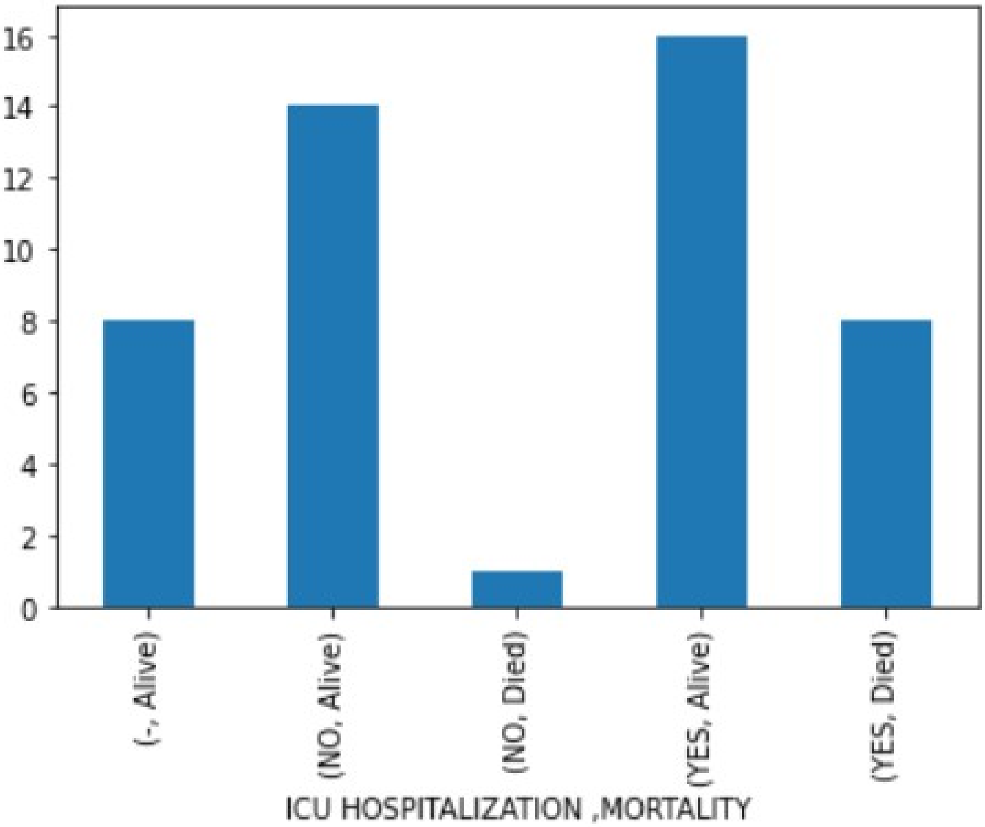


Figure 3.6: Figure of mortality count based on the admission of patients within ICU

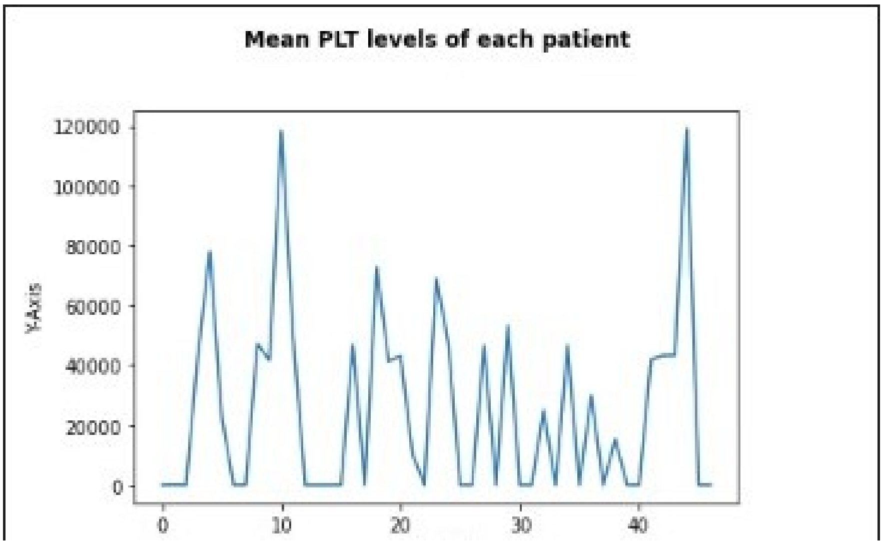


Figure 3.7: Plot of mean PLT values for each patient.

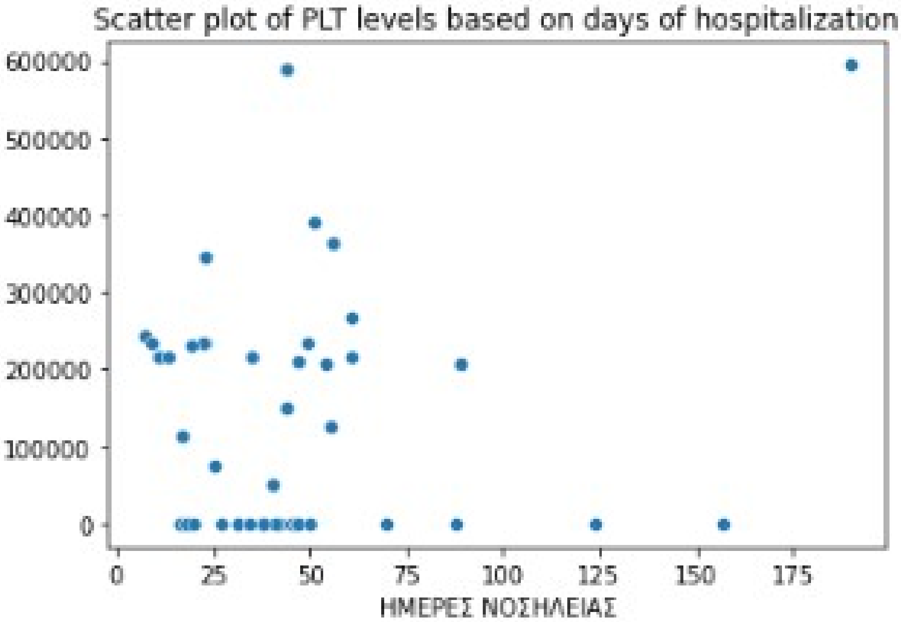


Figure 3.8: Average PLT levels based on the days of hospitalization.

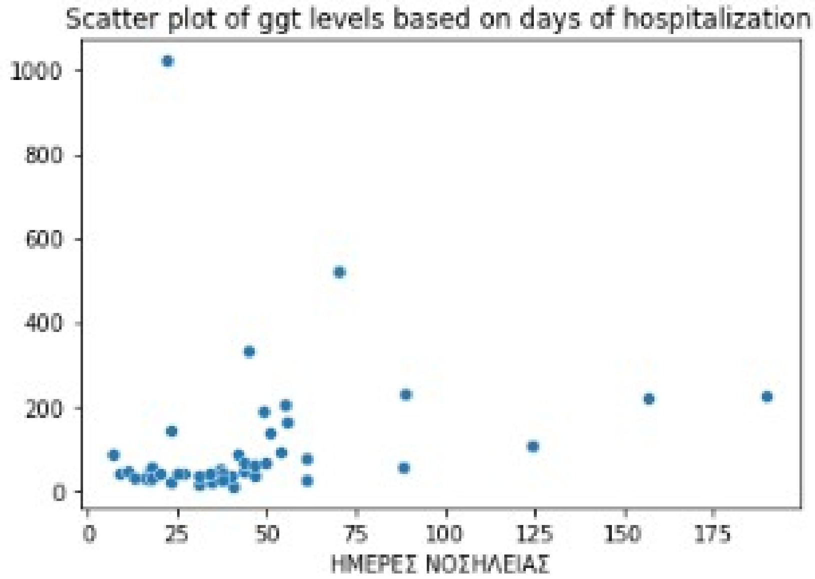


Figure 3.9: Average Ggt levels based on the days of hospitalization.

# Chapter 4

# Proposed Method

In this section, we are going to explain in depth the pipeline of our method to early predict infection to avoid impending sepsis. Our process contains the stages of: data collection, data cleaning, data pre-processing, imputation methods, K-means and Agglomerative Hierarchical Clustering, scaling, feature selection, grid search, train of the model using optimal features and using leave one out cross validation and prediction.

## 4.1 Problem Definition

Through the application of machine learning methods we aim at the timely prediction of infection in neurosurgical patients in order to avoid future sepsis that will have disastrous results for patients. Because of the numerous features of our dataset, we wanted to find the most important features and see if they match those in the literature. With the early prediction of the infection, the patient will have the possibility, by administering appropriate antibiotics, to deal with the infection he has in time and to achieve better results for him. In addition, doctors will have more time to devise an effective plan to deal with the infection. So in conclusion, we try to face sepsis by early detection of the infection which is an early stage of sepsis.Sepsis is a complex condition that is different for each person due to its heterogeneity, with symptoms varying from person to person.

## 4.2 Our Approach

### 4.2.1 Data Pre-Processing

The initial approach was the existence of this five-day measurements after the confirmation of the infection, but in the end we considered that such a thing would not give us the desired results and we would not have substantial and useful conclusions. We first replaced some categorical columns with columns that contain their information with one or zero and summed patient's comorbidities, illnesses and surgeries. Then in order to replace missing values, we applied KNN and multiple imputation in our data.

We note that the process of replacing these missing values could have only be avoided if we had a large number of data. In such as case, we could have deleted the records contain missing values, utilizing effectively a reduced dataset (and possibly losing important information). However, since we only have 47 instances in our dataset, every record is important. Furthermore, this tackling of the missing values allows our model to function in a more realistic setting, since missing values can be quite common in the actual gathered data.

### 4.2.2 Clustering Step

After the replacement of missing values, we had to deal with the lack of many records and the plethora of independent features in our dataset. So, in order to compact the information from our data, we decided to use clustering algorithms in two ways and then to decide which is the most efficient. Firstly, we created 5 clusters as the number of days of each laboratory measurement and treat them as categoricals. In essence, we considered 5 different datasets with each dataset corresponds to the columns of the same day. Secondly, we create one cluster for every 5 columns that represents one laboratory result within five days. So we ended up with 19 new categorical columns and adapted them in our dataset. Thus, we applied K-means and Agglomerative Hierarchical Clustering to compare our results[31],[29].

### 4.2.3 Scaling and correlation step

After data cleaning and data pre-processing, it was essential to apply scaling in our dataset, because we observed large deviation between the values of different columns. So we used MinMaxScaler in our dataset except the infection column to take values between zero and one. We then applied chi-square test for finding the correlations between categorical values, and ANOVA test for finding the correlations between numerical and categorical values. We found that between categorical values, the most correlated features were the identified microbe with identification culture, disease with surgical operation and ICU admission with surgical operation.

The below numerical and categorical features are correlated:

- Hospitalization in ICU - Surgery: The P-Value of the ChiSq Test is: $4.62 * 10^{-2}$
- Disease - Surgery: The P-Value of the ChiSq Test is: $4.78 * 10^{-6}$
- Identified Microbe - identification culture: The P-Value of the ChiSq Test is: 0.027

  By observing the correlation matrix between numerical values, we observed the following strong correlations.

  - Correlation between WBC and NEU: 0.98
  - Correlation between INR and APTT: 0.92

  Strong positive correlation means that as the value of the one variable increases, the value of the other feature increases in a similar pace and it does not mean that the one affects the other and vice versa.

### 4.2.4 Feature Selection Step

The next step was to apply feature selection methods to identify the most significant features of our dataset. This would help us to perform firstly data reduction getting rid of the noise in data and using only relevant features. Also feature selection was very useful for finding possible not tested before features in other surveys that maybe take into serious consideration when deal with patients that possibly develop infection. We found that the most significant features was the 5 day PLT measures, the 5 day $\gamma$gt measures and the day of hospitalization. Specifically, we applied chi-squared using SelectKBest, which computes chi-squared stats between each non-negative feature and class.

### 4.2.5 Model training and Grid search

Finally, after taking the most relevant features of our data, we perform grid search along with our selected machine learning models.

In order to find the optimal parameters in our 4 machine learning models to predict infection, we applied grid search method. We used grid search with cv=3, scoring = accuracy, refit = l2 and verbose = 1 along with our applied model and the dictionary of the parameters. So for each algorithm, we took their optimal parameters. Below we analyze which parameters we selected to apply in order to choose the best ones.

- GradientBoostingClassifier:

  1. learning_rate: 0.1,0.01,0.001,0.005
  2. n_estimators:50,100,150,200
  3. criterion:friedman_mse, squared_error, mse
  4. min_samples_split:2,4,7
  5. min_samples_leaf:1,2,3
  6. max_depth:3,5,9
  7. tol:0.001,0.0001,0.00001
  8. max_features:auto, sqrt, log2, None

- DecisionTreeClassifier:

  1. criterion:gini, entropy,log_loss
  2. min_samples_split:2,3,4,5
  3. max_depth:2, 6, 12, 15 , 4, 8
  4. ccp_alpha:0.0,0.1,0.2,0.3
  5. max_features:auto, sqrt, log2

- LogisticRegression:

  1. penalty:l1, l2, elasticnet, none
  2. tol:0.01, 0.001, 0.0001,0.00001,0.000001
  3. solver:lbfgs,liblinear
  4. max_iter:10,50,150, 250, 100,300
  5. C:np.logspace(-4,4,20)

- XGBoost:

1. subsample:0.5, 0.75, 1
2. colsample_bytree:0.5, 0.75, 1
3. max_depth:2, 6, 12, 15 , 4, 8, 12
4. min_child_weight:[1,5,15, 25, 10]
5. learning_rate:0.3, 0.1, 0.03, 0.1, 0.001, 0.001, 0.5
6. n_estimators:100
7. max_delta_step:0,5,10,8
8. alpha:0,1,2,3,4

We firstly tried using GradientBoosting Classifier, and we found that the most opimal model were by using the following parameters:

GradientBoostingClassifier(max_features='auto', n_estimators=50, random_state=0, tol=0.001,learning_rate = 0.1, criterion = friedman_mse,min_samples_split = 2, min_samples_leaf =1, max_features = None )

Then we tested the decision tree classifier, and we found the next optimal parameters: DecisionTreeClassifier(max_depth=2, max_features= auto, random_state=0 ,criterion = gini, ccp_alpha= 0.0 )

Furthermore, the logistic regression model using grid search gave as the following optimal parameters: LogisticRegression(C=0.089, max_iter=50, n_jobs=-1, tol=0.01, solver=lbfgs, penalty=l2)

And finally for the XGBoost classifier we ended up with the below model: XGBClassifier(alpha=0,

colsample_bytree=0.5,

learning_rate=0.3, max_delta_step=0, max_depth=2, min_child_weight=1, n_estimators=100, n_jobs=-1, random_state=0, subsample=0.75)

## 4.3 Experimental Evaluation

In this section we are going to explain which techniques we used during our analysis to evaluate our applied models. There are many classification metrics that are used extensively in literature according to each independent task[32],[33]. We used recall or sensitivity and specificity. We trained our models with different parameters until we get the optimal performance, thus we took multiple outcomes for each metric. Below, we are going to explain their contribution and their meaning.

- TN rate(specificity): The percentage of negative cases which predicted as negative $specificity = \frac{TN}{TN+FP}$
- TP rate(sensitivity- recall): The percentage of positive cases which predicted as positive $recall = \frac{TP}{TP+FN}$

The questions we meant to answer through the experiments and the above metrics were:

- Do different imputation approaches affect our results significantly, given the multitude of missing values?
- How does the preprocessing of the time-sensitive features affect our classification results? And more precisely, can different clustering approaches and re-engineering of these features improve prediction?
- How well can we predict infection based on our measurements? And which features appear to have the highest predictive value?

# Chapter 5

# Experiments and Discussion

In this chapter, we are going to analyze in depth our experimental ideas in order to evaluate and testing the applied models in our dataset by providing analytical figures. Also, we are going to mention some limitations of our study due to the small amount of the available data and some observations on our findings.

## 5.1 Experiments

To decide which imputation method to use for handling missing values in our dataset, we used two different approaches: kNN-based imputation and multiple imputation. Tables 5.1, 5.2, 5.3, 5.4 5.5, 5.6 5.7, 5.8 5.9, 5.10 5.11, 5.12 5.13, 5.14 5.15, 5.16 show how each approach performs in the presence of different clustering approaches with or without clustering and grid search. Also, the last 6 tables show the performance of each machine learning model by adding the most insignificant variables of each category in a category which a set of characteristics. We found that both kNN and Multiple Imputation offer a significant increase in the performance of both applied models (LR and Decision Tree) and at the same time contribute on decreasing the Log Loss. Thus, we finally concluded to use either kNN Imputation or Multiple Imputation.

Another part of our study related to generating new features based on time-sensitive features (such as PLT) by clustering. Our results showed that the selection of a clustering-based strategy for feature engineering bring no significant difference in the performance. So there is little reason to apply either of the two clustering strategies (5 clusters or one cluster for each 5 day feature) to improve performance.

| Algorithm with MI | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| Sensitivity | 1.00 | 0.00 | 1.00 | 1.00 |
| Specificity | 0.90 | 0.60 | 1.00 | 1.00 |
| Log_loss | 2.30 | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.1: Sensitivity and Specificity of prediction for Multiple Imputation case.

| Algorithm with kNN Imputation | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| Sensitivity | 1.00 | 0.00 | 1.00 | 1.00 |
| Specificity | 0.90 | 0.60 | 1.00 | 1.00 |
| Log_loss | 2.30 | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.2: Sensitivity and Specificity of prediction for kNN Imputation case.

| Algorithm with MI | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| K-means | 1.00 | 0.00 | 1.00 | 1.00 |
| Agglomerative Hierarchical Clustering | 1.00 | 0.00 | 1.00 | 1.00 |
| Log_loss 1 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |
| Log_loss 2 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.3: Sensitivity of prediction for different clustering approaches for the Multiple imputation case.

| Algorithm with MI | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| K-means | 1.00 | 0.60 | 1.00 | 1.00 |
| Agglomerative Hierarchical Clustering | 1.00 | 0.60 | 1.00 | 1.00 |
| Log_loss 1 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |
| Log_loss 2 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.4: Specificity of prediction for different clustering approaches for the Multiple imputation case.

| Algorithm with<br>kNN Imputation | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| K-means | 1.00 | 0.00 | 1.00 | 1.00 |
| Agglomerative Hierarchical<br>Clustering | 1.00 | 0.00 | 1.00 | 1.00 |
| Log_loss 1 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |
| Log_loss 2 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.5: Sensitivity of prediction for different clustering approaches for the kNN imputation case.

| Algorithm with<br>kNN Imputation | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| K-means | 1.00 | 0.60 | 1.00 | 1.00 |
| Agglomerative Hierarchical<br>Clustering | 1.00 | 0.60 | 1.00 | 1.00 |
| Log_loss 1 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |
| Log_loss 2 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.6: Specificity of prediction for different clustering approaches for the kNN imputation case.

| Algorithm with<br>Multiple Imputation | DT | LR | XGBoost | SVM |
|---|---|---|---|---|
| K-means | 1.00 | 0.78 | 1.00 | 0.57 |
| Agglomerative Hierarchical<br>Clustering | 0.86 | 0.78 | 0.86 | 0.57 |

Table 5.7: Sensitivity of prediction for different clustering approaches for the multiple imputation case without scaling and grid search.

| Algorithm with Multiple Imputation | DT | LR | XGBoost | SVM |
|---|---|---|---|---|
| K-means | 1.00 | 0.66 | 1.00 | 1.00 |
| Agglomerative Hierarchical Clustering | 1.00 | 0.66 | 1.00 | 0.57 |

Table 5.8: Specificity of prediction for different clustering approaches for the multiple imputation case without scaling and grid search.

| Algorithm with KNN Imputation | DT | LR | XGBoost | SVM |
|---|---|---|---|---|
| K-means | 0.83 | 0.80 | 0.83 | 0.40 |
| Agglomerative Hierarchical Clustering | 0.83 | 0.80 | 0.89 | 0.4 |

Table 5.9: Sensitivity of prediction for different clustering approaches for the kNN-based imputation case without scaling and grid search.

| Algorithm with KNN Imputation | DT | LR | XGBoost | SVM |
|---|---|---|---|---|
| K-means | 0.89 | 0.80 | 0.89 | 0.60 |
| Agglomerative Hierarchical Clustering | 0.89 | 0.80 | 0.89 | 0.60 |

Table 5.10: Specificity of prediction for different clustering approaches for the kNN-based imputation case without scaling and grid search.

| Algorithm with MI | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| Sensitivity | 1.00 | 0.00 | 1.00 | 1.00 |
| Specificity | 0.90 | 0.60 | 1.00 | 1.00 |
| Log_loss | 2.30 | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.11: Sensitivity and Specificity of prediction for Multiple Imputation case by adding the most insignificant variables of each category in a category which a set of characteristics.

| Algorithm with<br>kNN Imputation | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| Sensitivity | 1.00 | 0.00 | 1.00 | 1.00 |
| Specificity | 0.90 | 0.60 | 1.00 | 1.00 |
| Log_loss | 2.30 | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.12: Sensitivity and Specificity of prediction for kNN Imputation case by adding the most insignificant variables of each category in a category which a set of characteristics.

| Algorithm with MI | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| K-means | 1.00 | 0.00 | 1.00 | 1.00 |
| Agglomerative Hierarchical<br>Clustering | 1.00 | 0.00 | 1.00 | 1.00 |
| Log_loss 1 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |
| Log_loss 2 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.13: Sensitivity of prediction for different clustering approaches for the Multiple imputation case by adding the most insignificant variables of each category in a category which a set of characteristics.

| Algorithm with MI | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| K-means | 1.00 | 0.60 | 1.00 | 1.00 |
| Agglomerative Hierarchical<br>Clustering | 1.00 | 0.60 | 1.00 | 1.00 |
| Log_loss 1 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |
| Log_loss 2 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.14: Specificity of prediction for different clustering approaches for the Multiple imputation case by adding the most insignificant variables of each category in a category which a set of characteristics.

| Algorithm with<br>kNN Imputation | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| K-means | 1.00 | 0.00 | 1.00 | 1.00 |
| Agglomerative Hierarchical<br>Clustering | 1.00 | 0.00 | 1.00 | 1.00 |
| Log_loss 1 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |
| Log_loss 2 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.15: Sensitivity of prediction for different clustering approaches for the kNN imputation case by adding the most insignificant variables of each category in a category which a set of characteristics.

| Algorithm with<br>kNN Imputation | DT | LR | XGBoost | GB |
|---|---|---|---|---|
| K-means | 1.00 | 0.60 | 1.00 | 1.00 |
| Agglomerative Hierarchical<br>Clustering | 1.00 | 0.60 | 1.00 | 1.00 |
| Log_loss 1 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |
| Log_loss 2 | $9.92*10^{-16}$ | 13.81 | $9.92*10^{-16}$ | $9.99*10^{-16}$ |

Table 5.16: Specificity of prediction for different clustering approaches for the kNN imputation case by adding the most insignificant variables of each category in a category which a set of characteristics.

## 5.2 Discussion and Limitations

The goal of this study is to predict infections, which may result in sepsis, as early as possible. We gathered appropriately processed data in order to execute our analysis and concluded in some first observations. However, the amount of available data was very limited, so our conclusions should be confirmed on another dataset or on an extended version of our current data.

According to pre-existing studies, lactate levels, WBC and blood pressure seemed to affect the appearance of infection, but in our study we observed that PLT and gGt are highly possible to affect a patient's infection. We also observed that CR measurement is one of the least significant variables in our dataset. However, the limited number of data instances reduces our confidence in the generalizaiton of these findings. Moreover, the amount of features is disproportionate to the amount of records, which may bias learning. Also, concerning the replacement of missing values, it is possible that the predicted values do not correspond to reality, hence there may be differences in the final results. In that way we may have eliminated some important time dependent features in the dataset. In figure 5.1 we can see the first 11 most significant features that can be split in three different categories: PLT, gGt and days of hospitalization. PLT features are by far the most significant features of our dataset. The next figure shows a scatter plot of mean PLT levels on y-axis and gGt measures on x-axis. We showed that figure because, PLT levels and gGt levels are the two most significant measures in our dataset. We can observe that there are many patients that are beyond the normal limits of these two measures.

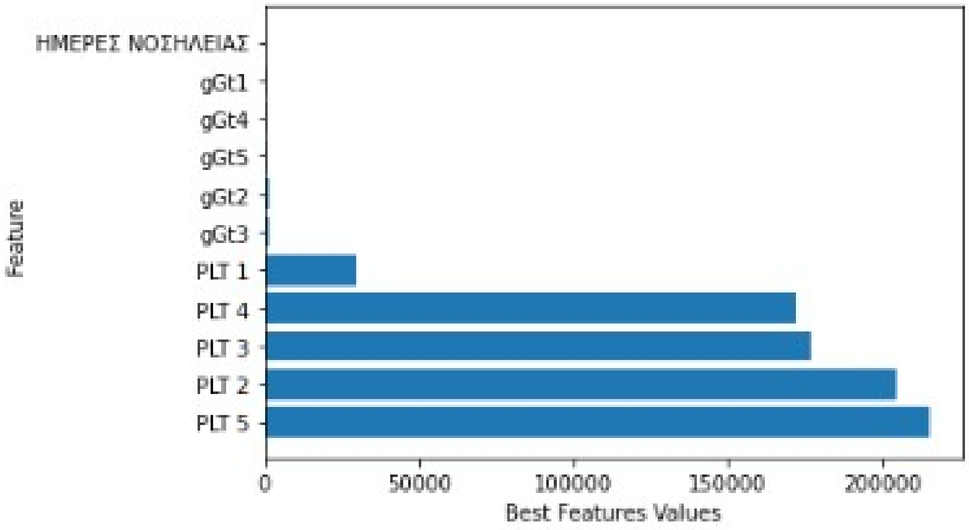


Figure 5.1: Bar plot of top 11 features

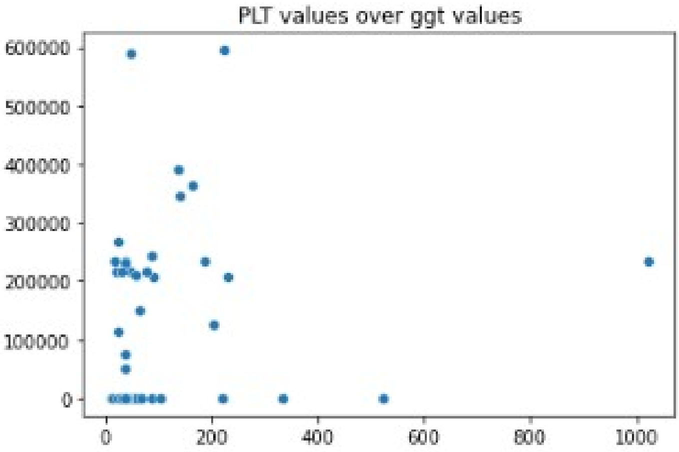


Figure 5.2: Plot of mean PLT levels with mean Ggt levels in a scatter plot

## 5.3 Experimental Setup

We ran our code using Jupyter Notebook taking advantage of numerous libraries of python programming language. The code scripts are located in Github in the following link:

https://github.com/eugenevlaxos/Github-thesis. We used jupyter notebook for performing our analysis. In order for someone to run the code, it is necessary to install Anaconda following the rules in the next link: https://sparkbyexamples.com/python/install-anaconda-jupyter-notebook/ . Also, the easiest approach is to run the code using

google colab(https://colab.research.google.com/), uploading the necessary files or by installing Jupyter Notebook in cmd.

Our dataset is not available due to personal information restrictions.

# Chapter 6

# Conclusion

We would outline the study prepared in the framework of this thesis, while also offering some potential future improvements to this project.

## 6.1 Summary and Conclusions

In this work, we evaluated the use of machine learning for the prediction of infections that can lead to sepsis in hospitalized patients. To this end, we gathered an appropriate dataset with various features (demographic, clinical and others) and appropriate labeling. We then proposed a pre-processing pipeline to perform feature engineering. This pipeline consists of the imputation of missing values using kNN imputation, then the creation of clusters to adapt them as columns in our dataset, as well as a feature selection process. Based on the resulting output, we train four different machine learning models and evaluate them using appropriate classification metrics. Based on our analysis, we observed that the most significant feature (in terms of predictive capacity) was the 5-day PLT measurement and the gGt along with the days of hospitalization. We can predict the final result with promising performance and detect infection early in order to avoid possible future sepsis. The majority of our applied models can predict infection with high efficiency. This is due to the fact that we applied grid search method for finding the optimal parameters of our machine learning models and also that we used very powerful machine learning models. However, we consider it is necessary to enrich our available dataset by adding many new instances so we can generalize our findings. Furthermore, a bigger dataset will change our findings according to the most important features, and it is possible that PLT and gGT measures will not contribute at this level to the to the formation of the final result. Moreover, we confirm the high efficiency of algorithms that have been used extensively in other researches on a dataset that contains features that have not been used to a large extent. For the early detection of sepsis, the basic prerequisite is the prediction of infection, which can be done on a daily basis so that the medical staff is aware of the health condition of each patient. In case of infection, the medical staff will be able to treat the patient appropriately by giving him the appropriate treatment and subjecting patient to additional tests.

## 6.2 Future Work

As a future work, we need to gather additional data to further support these preliminary findings and test them in our applied machine learning models or in other algorithms to test their performances. Then we need to examine the appropriateness and robustness of our preprocessing pipelines also in different datasets (if possible) to increase the generic applicability of our approach. Finally, we may examine how to deal with missing values / partial data through related individual prediction models to further improve the features input to the prediction model.

# Bibliography

[1] Singer M, Deutschman CS, and Seymour CW. The third international consensus definitions for sepsis and septic shock (sepsis-3). *JAMA*, 2016.

[2] David W Shimabukuro, Christopher W Barton, Mitchell D Feldman, Samson J Mataraso, and Ritankar Das. Effect of a machine learning-based severe sepsis prediction algorithm on patient survival and hospital length of stay: a randomised clinical trial. 4(1), 2017.

[3] Papin G, Bailly S, Dupuis C, Ruckly S, Gainnier M, and Argaud L. A community approach to mortality prediction in sepsis via gene expression analysis. *Nat Commun*, 9(694), feb 2018.

[4] Calvert JS, Price DA, Chettipally UK, Barton CW, Feldman MD, Hoffman JL, Jay M, and Das R. A. A computational approach to early sepsis detection. *Comput Biol Med*, jul 2016.

[5] Mohammadreza Sheykhmousa, Masoud Mahdianpari, Hamid Ghanbari, Fariba Mohammadimanesh, Pedram Ghamisi, and Saeid Homayouni. Support vector machine versus random forest for remote sensing image classification: A meta-analysis and systematic review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 2020.

[6] Supreeth P. Shashikumar, Matthew D. Stanley, Ismail Sadiq, Qiao Li, Andre Holder, Gari D. Clifford, and Shamim Nemati. Early sepsis detection in critical care patients using multiscale blood pressure and heart rate dynamics. *Journal of Electrocardiology*, 50(6):739–743, 2017.

[7] Guy W. Glover DNatalie McLymont. Scoring systems for the characterization of sepsis and associated outcomes. *Annals of translational medicine*, 24(527), December 2016.

[8] Eren Gultepe, Hien Nguyen, Timothy Albertson, and Ilias Tagkopoulos. A bayesian network for early diagnosis of sepsis patients: a basis for a clinical decision support system. In *2012 IEEE 2nd International Conference on Computational Advances in Bio and medical Sciences (ICCABS)*, pages 1–5, 2012.

[9] Moncef Gabbouj Morteza Zabihi, Serkan Kiranyaz. Sepsis prediction in intensive care unit using ensemble of xgboost models. *Computing in Cardiology (CinC)*, 2019.

[10] Simon Meyer Lauritsen, Mads Ellersgaard Kalør, Emil Lund Kongsgaard, Katrine Meyer Lauritsen, Marianne Johansson Jørgensen, Jeppe Lange, and Bo Thiesson. Early detection of sepsis utilizing deep learning on electronic health record event sequences. *Artificial Intelligence in Medicine*, 104:101820, 2020.

[11] Joseph Futoma, Sanjay Hariharan, and Katherine Heller. Learning to detect sepsis with a multitask gaussian process RNN classifier. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1174–1182. PMLR, 06–11 Aug 2017.

[12] Ran Liu, Joseph L. Greenstein, Sridevi V. Sarma, and Raimond L. Winslow. Natural language processing of clinical notes for improved early prediction of septic shock in the icu. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 6103–6108, 2019.

[13] Desautels T, Calvert J, Hoffman J, Jay M, Kerem Y, Shieh L, Shimabukuro D, Chettipally U, Feldman MD, Barton C, Wales DJ, and Das R. Prediction of sepsis in the intensive care unit with minimal electronic health record data: A machine learning approach. *Comput Biol Med*, July 2016.

[14] Jacob Calvert, Thomas Desautels, Uli Chettipally, Christopher Barton, Jana Hoffman, Melissa Jay, Qingqing Mao, Hamid Mohamadlou, and Ritankar Das. High-performance detection and early prediction of septic shock for alcohol-use disorder patients. *Annals of Medicine and Surgery*, 8:50–55, 2016.

[15] Mohammed Saqib, Ying Sha, and May D. Wang. Early prediction of sepsis in emr records using traditional ml techniques and deep learning lstm networks. In *2018 40th*

*Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 4038–4041, 2018.

[16] Ranjit Lall. How multiple imputation makes a difference. *Political Analysis*, 24(4):414–433, 2017.

[17] Peter C. Austin, Ian R. White, Douglas S. Lee, and Stef van Buuren. Missing data in clinical research: A tutorial on multiple imputation. *Canadian Journal of Cardiology*, 37(9):1322–1331, 2021.

[18] Tzu-Tsung Wong. Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation. *Pattern Recognition*, 48(9):2839–2846, 2015.

[19] Pedro Ferreira, Duc C. Le, and Nur Zincir-Heywood. Exploring feature normalization and temporal information for machine learning based insider threat detection. In *2019 15th International Conference on Network and Service Management (CNSM)*, pages 1–7, 2019.

[20] V N Ganapathi Raju, K Prasanna Lakshmi, Vinod Mahesh Jain, Archana Kalidindi, and V Padma. Study the influence of normalization/transformation process on the accuracy of supervised classification. In *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pages 729–735, 2020.

[21] Moh Abdul Latief, Alhadi Bustamam, and Titin Siswantining. Performance evaluation xgboost in handling missing value on classification of hepatocellular carcinoma gene expression data. In *2020 4th International Conference on Informatics and Computational Sciences (ICICoS)*, pages 1–6, 2020.

[22] Wei Dong, Yimiao Huang, Barry Lehane, and Guowei Ma. Xgboost algorithm-based prediction of concrete electrical resistivity for structural health monitoring. *Automation in Construction*, 114:103155, 2020.

[23] Suryakanthi Tangirala. Evaluating the impact of gini index and information gain on classification using decision tree classifier algorithm*. *International Journal of Advanced Computer Science and Applications*, 11(2), 2020.

[24] Rui Sun, Guanyu Wang, Wenyu Zhang, Li-Ta Hsu, and Washington Y. Ochieng. A gradient boosting decision tree based gps signal reception classification algorithm. *Applied Soft Computing*, 86:105942, 2020.

[25] Minxing Si and Ke Du. Development of a predictive emissions model using a gradient boosting machine learning method. *Environmental Technology Innovation*, 20:101028, 2020.

[26] Leo Guelman. Gradient boosting trees for auto insurance loss cost modeling and prediction. *Expert Systems with Applications*, 39(3):3659–3667, 2012.

[27] Iwan Syarif, Adam Prügel-Bennett, and Gary B. Wills. Svm parameter optimization using grid search and genetic algorithm to improve classification performance. *TELKOMNIKA Telecommunication Computing Electronics and Control*, 14:1502–1509, 2016.

[28] Petro Liashchynskyi and Pavlo Liashchynskyi. Grid search, random search, genetic algorithm: A big comparison for NAS. *CoRR*, abs/1912.06059, 2019.

[29] Papin G, Bailly S, Dupuis C, Ruckly S, Gainnier M, and Argaud L. Clinical and biological clusters of sepsis patients using hierarchical clustering. *PLOS ONE*, 2021.

[30] Mesut Toğaçar, Burhan Ergen, and Zafer Cömert. Classification of white blood cells using deep features obtained from convolutional neural network models based on the combination of feature selection methods. *Applied Soft Computing*, 97:106810, 2020.

[31] Ran Liu, Joseph L Greenstein, James C Fackler, Melania M Bembea, and Raimond L Winslow. Spectral clustering of risk score trajectories stratifies sepsis patients by clinical outcome and interventions received. 9:e58142, sep 2020.

[32] Ewout W. Steyerberg, Andrew J. Vickers2, Nancy R. Cook, Thomas Gerds, Mithat Gonen, Nancy Obuchowski, Michael J. Pencina, and Michael W. Kattan. Assessing the performance of prediction models: a framework for some traditional and novel measures. *Epidemiology*, 21, jan 2010.

[33] Salvador García, Sergio Ramírez-Gallego, Julián Luengo, José Manuel Benítez, and Francisco Herrera. Big data preprocessing: methods and prospects. *Big Data Analytics*, 1(9), nov 2016.