

**DOKUZ EYLÜL UNIVERSITY**  
**GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES**

**EXPERIMENTING WITH SOME DATA MINING  
TECHNIQUES TO ESTABLISH PEDIATRIC  
REFERENCE INTERVALS FOR CLINICAL  
LABORATORY TESTS**

by

**Deniz ERASLAN**

September, 2019

İZMİR

**EXPERIMENTING WITH SOME DATA MINING  
TECHNIQUES TO ESTABLISH PEDIATRIC  
REFERENCE INTERVALS FOR CLINICAL  
LABORATORY TESTS**

**A Thesis Submitted to the  
Graduate School of Natural and Applied Sciences of Dokuz Eylül University In  
Partial Fulfillment of the Requirements for the Degree of Master of Science in  
Computer Engineering, Computer Engineering Program**

**by**

**Deniz ERASLAN**

**September, 2019**

**İZMİR**

**M.Sc. THESIS EXAMINATION RESULT FORM**

We have read the thesis entitled “EXPERIMENTING WITH SOME DATA MINING TECHNIQUES TO ESTABLISH PEDIATRIC REFERENCE INTERVALS FOR CLINICAL LABORATORY TESTS” completed by DENİZ ERASLAN under supervision of PROF. DR. SÜLEYMAN SEVİNÇ and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.



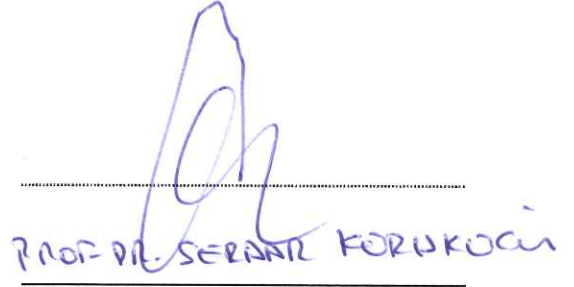
Prof. Dr. Süleyman SEVİNÇ

Supervisor



Asst. Prof. Dr. Özlem AKTAŞ

(Jury Member)



(Jury Member)



Prof. Dr. Kadriye ERTEKİN

Director

Graduate School of Natural and Applied Sciences

## ACKNOWLEDGMENTS

I would like to thank my advisor Prof. Dr. Süleyman Sevinç for his guidance. He was very helpful, patient and positive throughout the process. Also I would like to thank all of my friends and family for their support. They supported me in every aspect and motivated me to do better.

This project is achieved with sponsorship of Labenko Bilisim Inc. and all rights of the system belongs to Labenko Bilisim Inc.

Deniz ERASLAN

# **EXPERIMENTING WITH SOME DATA MINING TECHNIQUES TO ESTABLISH PEDIATRIC REFERENCE INTERVALS FOR CLINICAL LABORATORY TESTS**

## **ABSTRACT**

Aim of this thesis is to conduct some experiments on different datasets by using some data mining techniques under the multi-disciplinary study of establishing pediatric reference intervals from laboratory test results. While traditional methods can be applied for adults, it is difficult and inconvenient to apply these rules for pediatric patient groups. In order to resolve this need, this study has been performed to make the reference interval establishment process easier with data mining techniques.

To achieve this goal, a tool has been developed to allow specialists to easily load laboratory test data and to perform rapid data mining operations. With the help of this tool, specialists will be able to apply data mining techniques such as filtering, examine the distribution of parameters age-, gender-, and diagnosis-related data and apply machine learning algorithms in their reference interval studies based on hospital or device and publish it after reviewing their clinical accuracy. Pediatric reference intervals established by The Canadian Laboratory Initiative on Pediatric Reference Intervals (CALIPER) are used for the experiments on developed tool and the results were compared with the reference intervals published by CALIPER (Colantonio et al., 2012). As a result, by using data mining techniques via developed tool instead of classical statistical methods through the process of establishing reference intervals from laboratory test data, we can obtain faster and more effective results.

**Keywords:** Pediatric reference intervals, data mining, machine learning

# TIBBİ LABORATUVAR TESTLERİNİN PEDİYATRİK REFERANS ARALIKLARININ BELİRLENMESİ İÇİN BAZI VERİ MADENCİLİĞİ TEKNİKLERİ İLE DENEYLER

## ÖZ

Tezin amacı, bir multi-disipliner çalışma olan pediatrik referans aralıklarının laboratuvar test sonuç verilerinden hesaplanması projesi genel şemsiyesi altında bazı veri madenciliği teknikleri kullanarak farklı veri setleri üzerinde deneyler yapmaktır. Geleneksel yöntemler yetişkinler için uygulanabilirse de, bu kuralları çocuk hasta gruplarına uygulamak zor ve zahmetlidir. Bu ihtiyacı gidermek için bu çalışma, veri madenciliği teknikleri ile referans aralığı oluşturma işlemini kolaylaştırmak amacıyla geliştirilmiştir.

Bu amacı gerçekleştirebilmek için, uzmanların laboratuvar test verilerini kolayca yüklemelerine ve hızlı veri madenciliği işlemleri gerçekleştirmelerine olanak tanıyan bir araç geliştirilmiştir. Bu araç sayesinde uzmanlar filtreleme, yaş, cinsiyet ve tanıya ilişkin verilerin dağılımını inceleme ve makine öğrenmesi algoritmaları uygulama gibi veri madenciliği tekniklerini uygulayarak hastane veya cihaza dayalı çalışmalar yapıp klinik doğruluğunu gözden geçirdikten sonra kabul edilebilir bir referans aralığı olarak yayımlayabilecekler. The Canadian Laboratory Initiative on Pediatric Reference Intervals (CALIPER) tarafından belirlenen pediatrik referans aralıkları, geliştirilen araç ile yapılan deneyler için kullanılmış ve sonuçlar CALIPER tarafından yayınlanan referans aralıklarıyla karşılaştırılmıştır (Colantonio ve diğer., 2012). Sonuç olarak, laboratuvar test verilerinden referans aralıkları oluşturma sürecinde klasik istatistiksel yöntemler yerine geliştirilmiş araçlarla veri madenciliği teknikleri kullanılarak daha hızlı ve daha etkili sonuçlar elde edilebilir.

**Anahtar kelimeler:** Pediatrik referans aralığı, veri madenciliği, makine öğrenmesi

## CONTENTS

	<b>Page</b>
M.Sc. THESIS EXAMINATION RESULT FORM .....	ii
ACKNOWLEDGMENTS.....	iii
ABSTRACT .....	iv
ÖZ.....	v
LIST OF FIGURES .....	viii
LIST OF TABLES.....	x

## **CHAPTER ONE - INTRODUCTION ..... 1**

1.1 Background .....	2
1.1.1 Machine Learning .....	2
1.1.1.1 Supervised Learning .....	3
1.1.1.2 Unsupervised Learning.....	4
1.1.2 Data Mining.....	4
1.1.2.1 Data Selection .....	5
1.1.2.2 Preprocessing .....	6
1.1.2.3 Model Building-Selection.....	6
1.1.2.4 Application .....	6
1.1.2.5 Evaluation.....	7
1.1.3 Normal Distribution .....	7
1.1.4 Reference Interval.....	8
1.1.4.1 Classical Methods in Reference Interval Determination.....	10
1.1.4.1.1 Parametric Method .....	10
1.1.4.1.2 Non-parametric Method.....	11
1.2 Literature Review .....	12
1.3 Methodology and Tools .....	19

## **CHAPTER TWO - IMPLEMENTATION .....20**

2.1 Physical Model.....	20
2.2 User Interface .....	24

2.2.1 File Upload .....	25
2.2.2 Filter Data.....	28
2.2.2.1 Age Filter .....	30
2.2.2.2 Date Filter .....	31
2.2.2.3 Value Filter .....	31
2.2.2.4 List Filter .....	32
2.2.2.5 Text Filter .....	33
2.2.3 Statistical Analysis.....	35
2.2.4 Run Different Learning Algorithms .....	38
2.2.5 Learning Results .....	40
2.2.6 Edit Data.....	43
2.2.7 Export Data.....	44
<b>CHAPTER THREE – EXPERIMENTAL RESULTS .....</b>	<b>45</b>
<b>CHAPTER FOUR – CONCLUSION AND FUTURE WORK .....</b>	<b>52</b>
<b>REFERENCES .....</b>	<b>53</b>



## LIST OF FIGURES

	<b>Page</b>
Figure 1.1 Basic machine learning techniques .....	3
Figure 1.2 Steps of data mining .....	5
Figure 1.3 Normal distribution .....	8
Figure 1.4 Example distribution of test results in a healthy population .....	9
Figure 1.5 95% confidence interval .....	11
Figure 1.6 CALIPER study data-analysis algorithm based on CLSI guidelines.....	15
Figure 1.7 Robust statistical algorithm used by CALIPER to establish pediatric reference intervals using the direct method in accordance with CLSI .....	16
Figure 2.1 Physical model (applyLearning) .....	21
Figure 2.2 MongoDB collection .....	22
Figure 2.3 Physical model (showResults) .....	24
Figure 2.4 File upload .....	26
Figure 2.5 Sample xlsx file .....	26
Figure 2.6 Select columns, select data types and rename column header .....	28
Figure 2.7 Data filtering .....	29
Figure 2.8 Filter drop-down list.....	30
Figure 2.9 Age filter.....	30
Figure 2.10 Date-range picker .....	31
Figure 2.11 Value filter .....	32
Figure 2.12 List filter .....	33
Figure 2.13 Include/exclude toggle .....	33
Figure 2.14 Text filter example (for hospital unit) .....	34
Figure 2.15 Text filter example-2 (for gender) .....	34
Figure 2.16 Data after filtering .....	35
Figure 2.17 Gender distribution.....	36
Figure 2.18 Ordering unit distribution .....	36
Figure 2.19 Diagnosis distribution .....	36
Figure 2.20 Age distribution (year) .....	37
Figure 2.21 Age distribution (month) .....	37
Figure 2.22 Age distribution (week).....	37

Figure 2.23 Age distribution (day) .....	38
Figure 2.24 Select learning algorithm.....	38
Figure 2.25 Learning algorithm parameters .....	39
Figure 2.26 Learning result selection.....	40
Figure 2.27 Learning results mapping .....	41
Figure 2.28 Cluster chart.....	42
Figure 2.29 Cluster distribution (x-axis) .....	42
Figure 2.30 Cluster distribution (y-axis) .....	42
Figure 2.31 Cluster information .....	43
Figure 2.32 Edit data.....	44
Figure 2.33 Export data.....	44
Figure 3.1 GMM Calcium female 2 cluster .....	46
Figure 3.2 GMM Calcium female 2 cluster distribution.....	46
Figure 3.3 GMM Calcium male 2 cluster .....	47
Figure 3.4 GMM Calcium male 2 cluster distribution.....	47
Figure 3.5 GMM Albumin G female 5 cluster .....	48
Figure 3.6 GMM Albumin G female 5 cluster distribution .....	49
Figure 3.7 GMM Albumin G male 5 cluster .....	50
Figure 3.8 GMM Albumin G male 5 cluster distribution .....	50

## LIST OF TABLES

	<b>Page</b>
Table 3.1 CALIPER Calcium reference intervals .....	45
Table 3.2 GMM Calcium female 2 cluster values .....	46
Table 3.3 GMM Calcium male 2 cluster values .....	47
Table 3.4 CALIPER Albumin G reference intervals .....	48
Table 3.5 GMM Albumin G female 5 cluster values .....	49
Table 3.6 GMM Albumin G male 5 cluster values .....	50



## **CHAPTER ONE**

### **INTRODUCTION**

Reference intervals are taken as basis for clinical interpretation of biochemical laboratory tests and evaluation of the results. Despite its importance, reference intervals for pediatric patients remain inadequate or unavailable for many analytes (Colantonio et al., 2012). Moreover, most institutions use reference intervals of the manufacturers. Reference intervals differs among different populations, because of this reason, it is recommended that each institution should establish its own reference interval. Therefore, there is a need for detailing in the available reference interval.

Classical methods for establishing reference intervals are mainly based on statistical procedures. In many studies, it is assumed that laboratory tests form a normal distribution, and reference intervals are defined by threshold values between a specific percentage, commonly as 95% of healthy individuals. This means that 2.5% of individuals with the lowest results and 2.5% of individuals with the highest results will be excluded. By using machine learning algorithms, we aim to extract possible sub-groups that a normal distribution may contain. In this way, unhealthy individuals can be detected and eliminated.

In this study, as an alternative to classical methods, big amount of test results stored in hospital databases are processed using data mining techniques to establish reference intervals. We have developed a software tool to help solve this problem. With this tool, it is aimed to assist the experts by large variety of filtering and visualizing capabilities over large data stored in the hospital database and in the establishment of new reference intervals by applying data mining and machine learning techniques. It is also aimed to facilitate cleaning of defective data and splitting healthy and diseased groups in hospital database by filtering many features, such as diagnosis, gender and age. And there is need to make biochemical tests because of their health problems. By using common statistical processes to establish reference intervals requires long time and

big effort. Most studies include only a limited number of tests. It needs heavy workload to do a wide study.

We have to work separately for each test. The number of partitions required for age- and gender-stratified reference intervals can be quite numerous in the rapidly growing and changing neonatal and pediatric population (Adeli et al., 2017). Determining the reference intervals using conventional methods requires quite a long work. It is not even possible to determine the pediatric reference intervals due to these difficulties. It is quite difficult and costly to do a wide study. This is even harder on children. There are thousands of children coming to hospitals every day and due to their health problems. These test results accumulate in the hospital database, but they are not used for any purpose. Most of the data here belong to a healthy group, and they need to be separated and used. Not all of the tests conducted in other polyclinics are related to their illness. There are children coming to the hospital for regular check or follow-up (well child follow-up unit). Therefore, this data can be cleaned and healthy data can be extracted with the help of the tool mentioned in next chapter.

## **1.1 Background**

### ***1.1.1 Machine Learning***

Machine learning is the general name of computer algorithms that can learn structurally and make meaningful predictions on data. These algorithms mainly work by creating models from sample data. Today, it is not possible to process and analyze a large amount of data manually. Therefore, to solve a given problem, it is aimed to reach the solution by studying the data obtained from the environment of the problem by using machine learning algorithms. Machine Learning algorithms are basically divided into two groups, which are supervised learning and unsupervised learning. Figure 1.1 shows basic machine learning techniques.

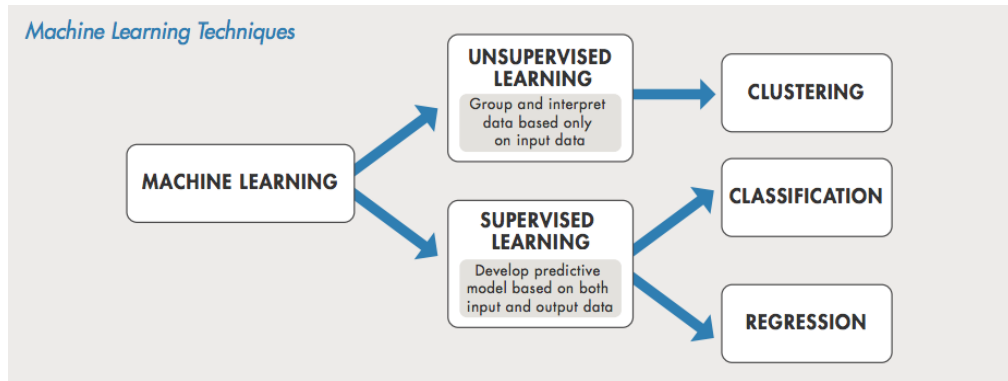


Figure 1.1 Basic machine learning techniques

### 1.1.1.1 Supervised Learning

In supervised learning, it is aimed to produce a function using training data between the inputs (labelled data) and the expected outputs. Training data consists of both inputs and outputs. In other words, data with known results are used in model development. In this way, it is aimed to estimate the results of data without label information in the data set based on the model created. Classification or regression algorithms can be used to determine this function. The most used supervised learning algorithms are:

- k-Nearest Neighbors (KNN)
- Artificial Neural Network (ANN)
- Support Vector Machine (SVM)
- Decision Trees (DTs)
- Linear Regression
- Logistic Regression

The worst part of supervised learning is to create training data. With training data, a function is generated by a machine learning method. The most time-consuming part

of supervised learning is the preparation of this training data. Poorly prepared training data will result with making inaccurate predictions.

#### *1.1.1.2 Unsupervised Learning*

Unsupervised learning is a machine learning technique that applies a function to predict an unknown structure on unlabelled data. In this case, the classes to which the inputs belong are uncertain. It is aimed to reveal hidden relationships or groups based on the components in the data set. The most used unsupervised learning algorithms can be listed as:

- Clustering
- Association Rules
- Principal Component Analysis (PCA)

There is no training data in non-supervised learning. The algorithms in this section try to assign the new data to the most appropriate group by grouping the data. It is relatively easy to apply since it has no training data. However, you may not get accurate results in difficult problems.

#### *1.1.2 Data Mining*

Data mining is the use of computer programs to determine the relationships and rules through big data that will help us to make predictions about the future. In this process, many different techniques can be used such as clustering, data classification, finding dependences, variability analysis and anomaly detection. With data mining, it is ensured that hidden information is stored in database systems consisting of big data. This is done by using statistics, mathematical disciplines, modelling techniques and

various computer programs. In recent years, data mining has focused on the information industry. The main reason for this interest is that a large amount of data can be obtained as a raw data and this obtained raw data should be converted into useful information as quickly as possible. Figure 1.2 shows the steps of data mining to accomplish this.

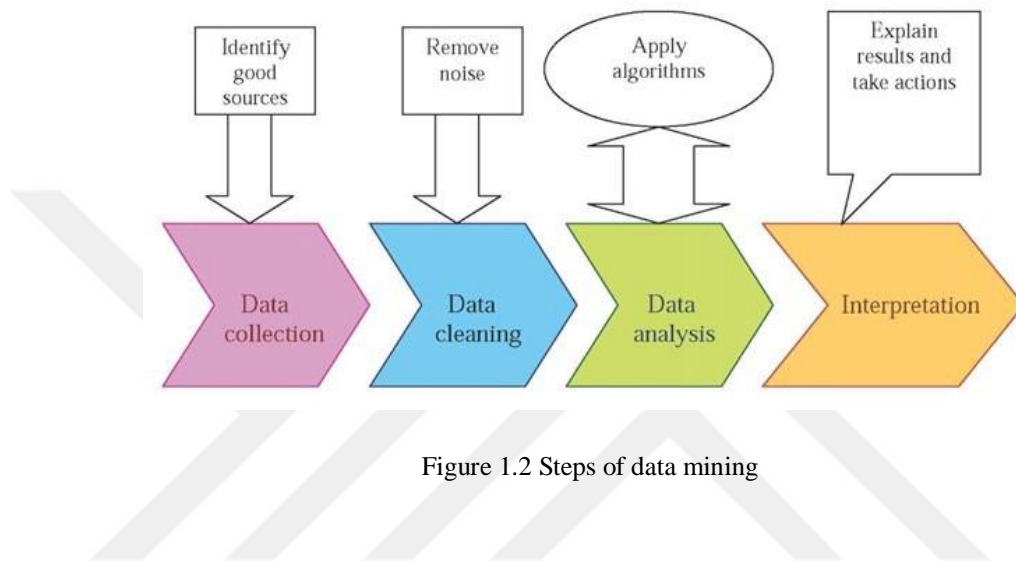


Figure 1.2 Steps of data mining

### 1.1.2.1 Data Selection

Data selection is one of the most time consuming step in data mining. In the first step, a variety of data from different sources are collected and combined. At this stage, the information generated in the information systems should be well analyzed and associated with the problem. The combination of big data into a single database or data warehouse is required for data mining application. In this step, we may not use all the data we have put together. Therefore, we need to extract data that may be useful for our data mining operations.



### *1.1.2.2 Preprocessing*

The preprocessing phase is important for the success of data mining. At this stage, the data is made available to be used in subsequent stages. It is possible to achieve precise and clear results with a successful preprocessing phase.

At this stage, it is possible to identify and eliminate unnecessary features by feature selection, which will make it troublesome to build our model. The data is required to be cleaned and missing data should be replaced with a new one. Missing data can be excluded from the data set, or a constant value can be used to fill missing values, such as the average.

### *1.1.2.3 Model Building-Selection*

This is the stage of thinking on various methods and choosing the one that is most suitable for our situation. Although it may appear to be only one operation, it may include detailed processes. Various techniques have been developed to accomplish this goal. Many of these methods are based on a technique that tries to model different models with the same data set and to perform their best to choose the best.

### *1.1.2.4 Application*

It is the stage where the data mining work is fully used. The data needs to be prepared before executing this stage. In this step, depending on the purpose of the study, one or more data mining techniques are applied on the data prepared according to the previous steps. These techniques are performed with the help of several algorithms that have their own characteristics.

### *1.1.2.5 Evaluation*

After data mining is applied on the data, the results are interpreted and the study is evaluated whether it has reached the correct result. If different methods are applied at this stage, their comparison is made. The results obtained are compared with the results of other studies. Generally, accuracy and easy applicability are the basis for the selection of methods for the study.

### *1.1.3 Normal Distribution*

Normal distribution is a hypothetical universe distribution. The normal distribution curve, also known as the 'Gaussian distribution' or 'Gaussian curve', is a continuous and probability function curve. The function of the normal distribution curve is as follows:

$$y = \frac{1}{\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma} \quad (1.1)$$

As seen in the formula, the parameters that define the normal distribution are mean ( $\mu$ ) and standard deviation ( $\sigma$ ). Normal distribution refers to the shape of a distribution in which the results are concentrated at the mean point and become infrequent at the endpoints. A sample normal distribution is shown in Figure 1.3.

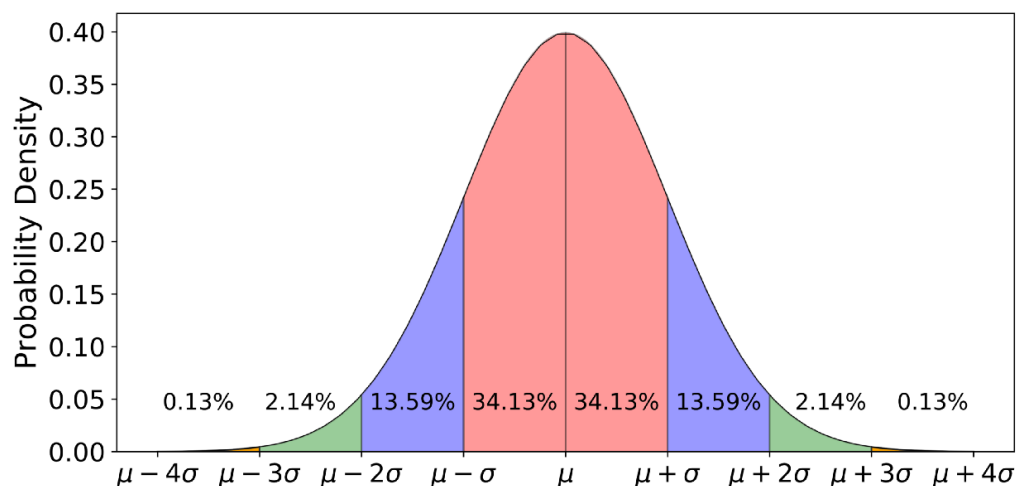


Figure 1.3 Normal distribution

Features of the normal distribution can be listed as follows:

- Symmetric
- Asymptotic
- Gets values between  $-\infty$  and  $+\infty$
- The probability of the total area under the curve is 1

#### ***1.1.4 Reference Interval***

The results obtained in clinical laboratories are interpreted by considering the reference interval values. In order for the clinicians to benefit from the results efficiently, the results should be presented in accurate and reliable reference intervals. Determination of reference intervals is made according to the recommendations of National Committee for Clinical Laboratory Standards (NCCLS) and International Federation of Clinical Chemistry (IFCC) (Sasse, 2000).

A reference interval determined by the manufacturer is available for each test. However, we do not know how accurate this interval reflects the reference values of the society we address. Therefore, we should determine the reference values of our

own population and use these values instead of reference values determined by the manufacturer. Baadenhuijsen and Smit (1985) suggest that the use of the hospital population in the selection of reference individuals will produce more reliable reference ranges than the use of healthy individuals.

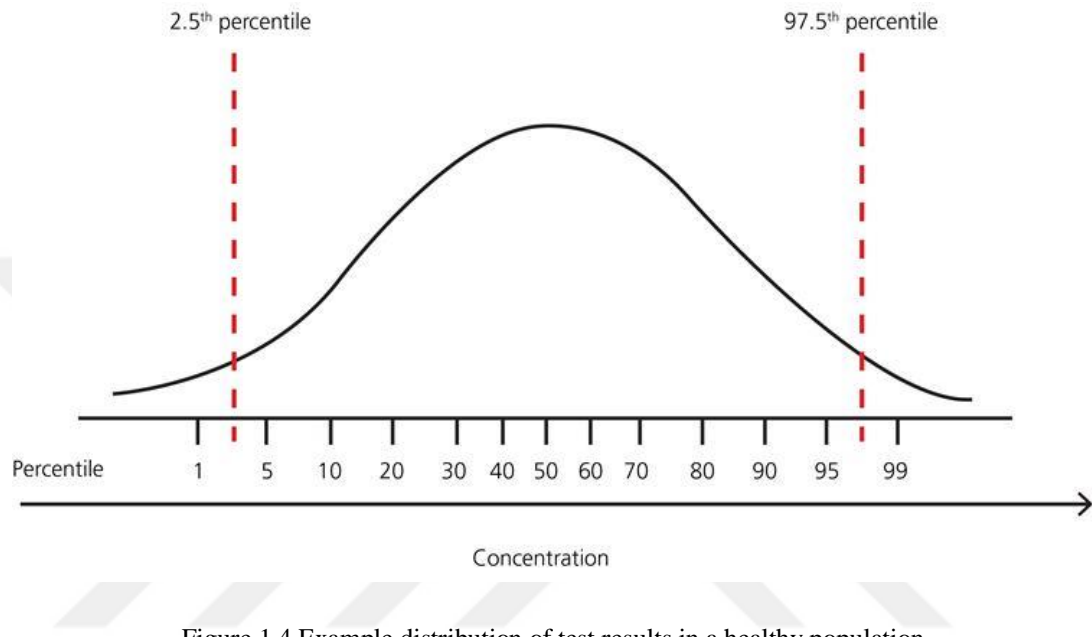


Figure 1.4 Example distribution of test results in a healthy population

A reference population may be formed by bringing together reference individuals selected from a sample population. The selection of reference individuals constitutes one of the most important steps in determining the reference interval. As shown in Figure 1.4, these values will form a distribution, and if this distribution is subjected to statistical analysis, the upper and lower values that limit a certain part of the distribution can be obtained. In this case, the area between the upper and lower values will represent a certain percentage of the distribution.

#### *1.1.4.1 Classical Methods in Reference Interval Determination*

We can classify the classical methods used in the reference interval determination as:

- Parametric methods
- Non-parametric methods

If our distribution is a normal distribution, we can apply parametric methods and get meaningful results even in low amounts of data. However, non-Gaussian distributions are observed frequently in biological data. This forces the usage of non-parametric methods instead of parametric methods in reference interval studies. A disadvantage of non-parametric methods is that they require higher number of data.

*1.1.4.1.1 Parametric Method.* Gaussian distributions are defined by certain parameters, such as mean ( $\mu$ ) and standard deviation ( $\sigma$ ). These parameters define the shape of the distribution. The parametric approach includes calculating the average and standard deviation in order to determine 95% confidence interval that the values fall into. The distribution must be a Gaussian distribution, otherwise it can be transformed to a Gaussian distribution by several methods (Linnet, 1987). In such distributions, parametric methods can be used to calculate the reference intervals.

In parametric method, the upper and lower values which constitute the boundaries of a region of 2.5% - 97.5%. This region corresponds to 95% of the distribution. The points forming the  $\pm 2$  SD range of the mean are defined as the reference range. Figure 1.5 shows an example 95% confidence interval calculated from a sample test.

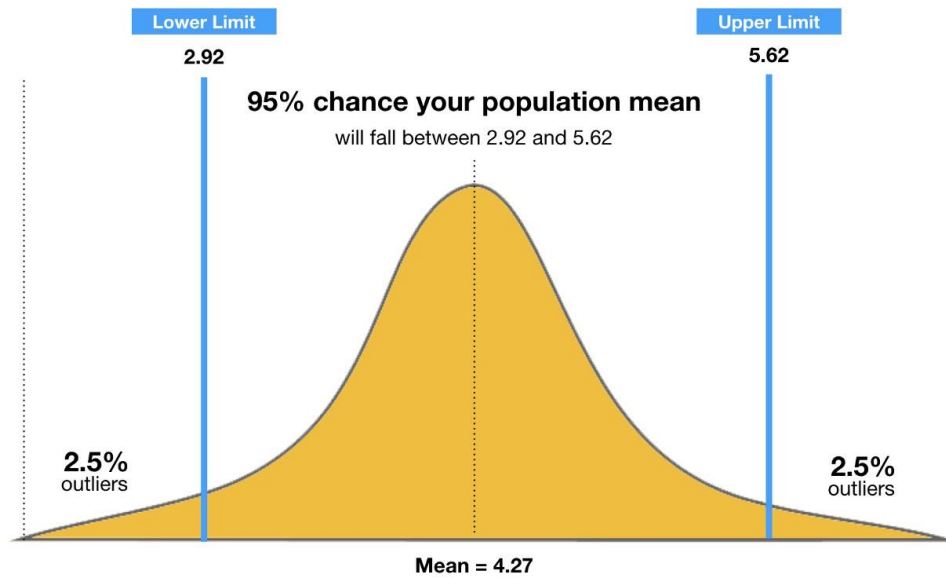


Figure 1.5 95% confidence interval

*1.1.4.1.2 Non-parametric Method.* With non-parametric method, we can handle non-Gaussian distributions. This makes the selection of the reference individuals easier and hospital databases can be used for reference interval studies. However, the number of data for non-parametric method to be used should be at least 120, while in the low number of data they are quite inadequate.

In non-parametric method we look for the values that correspond to 95% of the distribution, i.e. 2.5% to 97.5%, the following formulas are given below:

$$\text{Lower value} = 0.025 \times (n + 1) \quad (1.2)$$

$$\text{Top value} = 0.975 \times (n + 1) \quad (1.3)$$

‘n’ indicates the number of data. Firstly, the data is sorted in ascending order. Results in these formulas correspond to the index number of lower value and upper value of our reference interval. If fractional numbers appear, numbers are rounded to remove the fraction (Lumsden & Mullen, 1978).

## 1.2 Literature Review

Horn, Pesce and Copeland (1998) suggested a new technique for the estimation of reference intervals for sparse data sets with significant number of outliers. They introduced a prediction interval that uses solid placing and scale predictions. To make these calculations, the SAS program can be easily altered. In this study, different reference interval estimation procedures are compared, which are;

- Non-parametric
- Transformed
- Robust with a non-parametric lower limit
- Transformed robust

The reference intervals were predicted both with outlier detection and without outlier detection. In this situation, the robust approach continuously yielded upper reference interval values which are closer to those of the real underlying distributions. It is suggested in the study that robust statistical analysis from restricted or potentially inaccurate data can be of excellent use for the prediction of reference intervals. This proposed robust method by Horn, Pesce and Copeland (1998) should be used if the sample size is so narrow, in order to detect at least the upper part of the reference intervals.

A study by Jagarinec et al. (1998) aims to establish the reference intervals for 34 biochemical analytes by local school children aged between 8 and 18 from Zagreb, Croatia. Healthy children that are used in this study were chosen by specialists according to their medical examinations. A software named "Statistics" were used to analyse the data at the Faculty of Science, University of Zagreb, Croatia. The nominal variables, which characterized the features of children's population, were analyzed by sex- and age-specific frequency distributions. The reference intervals represented the 2.5 - 97.5 percentiles of non-parametric distributions. Results have been eliminated

that varied by more than 3 standard deviations. To estimate the differences between sex and age groups, Mann Whitney U-test was implemented. According to Jagarinec et al. (1998), the results obtained from this study, which focused on children aged between 8 and 18, did not show significant differences with similar studies carried out in USA, Canada, England, France and Spain.

Kapelari et al. (2008) mentioned the importance of age- and sex-specific reference intervals for interpreting thyroid hormone measurements in pediatric patients in their article. They also indicated that the work done on TSH, free T3 (fT3), and free T4 (fT4) is limited. Therefore, they decided to focus on this topic. For this purpose, thyroid hormone test results which are obtained from a hospital-based pediatric population classified by diagnostic categories using the International Classification of Diseases (ICD-10) codes. The reference group is cleaned from children with conditions that likely to affect thyroid function. Analysis also did not include patients with eating disorders, with pituitary disease or chromosomal anomalies. Two-tailed values of  $p < 0.05$  were regarded as statistically significant. Medians and percentiles were identified and held as the reference interval for each variable. The Social Sciences Statistical Package (SPSS) has been used for statistical analysis and to create percentiles. The results indicate that thyroid hormone levels change significantly during childhood. Therefore, applying adult reference intervals to children is not suitable for most cases. Furthermore, some differences were detected compared to previously established reference intervals and previous studies, which could be caused by distinct features of the antibody, different races or undefined geographical covariations.

Humberg et al. (2010) aimed to calculate reference intervals for main haematological and biochemical parameters for infants and children living in Gabon. The data is derived from healthy Gabonese children who have visited Albert Schweitzer Hospital in Lambaréné, Gabon for routine controls. The establishment of reference intervals was done using the approved guideline of the Clinical and Laboratory Standards Institute (Horowitz, 2010). They applied the non-parametric



method which uses the 2.5th and 97.5th percentile of the data to calculate intervals. Dixon test was applied for detecting outliers. The standard normal deviate test (ztest) was used to decide if the reference values should be divided into subgroups as males and females, which measures the degree of statistical difference between these subgroups. Microsoft Excel 2002 is used to store laboratory test values. jmp 7.0 statistical software was used for calculations. The resulting reference intervals were compared with reference intervals belonging to a European population. As a result, values for haemoglobin, haematocrit, red blood cell count, mean corpuscular volume were found lower and platelet counts were higher than the comparative results. The research proves the significance of establishing reference intervals for local communities. This study also indicates that established reference intervals can be used for similar populations in Central Africa.

Colantonio et al. (2012) indicate in their article that we don't have enough knowledge about how age, sex, and ethnicity effect reference interval values. Based on this idea, a reference interval database needs to be built in order to fill this gap. To this end, healthy children and adolescents were chosen from a multiethnic population and evaluated according to specified exclusion criteria from conducted questionnaires. Whole-blood samples were gathered for 40 serum biochemical markers to establish age and sex-specific reference ranges. Analysis were done over the data according to the Clinical and Laboratory Standards Institute (CLSI) guidelines, as outlined in Fig.1.6 (Horowitz, 2010). Statistical analysis was performed using Microsoft Excel and SPSS softwares.

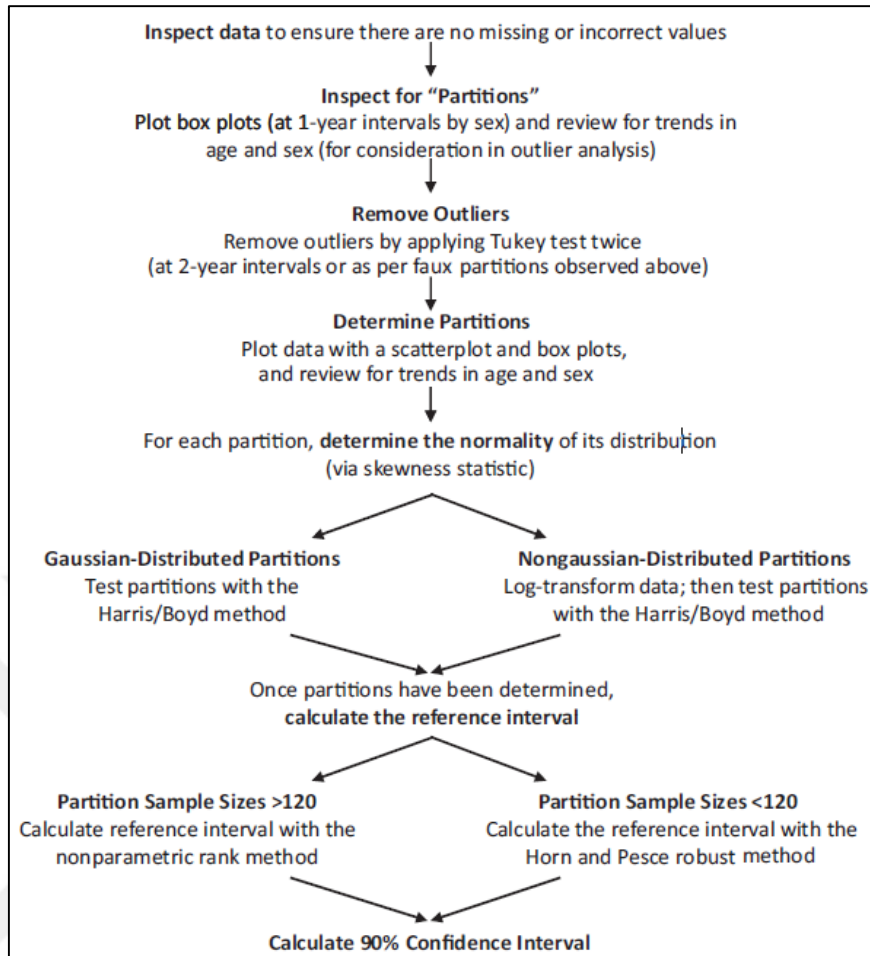


Figure 1.6 CALIPER study data-analysis algorithm based on CLSI guidelines (Horowitz, 2010)

Adeli et al. (2017) states that in pediatric laboratory medicine field, several clinical laboratories are obliged to report adult reference intervals for pediatric test data, since there is a lack of age- and sex-specific pediatric reference intervals determined by healthy children and adolescents. For this purpose, the Canadian Laboratory Initiative on Pediatric Reference Intervals (CALIPER) started to recruit healthy children and adolescents from the community and developed an algorithm showed in Figure 1.7 to develop accurate age- and sex-specific pediatric reference intervals regarding the Clinical and Laboratory Standards Institute (CLSI) guidelines (Horowitz, 2010). Nowadays CALIPER reference interval database is used by clinical laboratories across Canada and globally.

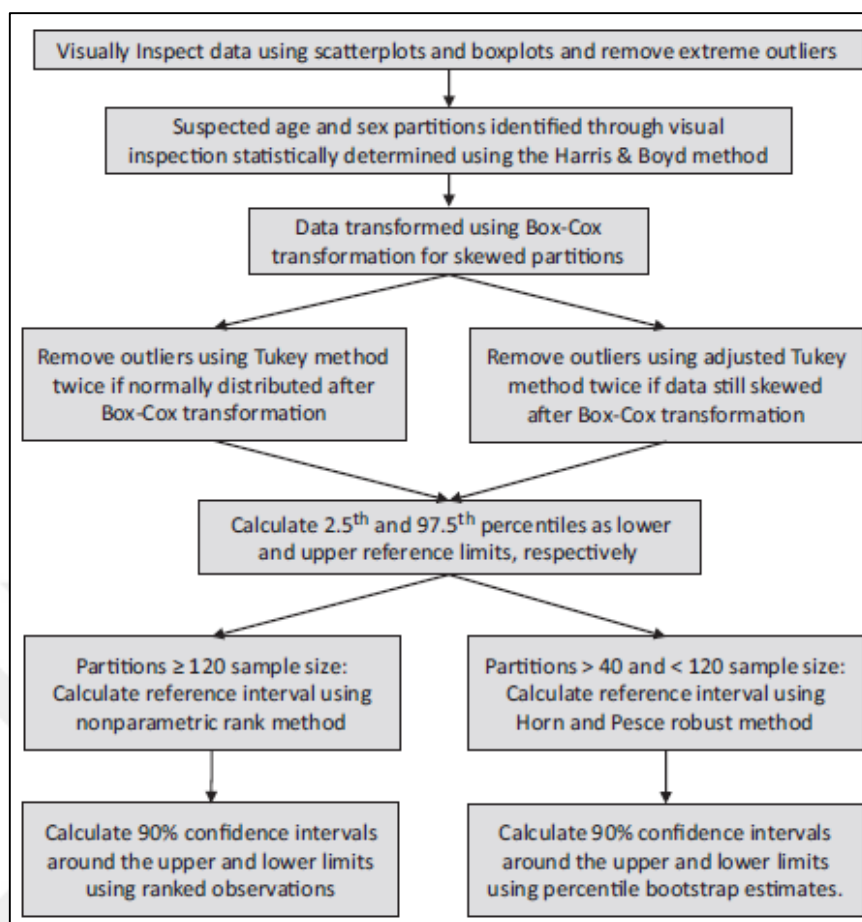


Figure 1.7 Robust statistical algorithm used by CALIPER to establish pediatric reference intervals using the direct method in accordance with CLSI (Horowitz, 2010)

Katayev, Balciza and Seccombe (2010) collected data from a nationwide chain of clinical laboratories in the United States for 5 analytes without any cleaning, exclusion or processing operation. The Hoffmann indirect method has been programmed to derive reference intervals. For the identification and elimination of outliers, Chauvenet criteria were used. After removing the outliers, the data was refined to use only values from the linear portion of the cumulative frequency graph. The computer-controlled Hoffmann method produced reference intervals which were quite similar to peer-reviewed reference intervals. There were no statistically significant variations in the calculated upper or lower limits from the respective published limits. In most cases, the calculated ranges were slightly narrower compared to the ranges obtained from direct sampling methods. The technique outlined to indirectly estimate the reference

intervals from the data obtained from laboratory database has proven to be accurate and admissible.

Another study by Katayev et al. (2015), for the establishment of clinical laboratory test reference intervals, an application is presented which uses a data mining statistical algorithm. In this study, 8 analytes stored in the laboratory database were used for reference intervals calculations with different age- and sex-specific groups by using the altered algorithm for data mining. The laboratory test results were gathered from the database of the Laboratory Corporation of America for the parameters age, sex, and timeframe. The analytes and age- and sex-specific groups were selected by the indirect data-mining and direct sampling techniques for the reference interval calculations. Age- and sex-specific partitioning was carried out as outlined in the previous studies. The computerized statistical algorithm of Hoffmann has been updated with a couple of new functionalities and improvements from the earlier released prototype. Afterwards, these outcomes have been compared to peer-reviewed studies released using direct sampling. Most of presented reference intervals in this study do not show any statistically significant difference from previously defined reference intervals. It is indicated that the presented computerized statistical algorithm which is based on Hoffmann's approach for reference interval calculations is a precise and practical tool for calculating reference intervals.

Akbayir et al. (2011) evaluated homocysteine, vitamin A and vitamin E results from Mersin University Hospital of Faculty of Medicine, Clinical Biochemistry Laboratory and established new age- and sex-specific reference intervals for Mersin region. Initially, homocysteine, vitamin A and vitamin E levels were split into the subclass of gender and descriptive statistics were drawn up for the ages of each variable. Multivariate Adaptive Regression Splines (MARS) method was used to evaluate age subgroups. Depending on the distribution, traditional parametric and non-parametric reference interval methods were preferred. MedCalc package program is then used to calculate reference intervals.

Author also points out that (Akbayir et al., 2011) each laboratory must find and apply reference values for its own community. For each laboratory, when considering aspects such as complexity and cost, it is advantageous to use indirect methods in calculating reference intervals. Therefore, in this study, the indirect method was used to determine the reference intervals according to age and sex. According to this study, homocysteine and vitamin A levels should be evaluated according to age and gender, where vitamin E levels should only be evaluated according to gender. These findings underline the significance of using suitable reference ranges for each laboratory from its population. However, it is difficult, time-consuming and costly to calculate the reference range for each laboratory following these steps.

Orekici Temel et al. (2015) used Bhattacharya procedure in their study which provide a healthy population selection by using hospital data to calculate the reference interval. Healthy and diseased groups mix while using the indirect method. Calculating the reference intervals by this data can be a problem. Using the Bhattacharya procedure, it is possible to obtain a healthy subgroup from hospital data. A graphical model was proposed by Bhattacharya to detect overlapping normal distributions. However, there are some disadvantages with this method. The reference range established with the Bhattacharya procedure is extremely dependent on data distribution. In order to get healthy results, the data should be largely suitable for normal distribution.

Çaycı et al. (2015) aimed to determine whole blood reference intervals from results stored in hospital database for the hemogram test. They used non-parametric percent estimation method on laboratory test data and compared results with existing manufacturers reference intervals. The SPSS 15.0 (Statistical Packages for Social Sciences) program was used in the statistical evaluation of the data. This study concentrated specifically on people under 12 years and over 65 years. Reference intervals were calculated separately for age and sex subgroups. Results indicated that reference intervals should be calculated separately for all sub-parameters in both genders and age groups between 0 and 12. Authors mentioned that reference intervals

calculated from corresponding hospital population by indirect method are more suitable for each laboratory since there are significant differences between the manufacturer recommended reference intervals and calculated reference intervals in this study.

### **1.3 Methodology and Tools**

In this study, HTML5, CSS3 and JavaScript programming languages are used to develop client side, where Node.js v6.11.4 is used for server side because of its functionalities like productivity, high speeds, and scalability. Brackets and Notepad++ are selected as editing tools. Learning algorithms run on Python 3.6.3. Experiments were done on a machine with Microsoft Windows 10 v1803 operating system. Google Chrome was selected as a testing environment.

MongoDB is selected as database since it provides high performance, high availability, and a dynamic structure. MongoDB Community Server 4.0.3 and Robomongo Robo 3T 1.2.1 is used for this purpose in this study.

## **CHAPTER TWO IMPLEMENTATION**

Data obtained from the hospital information system is not suitable for direct use. The anomalies inside the data should be cleaned before being used for reference range studies. This cleaning process may need to be repeated several times for each test. Therefore, there is a need for a software that can easily handle these cleaning processes. With the software we developed, filtering operations on big data can be handled in a very short time.

Laboratory test results obtained from the hospital database should be subjected to a statistically detailed examination before performing the reference interval study. Having an idea about the characteristics of hospital data will have a positive effect on the results of our studies. Therefore, an easy and detailed implementation of statistical analysis is necessary for different parameters of test results, such as age and gender.

It is necessary to carry out many experiments on the preprocessed and analyzed data. Determining the most suitable learning algorithm and parameters for the test results requires many experiments. This software has been developed to perform these operations quickly and efficiently.

### **2.1 Physical Model**

After preprocessing operations, a selected machine learning algorithm is applied on filtered data through `applyLearning()` function. Physical model of this process can be seen in Figure 2.1.

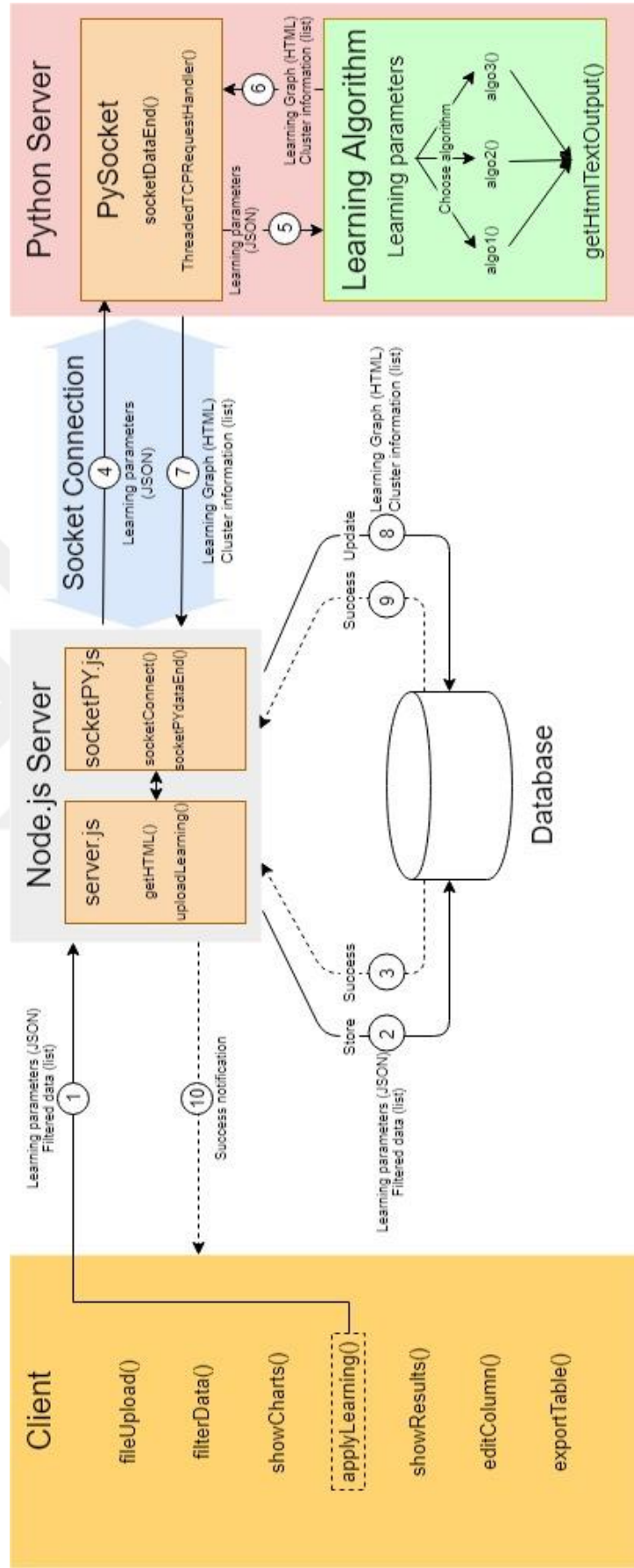


Figure 2.1 Physical model (applyLearning)



In the first step, specified learning parameters and filtered data is passed to node.js server via an AJAX post operation. After this step, uploadLearning() function in Node.js server establishes a MongoDB connection to store these data in the database. An example of MongoDB collection shown in Figure 2.2 displays four different learning results. Since the maximum BSON document size is 16 megabytes and filtered data often exceeds this limit, data is stored as 50,000 element chunks. Whenever the data exceeds this number, a new document is created to store the rest of the filtered data with a following order. “order” property in the database collection is used to keep the order while combining filtered data, since they must be updated and retrieved in the same order later. ”timestamp” is used for distinguishing studies from each other and keeping the date and time of the study to display them to the user while loading learning results.

_id	timestamp	cluster	xParam	yParam	xParamTitle	yParamTitle	thresholdtitle	threshold	data	_comment	props	html	mean_colors	means	labels	order	filtered_data
1	Objectid... 1557242184...	2	AgelInHour	iron	Yag	Demir		null	[ 869 eleme...	demir_2clust...	( 9 fields )		[ 2 eleme...	[ 2 elem...	[ 869 elements ]		
2	Objectid... 1557242184...	2	AgelInHour	iron	Yag	Demir		null	[ 869 eleme...	demir_2clust...	( 9 fields )		[ 2 eleme...	[ 2 elem...	[ 869 elements ]	0	[ 869 elemen...
3	Objectid... 1557242338...	2	AgelInYear	Result	Yil	Sonuc Dejeri		null	[ 869 eleme...	demir_2	( 8 fields )		[ 2 eleme...	[ 2 elem...	[ 869 elements ]	0	[ 869 elemen...
4	Objectid... 1557242338...	2	AgelInYear	Result	Yil	Sonuc Dejeri		null	[ 869 eleme...	demir_2	( 8 fields )		[ 2 eleme...	[ 2 elem...	[ 869 elements ]	0	[ 869 elemen...
5	Objectid... 1558097261...	5	AgelInWeek	sonuc	Hafta	Sonuc Dejeri		null	[ 2524 elem...	iga_Scluster...	( 13 fields )		[ 5 eleme...	[ 5 elem...	[ 2524 elemen...	0	[ 2524 eleme...
6	Objectid... 1558097261...	5	AgelInWeek	sonuc	Hafta	Sonuc Dejeri		null	[ 2524 elem...	iga_Scluster...	( 13 fields )		[ 5 eleme...	[ 5 elem...	[ 2524 elemen...	0	[ 2524 eleme...
7	Objectid... 1558098496...	7	AgelInWeek	sonuc	Hafta	Sonuc Dejeri		null	[ 149999 ele...	hgb_7cluster	( 13 fields )		[ 7 eleme...	[ 7 elem...	[ 149999 elem...	0	[ 149999 elem...
8	Objectid... 1558098496...	7	AgelInWeek	sonuc	Hafta	Sonuc Dejeri		null	[ 149999 ele...	hgb_7cluster	( 13 fields )		[ 7 eleme...	[ 7 elem...	[ 149999 elem...	0	[ 149999 elem...
9	Objectid... 1558098496...															1	[ 50000 elem...
10	Objectid... 1558098496...															2	[ 49999 elem...

Figure 2.2 MongoDB collection

After storing filtered data and learning parameters in the database, learning parameters are passed to the socket connection via fnPYsocketConnect() function. Python side mainly aims to apply a machine learning algorithm to a set of data. On python server, PySocket.py file requests the learning parameters via socketDataEnd() function and parse the requesting json data. From “algo” parameter specified in

learning parameters, desired learning algorithm is identified. Currently we can only work with the “GaussianMixture” algorithm of “sklearn”, but new algorithms are planned to be added in the future. Based on the learning parameters, learning method is applied to data and `file_html()` function is used from “bokeh.embed” library to get output of HTML bokeh graph as a string value. Besides the HTML string, information about the clusters that are organized as a json file is also passed to `PySocket.js` with `getHtmlTextOutput()` function.

Node.js server receives responded json data (HTML string and cluster information) via `socketPYDataEnd()` function in `socketPY.js` file. This function provides the MongoDB connection after receiving data and updates the corresponding document. The function checks “timestamp” parameters to decide which document should be updated. An updated MongoDB collection with HTML result, means, `mean_colors` and cluster labels is shown in Figure 2.2. Our filtered data stored in the database is updated with cluster labels stored in “labels” parameter of response json data, which means “cluster” parameter is added to each data point that we sent to learning.

User can access previous learning results stored in the database at any time from user interface via `showResults()` function. Physical model of this process can be seen in Figure 2.3.

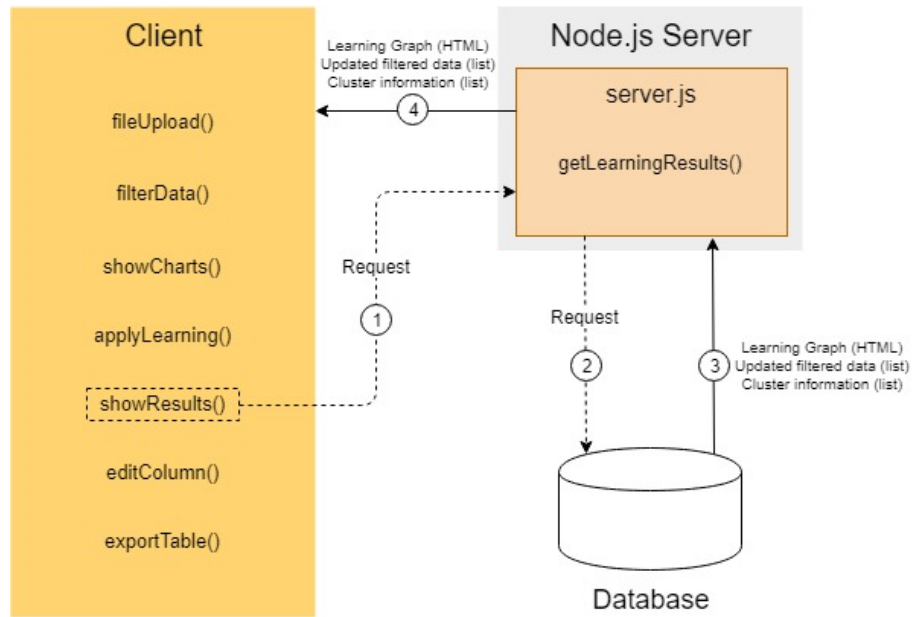


Figure 2.3 Physical model (showResults)

An AJAX request is sent to Node.js server and `getLearningResults()` function establishes a MongoDB connection to retrieve HTML string, updated filtered data and cluster information. Cluster information consists of means and mean\_colors. Client side gets these information as a response from Node.js server and updates the user interface (data table, filters, graphs) according to the response data.

## 2.2 User Interface

The purpose of this software is to provide an environment where specialists can easily perform reference interval studies. Therefore, a simple and easy to use interface was preferred in the process of development. User interface consists of a single homepage. All transactions are executed through this homepage. At the top of the page you will find a navigation bar created using the Navbar library of Bootstrap. Most operations such as uploading a file, sending data for machine learning, displaying learning results and exporting data are carried out via this navigation bar. On the left side, our filtering panel is displayed. Through this panel, we can apply various filtering

processes on our uploaded data. The rest of the page contains the data table with multiple columns. The data that we have uploaded is displayed in this table.

Patient data is kept as different data models in the databases used by each hospital. Therefore, the data exported from hospital database should be loaded into the software in .xls or .xlsx extensions. Patient data must be preprocessed before being sent to the learning algorithm. Statistical analysis is required to learn the characteristics of the data. The prepared data is sent to different learning algorithms with the determined parameters. The results returned from the learning algorithm are stored in the database and can be displayed at any time. The filtered data can then be exported and stored for further use.

### ***2.2.1 File Upload***

We can select a file from our local storage by clicking to “Dosya Yükle” tab as shown in Figure 2.4 to load the content of the file to our user interface. Only files with extensions .xlsx and .xls are allowed to load. A sample Microsoft Excel file and its content is shown in Figure 2.5. Javascript library xlsExport.js has been used to retrieve data from Excel file. It converts each row in the Excel file to a javascript array to make it available for use. There must be no filters or test data in the first line of the Excel file to be opened.

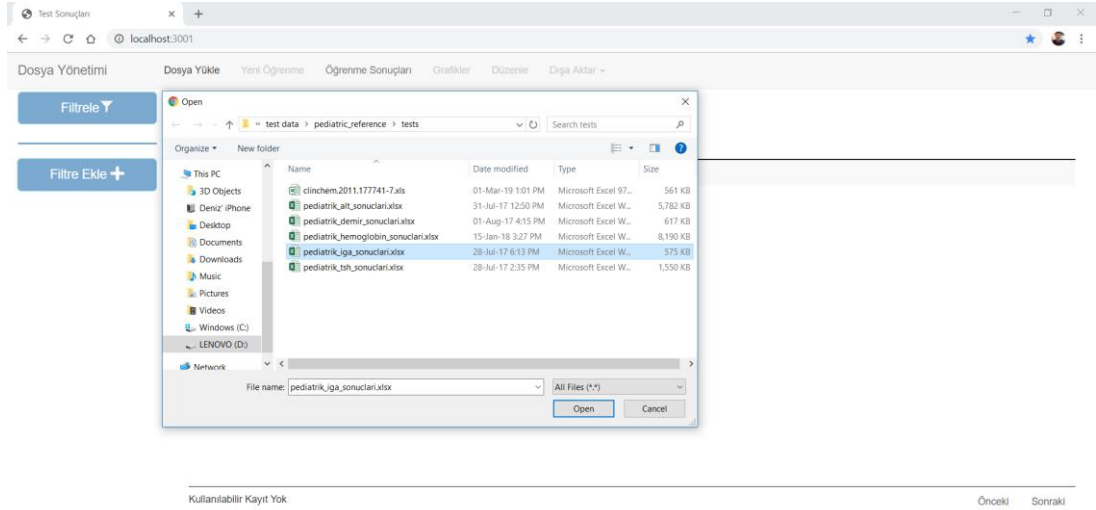


Figure 2.4 File upload

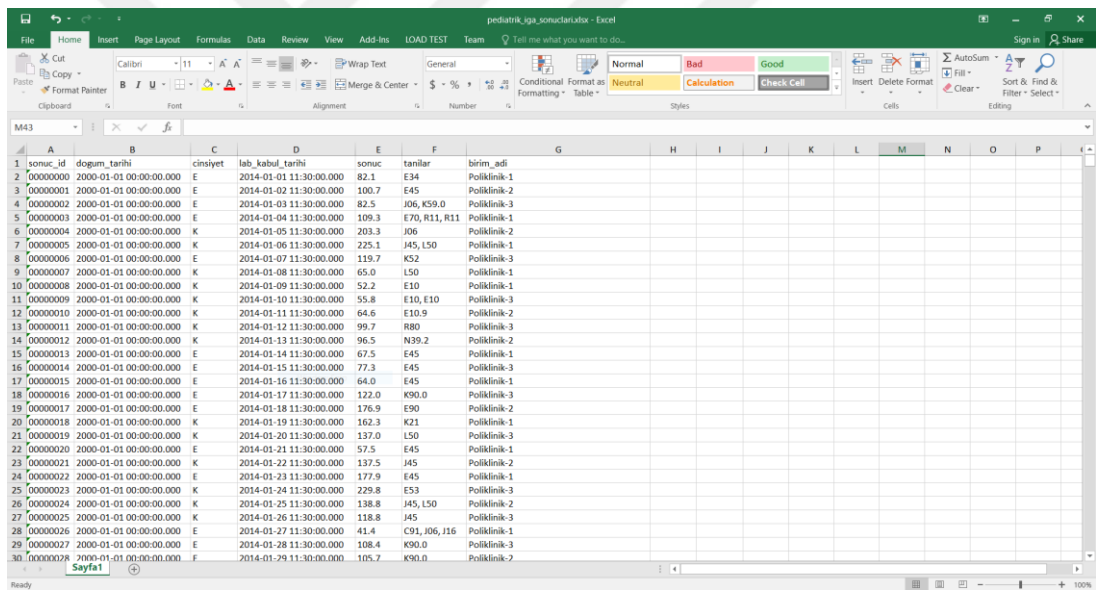


Figure 2.5 Sample xlsx file

After selecting and loading a proper Excel file to our user interface, a column select modal will show up as shown in Figure 2.6 to make us pick which columns we are going to use for further operations. In this modal, columns that we pick will be included in the table and special filters will be created for each column we selected depending on their data types. We can also rename each column header to display more meaningful texts on the table and filters which will prevent confusions that may

occur in the following processes. These information displayed in the modal shown in Figure 2.6 were created dynamically from the first row of the Excel file that we uploaded. We should also pick the suitable data type for each column from the dropdown list. This will let us to create suitable filters for each column, and display data in the correct form. Data types consists of;

- “Nümerik”: Data will displayed in the table. However, no filter will be created for this column.
- “Yaş”: If the age information is given directly in the file (if we do not have to calculate it via date of birth and date of measurement), this data type should be selected.
- “Doğum Tarihi”: If the age information is not provided directly in the file, date of birth must be selected in order to calculate the age.
- “Ölçüm Tarihi”: If the age information is not provided directly in the file, date of measurement must be selected in order to calculate the age.
- “Değer”: For float values.
- “Metin”: For plain text.
- “Liste”: If the cell contains multiple elements (it can be text, float or integer), and filtering will be applied seperately to each of these elements.

By selecting appropriate data types, user decides which type of data is going to be displayed in the table, and which type of filters are going to be used in data elimination step. We picked related data types for each column as shown in Figure 2.6 and replaced their headers with meaningful texts to avoid any potential confusions.

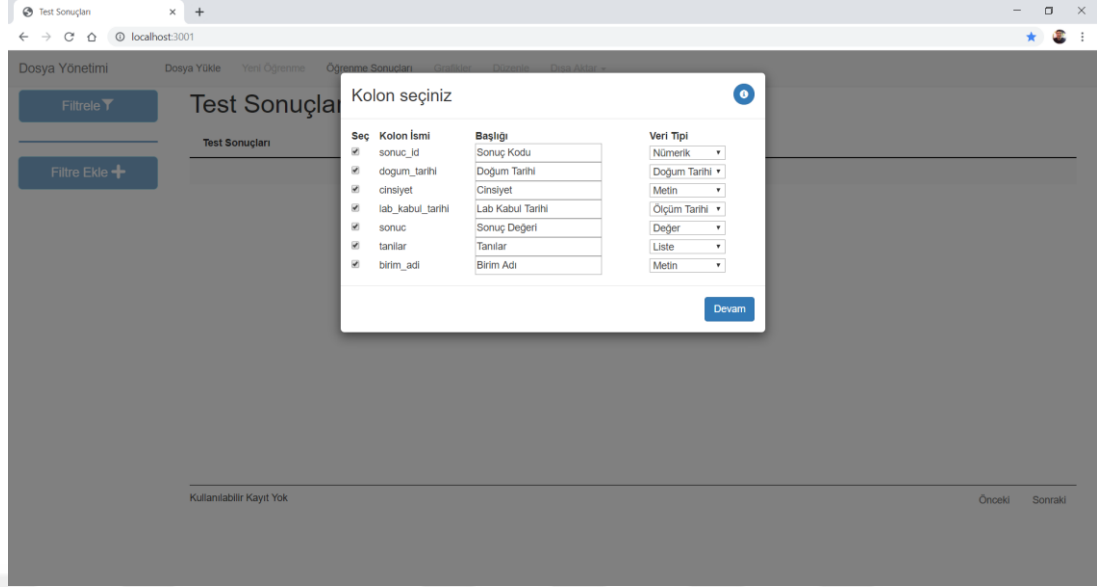


Figure 2.6 Select columns, select data types and rename column header

### 2.2.2 Filter Data

After determining the properties of the columns, the contents of the selected Excel file are processed into the table on our page as displayed in Figure 2.7. The title of the table gives information about uploaded test name and the total number of data found in the table. Each page of the table contains a total of 500 data. If the number of columns selected exceeds the width of the table, the remaining data can be displayed with the button next to the table rows. The table shows age parameter in years, months, weeks and days separately. In this way, it is possible to apply filtering in different units to the age parameter. Javascript library named “Datatables.js” is used to do these operations.

A filter container is included in the left side of the screen as shown in Figure 2.7. By default, a single filter named as “Filtre 1” is automatically added to the filter container as soon as data is loaded into the table. Users can add new filters by using "Filtre Ekle" button according to their plans. Added filters can be easily removed later if needed. Contents of each filter are filled dynamically depending on the data loaded

with the Excel file. Types of each of these filters are arranged by selected data types in column select modal shown. Bootstrap’s “accordion” library is used to add dropdown mechanism for filters.

Each of these filters inside this container (Filtre 1, Filtre 2 etc.) acts as “AND” operators in itself. However, each distinct filter works as an “OR” operator between each other. Which means, filtered data from different filters, such as “Filtre 1” and “Filtre 2” are combined together before updating the data table.

The screenshot shows a web application interface for managing test results. The main heading is "IgA Test Sonuçları - Sonuç Sayısı: 8071". The interface includes a sidebar with filter controls: "Filtrele" (dropdown), "Filtre 1", and "Filtre Ekle +" (plus icon). The main table displays the following data:

Sonuç Kodu	Cinsiyet	Yıl	Ay	Hafta	Gün	Lab Kabul Tarihi	Sonuç Değeri	Birim Adı	Tanıtar
00000000	E	10	128	557	3904	01/01/2014 11:30:00 AM	131.4	Poiklinik-1	R53
00000001	K	4	56	243	1707	02/01/2014 11:30:00 AM	68.3	Poiklinik-2	J45,L50
00000002	E	3	39	171	1202	03/01/2014 11:30:00 AM	83.9	Poiklinik-1	J45,L50
00000003	E	7	95	415	2913	04/01/2014 11:30:00 AM	5	Poiklinik-1	Q44
00000004	E	1	17	74	520	05/01/2014 11:30:00 AM	30.5	Poiklinik-3	A68,Q99
00000005	E	2	31	137	962	06/01/2014 11:30:00 AM	51.6	Poiklinik-2	L50
00000006	K	1	19	82	579	07/01/2014 11:30:00 AM	62.2	Poiklinik-1	A68
00000007	E	4	56	247	1732	08/01/2014 11:30:00 AM	64.7	Poiklinik-3	R05,J45
00000008	K	16	196	852	5972	09/01/2014 11:30:00 AM	131.8	Poiklinik-2	R05,J30.2,K21,H66
00000009	K	6	78	340	2386	10/01/2014 11:30:00 AM	98.7	Poiklinik-1	J45,L50
00000011	E	4	57	249	1745	11/01/2014 11:30:00 AM	73.5	Poiklinik-2	J45,L50
00000012	K	3	43	187	1317	12/01/2014 11:30:00 AM	41.0	Poiklinik-2	J45,L50
00000013	E	2	31	138	960	12/01/2014 11:30:00 AM	43.8	Poiklinik-2	J45,L50

At the bottom of the table, there is a pagination control showing "17 Sayfadan 1. Gösteriliyor" and a set of page numbers: "Önceki 1 2 3 4 5 ... 17 Sonraki".

Figure 2.7 Data filtering

When we click on one of the filter dropdown lists, the panel containing different kinds of filters with the perviously specified headings will show up. Figure 2.8 shows an example filter dropdown list with different types of filters. The following sections will provide information about different types of filters.



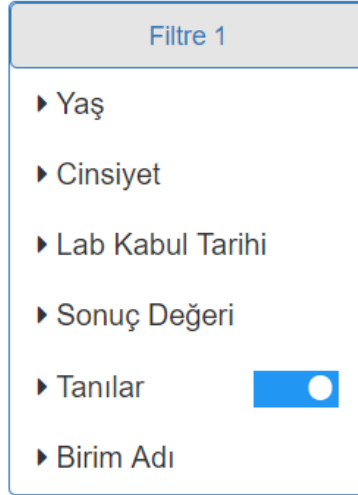


Figure 2.8 Filter drop-down list

### 2.2.2.1 Age Filter

This type of filter allows user to extract specific age range from the data. Bootstrap's "numpicker" library is used for this purpose. Numbers on top represents the lower limits, and numbers on bottom represents the upper limits for our filter. Numbers specified in upper limits will not be included in the filtered data. Entering numbers for different age types will result in applying both together. For example, if the user enters 5 and 3 as the upper limits of year and month respectively, the upper limit is set to 5 years 3 months. This feature allows us to perform more accurate filtering based on age. In the example of age filter shown in Figure 2.9, ages between 0 - <5 years are extracted from the data.

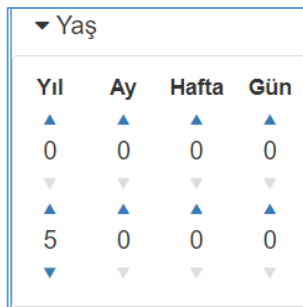


Figure 2.9 Age filter

### 2.2.2.2 Date Filter

This filter type is used to extract results between selected two dates. Bootstrap’s “daterangepicker” library is used for this purpose. Selections can be made through the calendar or manually entering the dates. The start and end dates are located at the top of the calendar. By default, the start and end dates are determined dynamically based on the first and last dates contained in the data. In the example of date filter shown in Figure 2.10, dates between 13/02/2014 and 21/03/2014 are extracted from the data.

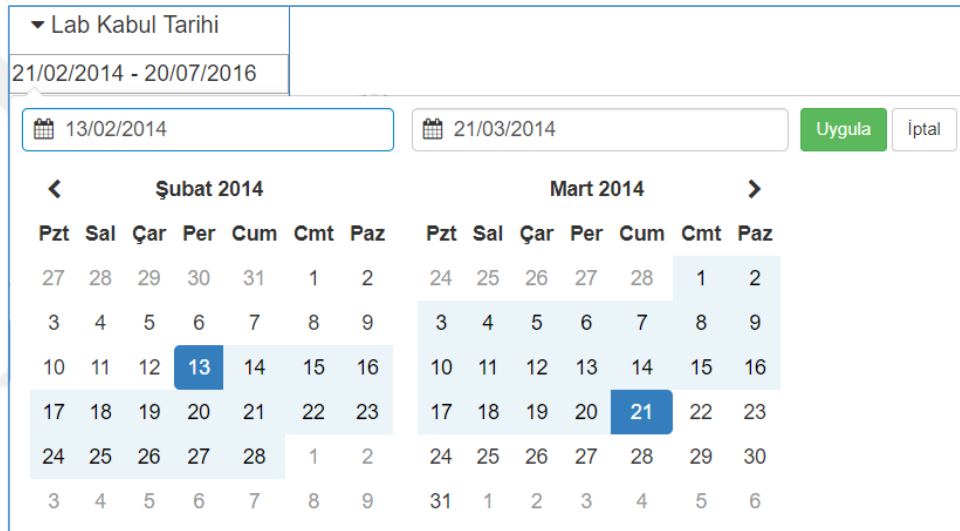


Figure 2.10 Date-range picker

### 2.2.2.3 Value Filter

Value filter is a maximum-minimum range picker for float values to filter data between specified range. Bootstrap’s “numpicker” library is used as in age filter but for float values. Which value corresponds to maximum or minimum is indicated by the texts on the pickers. By default, the minimum and maximum values are determined dynamically based on the minimum and maximum values contained in the data. Selections can be made using the arrows or if a more sensitive value is to be entered,

it can also be entered manually. Figure 2.11 shows an example value filtering, which is used to extract values between 10 and 50.

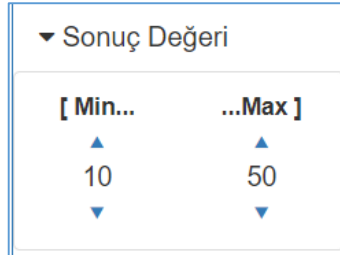


Figure 2.11 Value filter

#### 2.2.2.4 List Filter

List filter consists of a multi-select drop-down list used for filtering results with multiple text elements under a single column. Bootstrap’s “bootstrap-select” library is used which allows multiselection and searching options for a drop-down list. These elements in a single cell of a test result must be separated by commas in order to be detected as different elements. Each unique element detected in the test is added dynamically to the dropdown list. User may use the search box above to enter a keyword. By this means, the remaining results can be included or excluded from our data via “select all/deselect all” buttons. After selection/deselection, number of selected data is shown on the button. An example list filter used to filter diagnostics is shown in Figure 2.12.

Since a single test result may contain more than one element, even though an element is not selected in the drop-down list, a test result that contains that list item may not be filtered. Therefore, the toggle button next to the header determines whether the elements we selected are included or excluded from our data. Include/exclude states of the toggle button is shown in Figure 2.13.

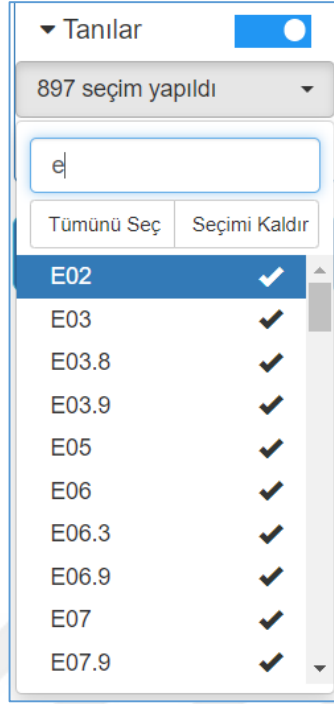


Figure 2.12 List filter

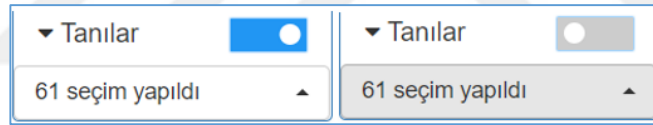


Figure 2.13 Include/exclude toggle

#### 2.2.2.5 Text Filter

Text filter is used for extracting strings from the data. As in list filter, Bootstrap’s “bootstrap-select” library is used for filtering data, which consists of a multi-select drop-down list. Unlike list filter, this type of filter should be used for columns containing only one text element. In this case, there is no separation operation taking place. Each element detected in the corresponding column in the test results is added dynamically to the dropdown list. User may use search box to enter keyword and eliminate some of the results and use “select all/deselect all” for the remaining results.

After selection/deselection, number of selected data is shown on the button. Figure 2.14 and Figure 2.15 shows two different text filters used for different columns.

2 seçim yapıldı	
poliklinik	
Tümünü Seç	Seçimi Kaldır
Poliklinik-1	✓
Poliklinik-2	✓
Poliklinik-3	
Poliklinik-4	
Poliklinik-5	

Figure 2.14 Text filter example-1 (for hospital unit)

Cinsiyet	
E	
Tümünü Seç	Seçimi Kaldır
E	✓
K	

Figure 2.15 Text filter example-2 (for gender)

After adjusting the filters, our data can be filtered by pressing the "Filtrele" button. After filtering, the number of results shown in the header is updated with the number of remaining results. Moreover, "Yeni Öğrenme", "Grafikler", "Düzenle" and "Dışa Aktar" tabs are now unlocked. Figure 2.16 shows the status of tabs and table after filtering.

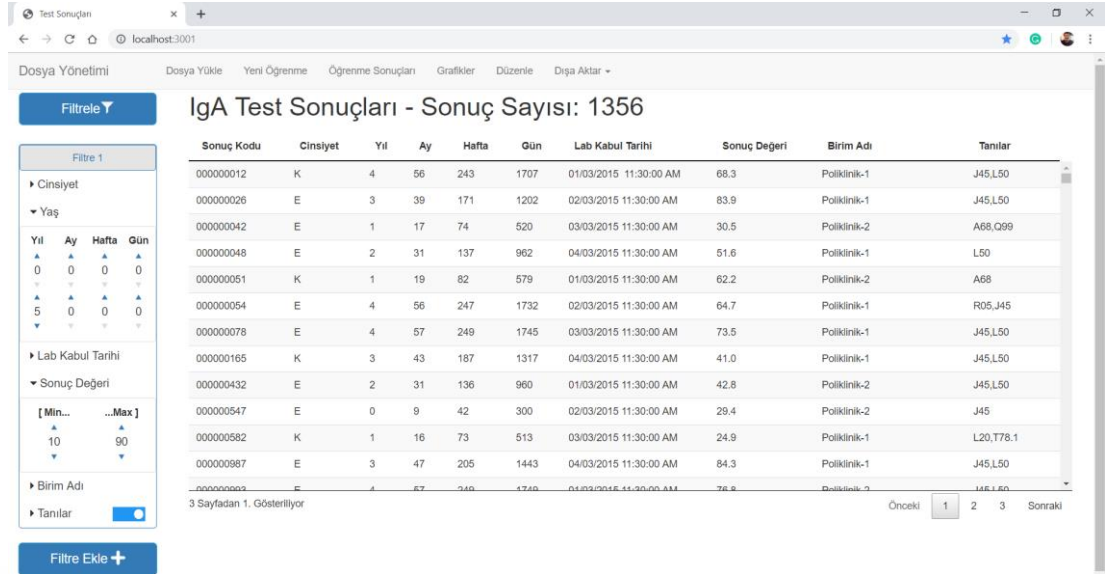


Figure 2.16 Data after filtering

### 2.2.3 Statistical Analysis

Under “Grafikler” tab, we can access several graphs of filtered data. These graphs show distribution of data in each column of the filtered data. An open source, high-level, declarative Javascript charting library “plotly.js” is used for doing these tasks. With the help of these graphs, filtered data can be analyzed deeply and make it easier to decide which parameters are more appropriate for learning algorithms. All graphs are created as histograms, where each bar shows the number of an element containing in a specific parameter. Histograms of each data type of different properties, such as distribution of genders, distribution of hospital units, distribution of diagnosis, and distributions of age groups (for year, month, week, and day) is displayed as shown from Figure 2.17 to Figure 2.23. In graphs that are created for data types list and text, hovering over the bars shows the text that the bar belongs to. Moreover, with a feature provided by the library, we can export graphs to our local storage with .png extension.



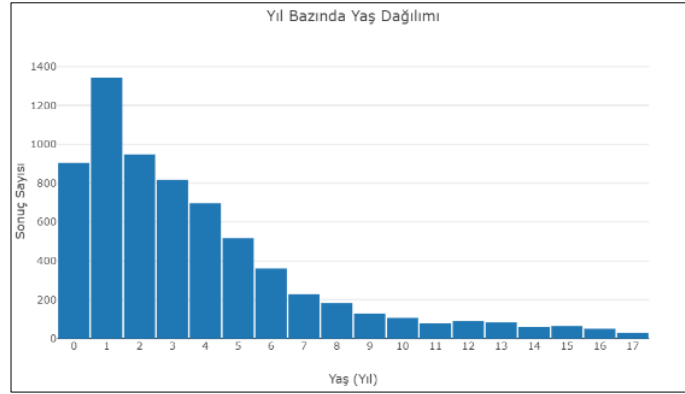


Figure 2.20 Age distribution (year)

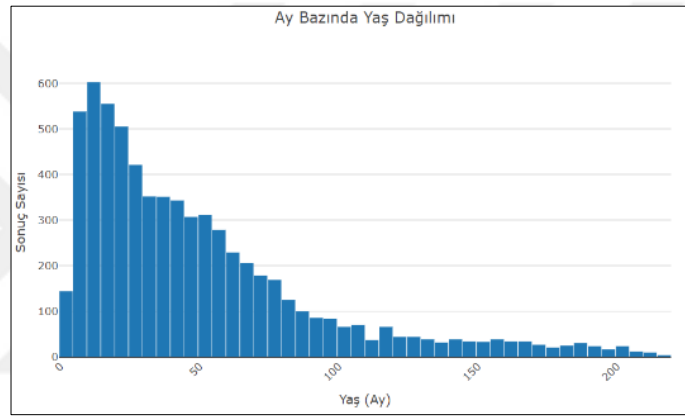


Figure 2.21 Age distribution (month)

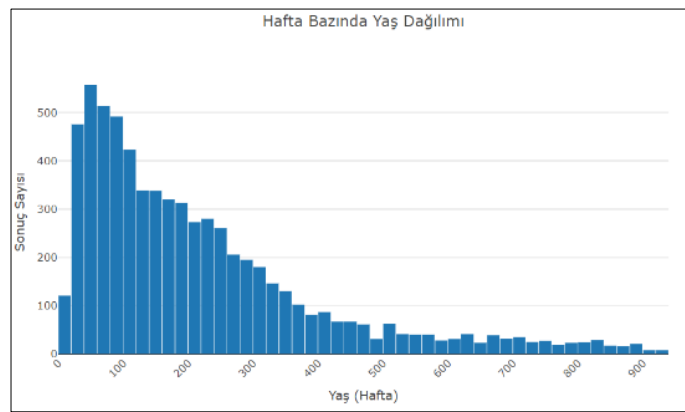


Figure 2.22 Age distribution (week)



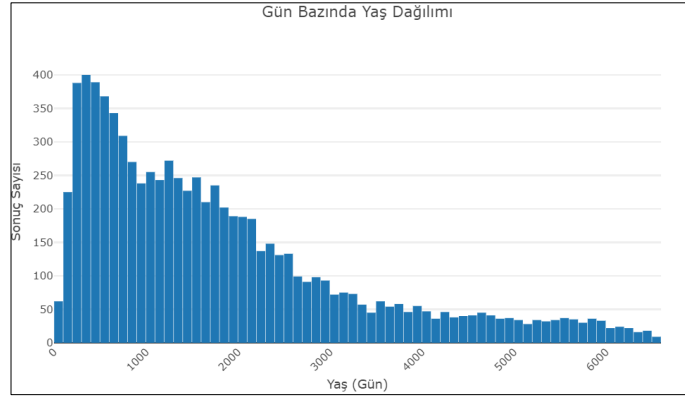


Figure 2.23 Age distribution (day)

### 2.2.4 Run Different Learning Algorithms

By clicking on the "Yeni Öğrenme" tab, we can open a modal on which we can apply learning algorithms to our filtered data as shown in Figure 2.24. In the first step, we need to give our study a title and choose the algorithm we want to apply from the drop-down list. Since our study will be held with this title in the database, it is important to enter a descriptive title in order to prevent possible confusions. At this stage we can only work with the GMM algorithm. However, it is planned to add new algorithms in the future.

**Öğrenme Paneli** ×

iga\_5cluster\_sonuc150altinda

Lütfen algoritma belirleyiniz...

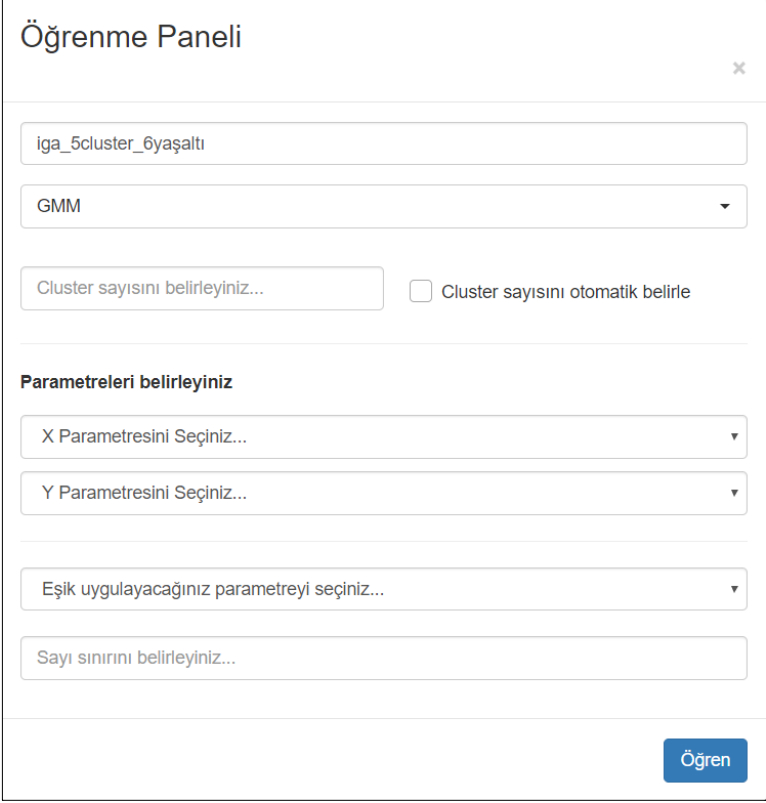
GMM

Öğren

Figure 2.24 Select learning algorithm

After the algorithm selection and title determination, we need to enter the parameters required for the selected algorithm in the panel below. In Figure 2.25, it can be seen that GMM algorithm requires us to specify the number of clusters, and x and y parameters to use in the algorithm. Parameters are picked from drop-down lists, which are filled dynamically with each column of the filtered data. If we do not have any prior knowledge about the number of clusters, we can enable the software to determine the optimal number of clusters by clicking "Cluster sayısını otomatik belirle" checkbox.

There is also an option to apply a threshold to one of these parameters to avoid unbalanced samples of parameters. To do this, bin sizes of histograms are taken as basis. If there are significant differences in number of samples for each bin, with the given threshold value, it is assumed that each bin has the same number of samples and the learning algorithm can be run without bias.



The screenshot shows a window titled "Öğrenme Paneli" with a close button (x) in the top right corner. The panel contains the following elements:

- A text input field containing "iga\_5cluster\_6yaşaltı".
- A dropdown menu showing "GMM".
- A text input field labeled "Cluster sayısını belirleyiniz..." and a checkbox labeled "Cluster sayısını otomatik belirle".
- A section header "Parametreleri belirleyiniz".
- A dropdown menu labeled "X Parametresini Seçiniz...".
- A dropdown menu labeled "Y Parametresini Seçiniz...".
- A dropdown menu labeled "Eşik uygulayacağınız parametreyi seçiniz...".
- A text input field labeled "Sayı sınırını belirleyiniz...".
- A blue button labeled "Öğren" in the bottom right corner.

Figure 2.25 Learning algorithm parameters

## 2.2.5 Learning Results

Under "Öğrenme Sonuçları" tab shown in Figure 2.26, we can display the results that previously sent for learning. These results are displayed with the date and time we send to learning besides the title we entered in the learning panel. These results are stored in a database and can be uploaded to our interface at any time.

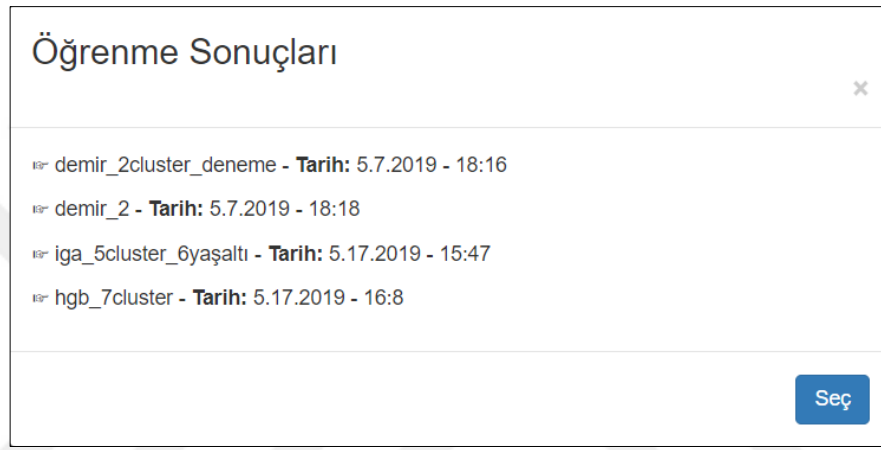


Figure 2.26 Learning result selection

After picking a learning result, the data that has been sent to learning will be loaded to the data table, with an additional cluster column. Cluster information for each data point is displayed in the leftmost column of the data table. Furthermore, with the additional cluster filter to our filter panel, we can apply filtering on cluster information. Bootstrap's "bootstrap-select" library is used for cluster filter as a multi-select drop-down list. Figure 2.27 shows an example of updated data table and cluster filter after loading a learning result.

Cluster	Sonuç Kodu	Hasta TC	Cinsiyet	Yıl	Ay	Hafta	Gün	Lab Kabul Tarihi	Sonuç Değeri	Birim Adı
0	00000012	0000000000	K	4	56	243	1707	01/01/2015 10:00:00 AM	68.3	Poliklinik-1
4	00000026	0000000001	E	3	39	171	1202	02/01/2015 10:00:00 AM	83.9	Poliklinik-2
3	00000042	0000000002	E	1	17	74	520	03/01/2015 10:00:00 AM	30.5	Poliklinik-3
1	00000048	0000000003	E	2	31	137	962	04/01/2015 10:00:00 AM	51.6	Poliklinik-1
1	00000187	0000000004	K	1	19	82	579	05/01/2015 10:00:00 AM	62.2	Poliklinik-2
0	00000215	0000000005	E	4	56	247	1732	06/01/2015 10:00:00 AM	64.7	Poliklinik-3
0	00000256	0000000006	E	4	57	249	1745	07/01/2015 10:00:00 AM	73.5	Poliklinik-1
4	00000312	0000000007	K	3	43	187	1317	08/01/2015 10:00:00 AM	41.0	Poliklinik-1
1	00000478	0000000008	E	2	31	136	960	09/01/2015 10:00:00 AM	42.8	Poliklinik-3
3	00001683	0000000009	E	0	9	42	300	10/01/2015 10:00:00 AM	29.4	Poliklinik-2
3	00002768	0000000010	K	1	16	73	513	11/01/2015 10:00:00 AM	24.9	Poliklinik-3
4	00002981	0000000011	E	3	47	205	1443	12/01/2015 10:00:00 AM	84.3	Poliklinik-3

Figure 2.27 Learning results mapping

After uploading a learning result, we can graphically display our data and corresponding clusters under "Grafikler" tab. Here we can find a scatter plot of our learning result and detailed information about each cluster. Figure 2.28 shows an example scatter plot, where each cluster is displayed with a different color. Plot also displays means and standart deviations of the clusters. Cluster distributions for x- and y-axes can be visualized with histograms as shown in Figure 2.29 and 2.30. Cluster information, such as number of samples, means for x- and y-axes, standart deviations for x- and y-axes, predicted reference intervals for x- and y-axes are displayed below these plots as shown in Figure 2.31. These are intended to provide a deeper understanding of the clusters.

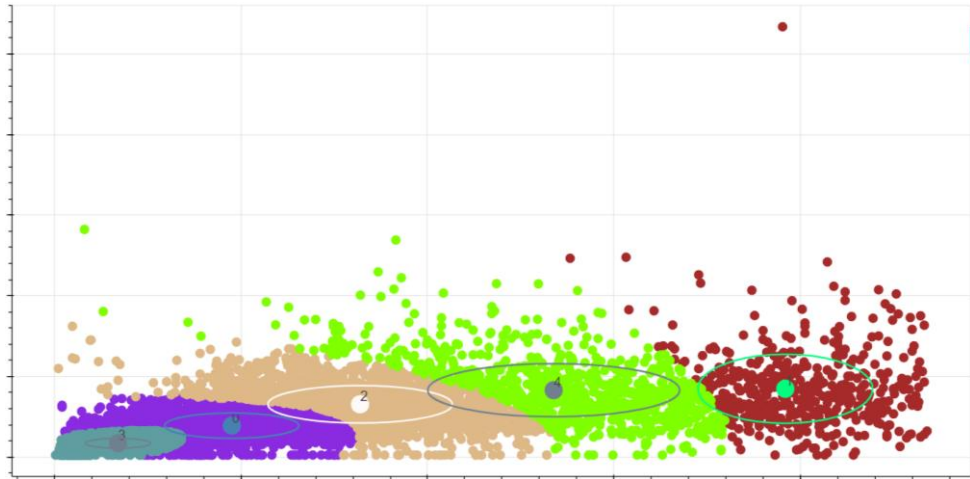


Figure 2.28 Cluster chart

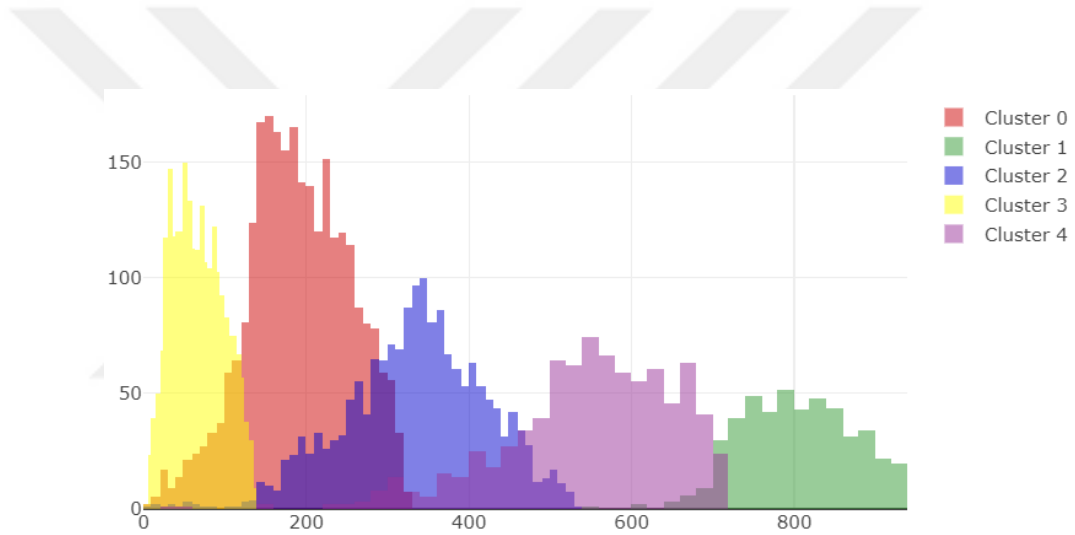


Figure 2.29 Cluster distribution (x-axis)

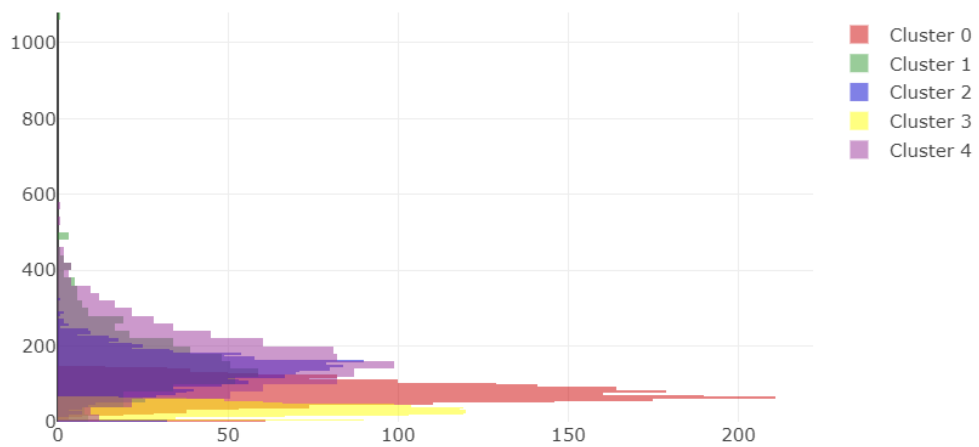


Figure 2.30 Cluster distribution (y-axis)

- Cluster 0
  - # of samples: 1763
  - Mean of x: 329
  - Mean of y: 132.89279636982417
  - Standard deviation of x: 84.39653319689816
  - Standard deviation of y: 47.753419269284294
  - Predicted reference interval of x: 160.20693360620368 to 497.7930663937963
  - Predicted reference interval of y: 37.38595783125558 to 228.39963490839276
- Cluster 1
  - # of samples: 500
  - Mean of x: 797.318
  - Mean of y: 169.32659999999998
  - Standard deviation of x: 70.4524582679696
  - Standard deviation of y: 93.35525155255058
  - Predicted reference interval of x: 656.4130834640607 to 938.2229165359392
  - Predicted reference interval of y: 0 to 356.03710310510115
- Cluster 2
  - # of samples: 2389
  - Mean of x: 66.29342821264127
  - Mean of y: 33.545332775219755
  - Standard deviation of x: 31.00483567803429
  - Standard deviation of y: 14.285430943360565
  - Predicted reference interval of x: 4.2837568856572694 to 128.30309956870985
  - Predicted reference interval of y: 4.974470888498626 to 62.11619466194088
- Cluster 3
  - # of samples: 845
  - Mean of x: 539.7609467455621
  - Mean of y: 167.81952662721898
  - Standard deviation of x: 106.04523826029887
  - Standard deviation of y: 80.54782715224239
  - Predicted reference interval of x: 327.6704702249644 to 751.8514232661598
  - Predicted reference interval of y: 6.723872322734195 to 328.91518093170373
- Cluster 4
  - # of samples: 2572
  - Mean of x: 187.7931570762053
  - Mean of y: 76.63662519440125
  - Standard deviation of x: 61.34481096122163
  - Standard deviation of y: 28.684447411304077
  - Predicted reference interval of x: 65.10353515376202 to 310.48277899864854
  - Predicted reference interval of y: 19.267730371793093 to 134.00552001700942

Figure 2.31 Cluster information

### 2.2.6 Edit Data

By clicking on the "Düzenle" tab, we can update all the values in the selected column from the drop-down list with the value that we specified. After entering the new value, all elements in that selected column will be updated. If a particular group

is planning to be updated, it must first be extracted using filters. Figure 2.32 shows the layout of the column edit modal.

Düzenle

Düzenlenmek istenen kolon

Yeni değeri giriniz

Onayla

Figure 2.32 Edit data

## 2.2.7 Export Data

The software has the ability to export data stored in the data table. “xls-export.js” library is used for doing this task. In this way, by clicking “Dışa Aktar” tab shown in Figure 2.33, user may export their current work to their local storage as .csv or .xls extensions and continue later by uploading them again.

Dışa Aktar

CSV

Excel

Cluster	Sonuç Kodu	Hasta TC	Cinsiyet	Yıl	Ay	Hafta	Gün	Lab Kabul Tarihi	Sonuç Değeri	Birim Adı
0	00000012	0000000000	K	4	56	243	1707	01/01/2015 10:00:00 AM	68.3	Poliklinik-1
4	00000026	0000000001	E	3	39	171	1202	02/01/2015 10:00:00 AM	83.9	Poliklinik-2
3	00000042	0000000002	E	1	17	74	520	03/01/2015 10:00:00 AM	30.5	Poliklinik-3
1	00000048	0000000003	E	2	31	137	962	04/01/2015 10:00:00 AM	51.6	Poliklinik-1
1	00000187	0000000004	K	1	19	82	579	05/01/2015 10:00:00 AM	62.2	Poliklinik-2
0	00000215	0000000005	E	4	56	247	1732	06/01/2015 10:00:00 AM	64.7	Poliklinik-3
0	00000256	0000000006	E	4	57	249	1745	07/01/2015 10:00:00 AM	73.5	Poliklinik-1
4	00000312	0000000007	K	3	43	187	1317	08/01/2015 10:00:00 AM	41.0	Poliklinik-1
1	00000478	0000000008	E	2	31	136	960	09/01/2015 10:00:00 AM	42.8	Poliklinik-3
3	00001683	0000000009	E	0	9	42	300	10/01/2015 10:00:00 AM	29.4	Poliklinik-2
3	00002768	0000000010	K	1	16	73	513	11/01/2015 10:00:00 AM	24.9	Poliklinik-3
4	00002981	0000000011	E	3	47	205	1443	12/01/2015 10:00:00 AM	84.3	Poliklinik-3
0	00003081	0000000012	E	4	67	240	1740	13/01/2015 10:00:00 AM	76.8	Poliklinik-2

12 Sayfadan 1. Gösteriliyor

Önceki 1 2 3 4 5 ... 12 Sonraki

Figure 2.33 Export data

### CHAPTER THREE

#### EXPERIMENTAL RESULTS

Some experiments were performed in order to test the performance of the software. During these experiments, GMM algorithm was used to calculate the reference intervals. Like other clustering algorithms, number of clusters need to be defined before running the algorithm. It is a problem to determine how many different distributions the data contains. For this reason, the number of clusters determined by CALIPER study (Colantonio, 2012) was used instead of dividing data into previously defined age groups. Biochemistry experts have chosen 11 biochemical tests that are frequently used for the trial studies to use during these experiments. Because of variations in algorithm outcomes between tests, data were normalized before running the algorithm. After that, 95% confidence intervals were calculated according to the mean and standard deviation values determined by the algorithm. Mean and standard deviation values of each cluster were included in the tables as well as calculated reference intervals. By comparing the outcomes of the experiments using the GMM algorithm and the outcomes of the CALIPER study, the performance of the algorithm was assessed.

Table 3.1 CALIPER Calcium reference intervals

Calcium mg/dL	Female Min.	Male Min.	Female Max.	Male Max.
0 to <1 year	8.5		11	
1 to <19 years	9.2		10.5	

Table 3.1 shows age groups, genders and corresponding reference intervals for Calcium test specified by CALIPER study. Colantonio et al. (2012) defined different reference ranges for age groups 0 to <1 year and 1 to <19 years. They also stated that gender is not a decisive criteria in Calcium test.



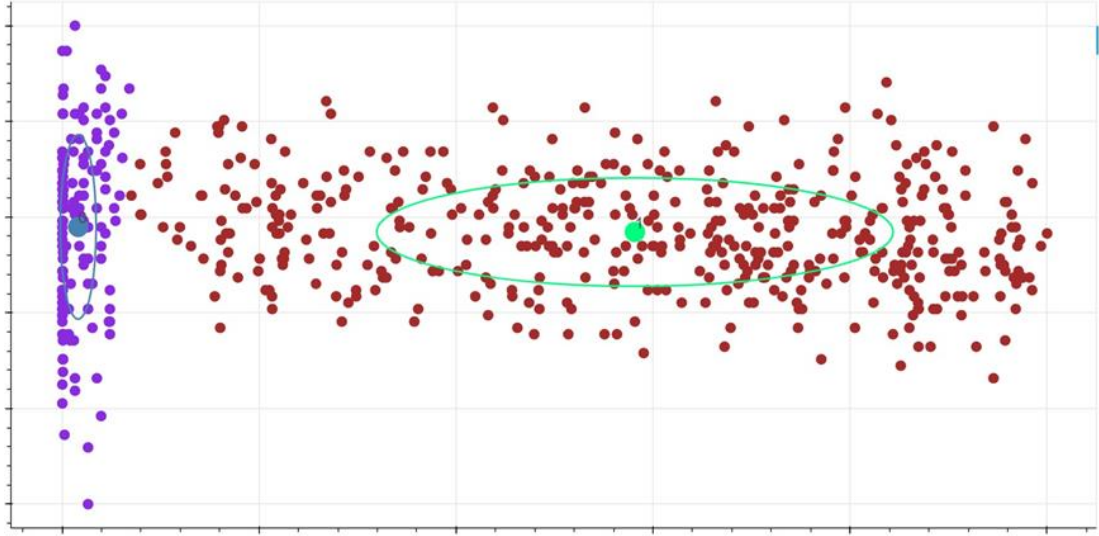


Figure 3.1 GMM Calcium female 2 cluster

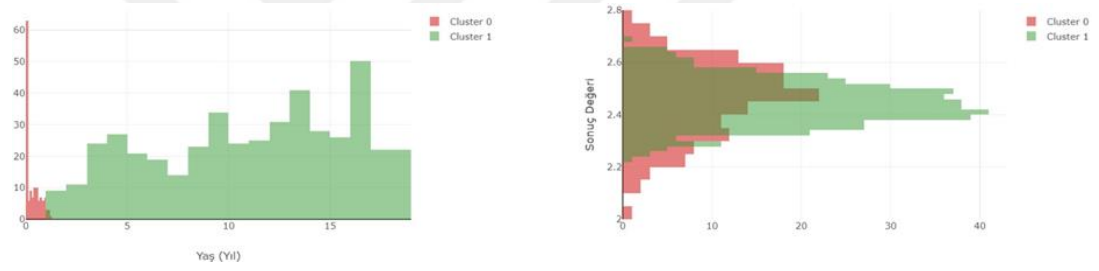


Figure 3.2 GMM Calcium female 2 cluster distribution

Table 3.2 GMM Calcium female 2 cluster values

Cluster	Age (Year)				Age Range	Result (mmol/L)				mg/dL
	mean	sd	min	max		mean	sd	min	max	Reference Range
0	0.3221	0.3492	0	1.0206	0Y - 1Y	2.4626	0.1449	2.1728	2.7524	8.7 - 11
1	11.0919	4.8158	1.4603	20.7234	1Y - 19Y	2.4514	0.0857	2.2800	2.6229	9.1 – 10.5

After running GMM algorithm with two clusters for both female and male results, we obtained plots shown in Figure 3.1 and Figure 3.2 for females, and in Figure 3.3 and Figure 3.4 for males. Detailed information about clusters are presented in Table 3.2 and Table 3.3.

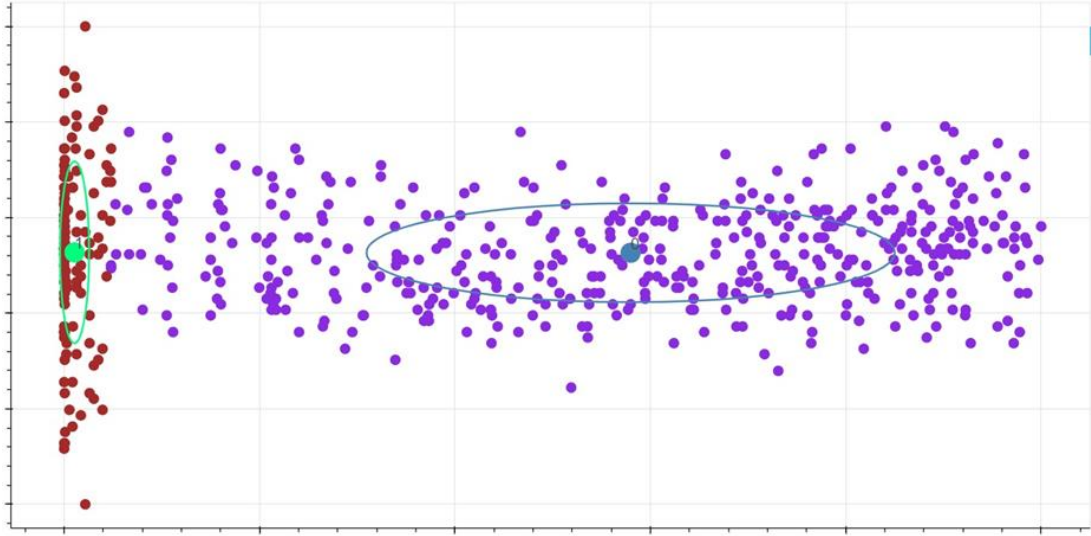


Figure 3.3 GMM Calcium male 2 cluster

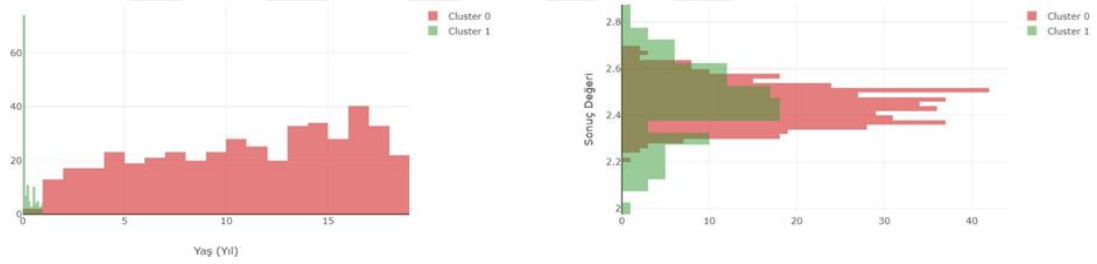


Figure 3.4 GMM Calcium male 2 cluster distribution

Table 3.3 GMM Calcium male 2 cluster values

Cluster	Age (Year)				Age Range	Result (mmol/L)				mg/dL
	mean	sd	min	max		mean	sd	min	max	Reference Range
1	0.2071	0.2729	0	0.7530	0M - 9M	2.4537	0.1624	2.1288	2.7785	8.5 – 11.1
0	11.0136	5.0279	0.9578	21.0693	1Y - 19Y	2.4526	0.0889	2.2750	2.6302	9.1 – 10.5

Before doing the comparison, calculated reference values were subjected to unit conversion (from mmol/L to mg/dL). When we evaluated the calculated results, we can say that GMM and CALIPER results for Calcium test are compatible with each other for both males and females.

Table 3.4 CALIPER Albumin G reference intervals

Albumin G g/dL	Female Min	Male Min	Female Max	Male Max
0 to 14 days	3.3		4.5	
15 days to <1 year	2.8		4.7	
1 to <8 years	3.8		4.7	
8 to <15 years	4.1		4.8	
15 to <19 years	4	4.1	4.9	5.1

Table 3.4 shows age groups, genders and corresponding reference intervals for Albumin G test specified by CALIPER study. For this test, Colantonio et al. (2012) defined different reference ranges for age groups 0 to 14 days, 15 days to <1 year, 1 to <8 years, 8 to <15 years and 15 to <19 years. They also stated that gender is a decisive criteria in Albumin G test between ages 15 and 19. Therefore, they have established different reference intervals for males and females from ages 15 to 19.

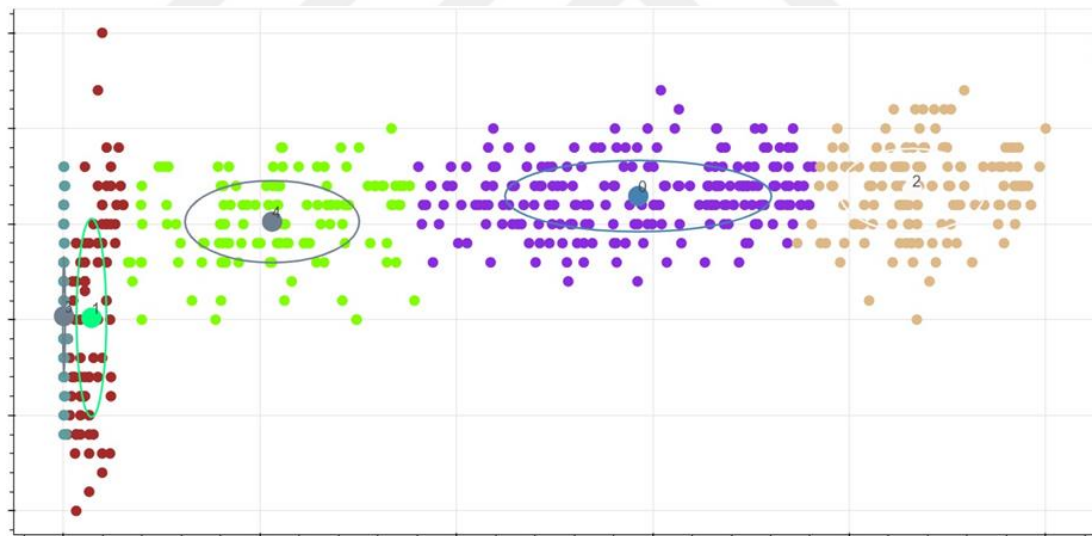


Figure 3.5 GMM Albumin G female 5 cluster

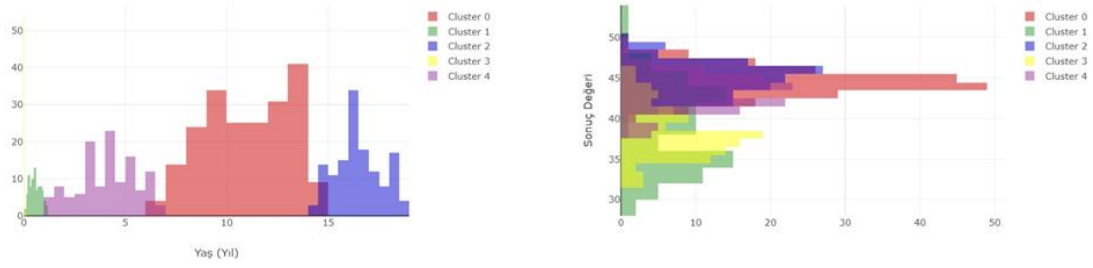


Figure 3.6 GMM Albumin G female 5 cluster distribution

Table 3.5 GMM Albumin G female 5 cluster values

Cluster	Age (Year)				Age Range	Result (g/L)				g/dL Reference Range
	mean	sd	min	max		mean	sd	min	max	
3	0.0166	0.0102	0	0.0372	0D - 14D	38.1760	3.1264	31.9231	44.4288	3.2 – 4.4
1	0.5590	0.2813	0	1.1215	0M - 13M	38.3851	5.2054	27.9743	48.7958	2.8 – 4.9
4	4.1021	1.4402	1.2217	6.9824	14M - 7Y	43.1230	2.1599	38.8031	47.4428	3.9 – 4.7
0	11.0074	2.1122	6.7829	15.2319	7Y - 15Y3M	44.4734	1.8173	40.8389	48.1080	4.1 – 4.8
2	16.4874	1.1418	14.2038	18.7710	14Y3M - 19Y	44.7941	2.2033	40.3875	49.2008	4 – 4.9

After running GMM algorithm with five clusters for both female and male results, we obtained plots shown in Figure 3.5 and Figure 3.6 for females, and in Figure 3.7 and Figure 3.8 for males. Detailed information about clusters are presented in Table 3.5 and Table 3.6.

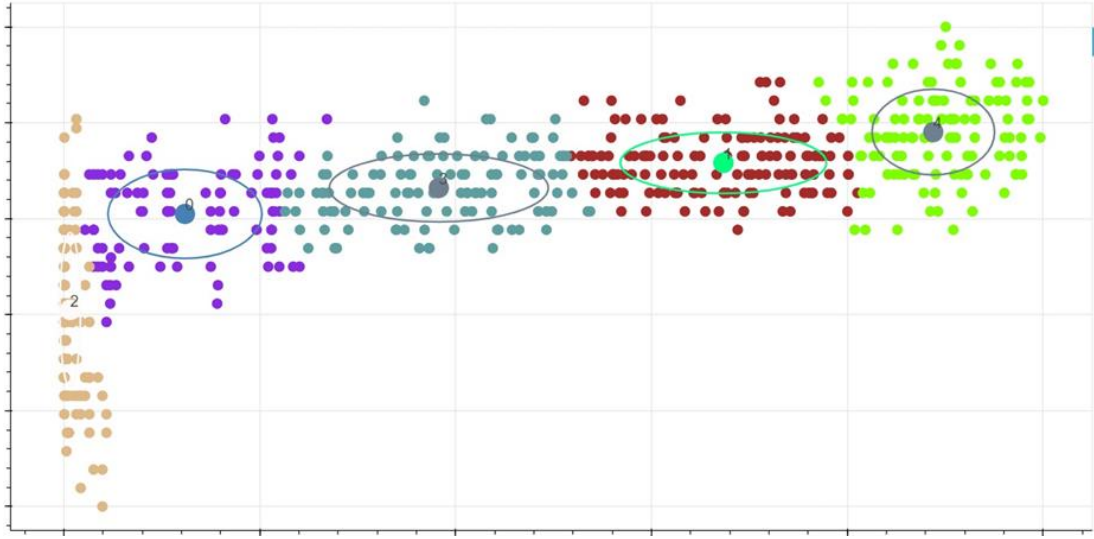


Figure 3.7 GMM Albumin G male 5 cluster

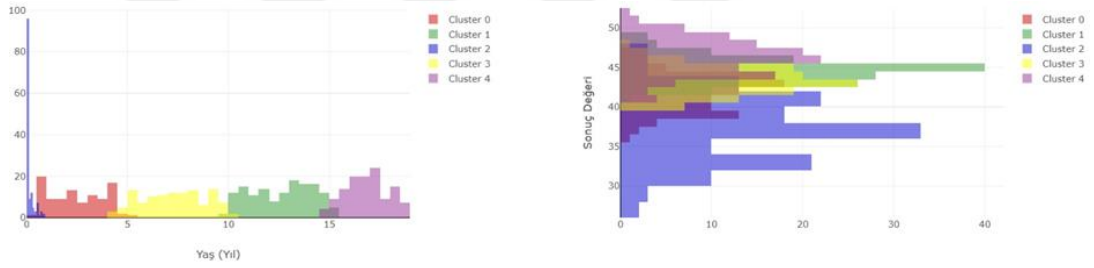


Figure 3.8 GMM Albumin G male 5 cluster distribution

Table 3.6 GMM Albumin G male 5 cluster distribution

Cluster	Age (Year)				Age Range	Result (g/L)				g/dL Reference Range
	mean	sd	min	max		mean	sd	min	max	
2	0.1269	0.1947	0	0.5163	0M - 6M	36.8152	4.2346	28.3461	45.2843	2.8 – 4.5
0	2.4393	1.3085	0	5.0564	0M - 5Y	41.9242	2.4051	37.1140	46.7345	3.7 – 4.7
3	7.2852	1.5664	4.1524	10.4182	4Y2M - 10Y5M	43.2609	1.7648	39.7312	46.7905	4 – 4.7
1	12.6233	1.5271	9.5691	15.6776	9Y6M - 15Y8M	44.6383	1.6035	41.4312	47.8454	4.1 – 4.8
4	16.8398	1.0226	14.7946	18.8850	14Y10M - 19Y	46.2705	2.3260	41.6184	50.9226	4.2 – 5.1

Before doing the comparison, calculated reference values were subjected to unit conversion (from g/L to g/dL). For Albumin G test, obtained results by GMM algorithm for females are compatible with the results established by CALIPER study. However, differences in age groups have been observed for males.



## **CHAPTER FOUR**

### **CONCLUSION AND FUTURE WORK**

The aim of this study is to make some experiments to establish pediatric reference intervals by using data mining techniques instead of conventional statistical methods, which are difficult and laborious to process. The software tool has been developed to assist specialists through the process of establishing new pediatric reference intervals using clinical laboratory tests. This tool includes many features, such as cleaning and advanced filtering to reduce the size or work on specific groups, visualizing dataset using histograms to get more statistical information about laboratory test result data, import/export data, apply machine learning algorithms to extract sub-groups and interpret their results by plot diagrams, and iterate this process to refine the results. With this tool, it is ensured that each clinic or hospital could make its own reference interval study, and if the results were consistent with the clinical values, they could be published as an acceptable reference interval.

The literature review showed that a lot of work has been done in establishing reference intervals but the number of studies on pediatric patients is not sufficient. Moreover, studies involving data mining techniques are infrequent. Therefore, there is a need to focus on this topic.

In the future, in addition to the GMM (Gaussian Mixture Model) algorithm (Sobay, 2019), different machine learning algorithms can be integrated into the software as a plugin to enhance the system. In addition, this thesis will be used as a tool for preprocessing operations in the PhD study (Yıldırım, 2019).

## REFERENCES

- Adeli, K., Higgins, V., Trajcevski, K., & White-Al Habeeb, N. (2017). The Canadian laboratory initiative on pediatric reference intervals: A CALIPER white paper. *Critical Reviews in Clinical Laboratory Sciences*, 54(6), 358-413.
- Akbayir, S., Balci Fidanci, S., Sen, F., Yurtsever Bakir, A., Orekici Temel, G., & Unal, N., et al. (2011). Mersin bölgesinde homosistein, vitamin A ve vitamin E düzeylerine ait referans aralıklarının belirlenmesi. *Mersin Üniversitesi Sağlık Bilim Dergisi*, 4(1), 7-11.
- Baadenhuijsen, H., & Smit, J. C. (1985). Indirect estimation of clinical chemical reference intervals from total hospital patient data: application of a modified Bhattacharya procedure. *Clinical Chemistry and Laboratory Medicine*, 23(12), 829-39.
- Colantonio, D., Kyriakopoulou, L., Chan, M., Daly, C., Brinc, D., Venner, A., et al. (2012). Closing the gaps in pediatric laboratory reference intervals: A CALIPER database of 40 biochemical markers in a healthy and multiethnic population of children. *Clinical Chemistry*, 58(5), 854-868.
- Çaycı, T., Kurt, Y. G., Honca, T., Taş, A., Özgürtaş, T., Agilli, M., et al. (2015). Hastane bilgi sistemindeki kayıtlı hasta sonuçlarından tam kan referans aralıklarının tayini. *Gulhane Tıp Dergisi*, 57, 111-117.
- Horn, P. S., Pesce, A. J., & Copeland, B. E. (1998). A robust approach to reference interval estimation and evaluation. *Clinical Chemistry*, 44(3), 622-631.
- Horowitz, G. (2010). *Defining, establishing, and verifying reference intervals in the clinical laboratory*. Wayne, PA: Clinical and Laboratory Standards Institute.



- Humberg, A., Kammer, J., Mordmüller, B., Kremsner, P., & Lell, B. (2010). Haematological and biochemical reference intervals for infants and children in Gabon. *Tropical Medicine & International Health*, 16(3), 343-348.
- Jagarinec, N., Flegar-Meštrić, Z., Šurina, B., Vrhovski-Hebrang, D., & Preden-Kereković, V. (1998). Pediatric reference intervals for 34 biochemical analytes in urban school children and adolescents. *Clinical Chemistry and Laboratory Medicine*, 36(5), 327-337.
- Kapelari, K., Kirchlechner, C., Högler, W., Schweitzer, K., Virgolini, I., & Moncayo, R. (2008). Pediatric reference intervals for thyroid hormone levels from birth to adulthood: a retrospective study. *BMC Endocrine Disorders*, 8(15), 1-10.
- Katayev, A., Balciza, C., & Seccombe, D. (2010). Establishing Reference Intervals for Clinical Laboratory Test Results. *American Journal of Clinical Pathology*, 133(2), 180-186.
- Katayev, A., Fleming, J., Luo, D., Fisher, A., & Sharp, T. (2015). Reference intervals data mining. *American Journal of Clinical Pathology*, 143(1), 134-142.
- Linnet, K. (1987). Two-stage transformation systems for normalization of reference distributions evaluated. *Clinical Chemistry*, 33(3), 381-386.
- Lumsden, J. H., & Mullen, K. (1978). On establishing reference values. *Canadian Journal of Comparative Medicine: Revue Canadienne de Medecine Comparee*, 42(3), 293-301.
- Orekici Temel, G., Ovla, H. D., Dericci Yildirim, D., & Kalafat, H. (2015). Hastane biyokimyasal verilerinden sağlıklı alt grubun belirlenmesi: Bhattacharya prosedürü. *Düzce Üniversitesi Sağlık Bilimleri Enstitüsü Dergisi*, 5(2), 28-31.

Sasse, E. (2000). *How to define and determine reference intervals in the clinical laboratory*. Wayne, PA: NCCLS.

