JULY, 2019

M. Sc. THESIS IN ELECTRONICS AND COMPUTER ENGINEERING

Mehmet Şükrü MUMBUÇOĞLU

REPUBLIC OF TURKEY

HASAN KALYONCU UNIVERSITY

GRADUATE SCHOOL OF

NATURAL AND APPLIED SCIENCES

CLASSIFICATION OF MICROARRAY GENE EXPRESSION CANCER DATA BY USING ARTIFICIAL INTELLIGENCE METHODS

M. Sc. THESIS

IN

ELECTRONICS AND COMPUTER ENGINEERING

BY

MEHMET ŞÜKRÜ MUMBUÇOĞLU

JULY 2019

CLASSIFICATION OF MICROARRAY GENE EXPRESSION CANCER DATA BY USING ARTIFICIAL INTELLIGENCE METHODS

M.Sc. Thesis

in

Electronics and Computer Engineering

Hasan KALYONCU University

Supervisor

Asst. Prof. Dr. Bülent HAZNEDAR

by

Mehmet Şükrü MUMBUÇOĞLU

JULY 2019

REPUBLIC OF TURKEY

HASAN KALYONCU UNIVERSITY

GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES

ELECTRONICS AND COMPUTER ENGINEERING

Name of the thesis: Classification of Microarray Gene Expression Cancer Data by Using Artificial Intelligence Methods

Name of the student: Mehmet Şükrü MUMBUÇOĞLU

Exam date: July 10, 2019

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Mehmet KARPUZCU

Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Muhammet Fatih HASOGLU

Head of Department

This is to certify that we have read this thesis and that in our consensus/majority opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Bülent HAZNEDAR

Supervisor

Examining Committee Members

Signature

Assoc. Prof. Dr. Muhammet Fatih HASOĞLU …….

Asst. Prof. Dr. Bülent HAZNEDAR …….

Asst. Prof. Dr. Tolgay KARA …….

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work

Mehmet Şükrü MUMBUÇOĞLU

Signature

ABSTRACT

CLASSIFICATION OF MICROARRAY GENE EXPRESSION CANCER DATA BY USING ARTIFICIAL INTELLIGENCE METHODS

MUMBUÇOĞLU, Mehmet Şükrü
M.Sc. in Computer Engineering
Supervisor: Asst. Prof. Dr. Bülent HAZNEDAR
July 2019, 145 pages

Today, the development of computer technologies has affected the studies in many areas. Advances in molecular biology and computer technologies have revealed the science of bioinformatics. Rapid developments in the field of bioinformatics have contributed greatly to the solution of many problems waiting to be solved in this field. The classification of DNA microarray gene expressions is one of these problems. DNA microarray studies are a technology used in the field of bioinformatics. DNA microarray data analysis plays a very effective role in the diagnosis of diseases related to genes such as cancer. By determining gene expressions depending on the type of disease, it can be determined with great success rate whether any individual possesses the diseased gene. The use of high-performance classification techniques on microarray gene expressions is of great importance to determine whether an individual is healthy.

There are many methods for classifying DNA microarrays. Support Vector Machines, Naive Bayes, k-Nearest Neighbour, Decision Trees, such as many statistical methods are widely used. However, when these methods are used alone, they do not always give high success rates in classifying microarray data. Therefore, the use of artificial intelligence-based methods to achieve high success rates in the classification of microarray data is seen in the studies.

In this study, in addition to these statistical methods, it is aimed to obtain higher success rates by using a method such as ANFIS based on artificial intelligence. K-Nearest Neighbourhood, Naive Bayes and Support Vector Machines were used as statistical classification methods. Here, studies on two different cancer data, namely breast and central nervous system cancer, have been conducted.

According to the information obtained from the results, it was found that artificial intelligence based ANFIS technique was more successful than statistical methods.

Keywords: Microarray Gene Expression, Data mining, Feature selection, Classification, ANFIS

ÖZET

YAPAY ZEKA YÖNTEMLERİ KULLANILARAK MIKROARRAY GEN İFADE KANSER VERİLERİNİN SINIFLANDIRILMASI

MUMBUÇOĞLU, Mehmet Şükrü
Yüksek Lisans Tezi, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Asst. Prof. Dr. Bülent HAZNEDAR
Temmuz 2019, 145 sayfa

Günümüzde bilgisayar teknolojilerinin gelişmesi ile birçok alanda yapılan çalışmaları etkilemiştir. Moleküler biyoloji ve bilgisayar teknolojilerinde meydana gelen gelişmeler biyoinformatik adlı bilimi ortaya çıkarmıştır. Biyoinformatik alanında meydana gelen hızlı gelişmeler, bu alanda çözülmeyi bekleyen birçok probleme çözüm olma yolunda büyük katkılar sağlamıştır. DNA mikroarray gen ekspresyonlarının sınıflandırılması da bu problemlerden birisidir. DNA mikroarray çalışmaları, biyoinformatik alanında kullanılan bir teknolojidir. DNA mikroarray veri analizi, kanser gibi genlerle alakalı hastalıkların teşhisinde çok etkin bir rol oynamaktadır. Hastalık türüne bağlı gen ifadeleri belirlenerek, herhangi bir bireyin hastalıklı gene sahip olup olmadığı büyük bir başarı oranı ile tespit edilebilir. Bireyin sağlıklı olup olmadığının tespiti için, mikroarray gen ekspresyonları üzerinde yüksek performanslı sınıflandırma tekniklerinin kullanılması büyük öneme sahiptir.

DNA mikroarray'lerini sınıflandırmak için birçok yöntem bulunmaktadır. Destek Vektör Makinaları, Naive Bayes, k-En yakın Komşu, Karar Ağaçları gibi birçok istatistiksel yöntemler yaygın olarak kullanlmaktadır. Fakat bu yöntemler tek başına kullanıldığında, mikroarray verilerini sınıflandırmada her zaman yüksek başarı oranları vermemektedir. Bu yüzden mikroarray verilerini sınıflandırmada yüksek başarı oranları elde etmek için yapay zekâ tabanlı yöntemlerin de kullanılması yapılan çalışmalarda görülmektedir.

Bu çalışmada, bu istatistiksel yöntemlere ek olarak yapay zekâ tabanlı ANFIS gibi bir yöntemi kullanarak daha yüksek başarı oranları elde etmek amaçlanmıştır. İstatistiksel sınıflandırma yöntemleri olarak K-En Yakın Komşuluk, Naive Bayes ve Destek Vektör Makineleri kullanılmıştır. Burada Göğüs ve Merkezi Sinir Sistemi kanseri olmak üzere iki farklı kanser veri seti üzerinde çalışmalar yapılmıştır.

Sonuçlardan elde edilen bilgilere göre, genel olarak yapay zekâ tabanlı ANFIS tekniğinin, istatistiksel yöntemlere göre daha başarılı olduğu tespit edilmiştir.

Anahtar Kelimeler: Mikrodizi gen ifadeleri, Veri madenciliği, Öznitelik seçimi, Sınıflandırma, ANFIS

To My Beloved Parent & Family

ACKNOWLEDGEMENTS

During the preparation of this thesis, I would like to express my deepest gratitude to my esteemed advisor, Asst. Prof. Dr.Bülent HAZNEDAR, who supported and encouraged me, who always keeps my motivation highest, who never lost his faith in me in this process, and who made recommendations and guided me by his experience.

I would also like to thank Assoc. Prof. Dr.Muhammet Fatih HASOĞLU for his suggestions and comments.

TABLE OF CONTENTS

Pages

LIST OF FIGURES

Pages

LIST OF TABLES

Pages

LIST OF ABBREVIATIONS

DNA	Deoxyribonucleic Acid
RNA	Ribonucleic Acid
mRNA	Messenger Ribonucleic Acid
SVM	Support Vector Machine
NB	Naïve Bayes
ANFIS	Adaptive Neural-Fuzzy Inference Systems
KNN	K Nearest-Neighbour
ANN	Artificial Neural Networks
DT	Decision Tree
SBR	Status-Based Reasoning
RSA	Rough Set Approach
EA	Evolutionary Algorithms
SI	Swarm Intelligence
RC	Rough cluster
TP	True Positive
FN	False Negative
miRNA	microRNA
RBF	Radial Basis Function
SMO	Sequential Minimal Optimization
MAE	Mean Absolute Error
RMSE	Root Mean Squared Error
TPR	True Positive Rate
FPR	False Positive Rate

CHAPTER 1

INTRODUCTION

1.1. General

Since the ancient times, the branches of science which are concerned by humanity are called basic sciences and the problems encountered are classified among these disciplines and solutions have been sought. In time, problems arose where several of the basic disciplines were to find solutions, and interdisciplinary work gained importance. The new disciplines have emerged with the increase of the issues that need to be studied in the same interdisciplinary. Bioinformatics is one of these disciplines. Bioinformatics; In addition to the field of biology, the integration of chemistry and medical science with information sciences, mathematics and statistics has emerged as a result of the collection of information about biological events and the evaluation of the collected information. Bioinformatics, based on the use of information technologies in the solution of biological problems, helps to explain biological phenomena at the molecular level. In bioinformatics, biological information is obtained numerically and stored in databases. Bioinformatics plays a very important role in medical sciences. In recent years, applications in medical sciences have focused on gene expression analysis. Generally, the expressions of cells affected by different diseases are compiled and compared with healthy cells and the diagnosis is made from the differences (Polat & Karahan, 2009).

Thanks to developments in the world of science, interest in the field of microarray studies is increasing day by day. DNA microarray studies are a very comprehensive technology used in molecular biology and medicine. DNA microarray data analysis; It plays an important role in the identification of diseases associated with genes such as cancer. The relevant genes for the type of disease can be determined and can be calculated in the high probability that any individual is patient or intact. For this, high performance classification methods are very important in microarray data (Korkem, 2013).

Data mining has contributed to classification methods for bioinformatics. Classification; It is a method used to reveal hidden patterns in the database. By classification, the database is divided into small homogeneous groups according to certain characteristics. Classification is an analysis technique that shows which class a new parent belongs to and is based on a learning algorithm. The purpose of this algorithm is; creating a classification model, which class to belong to an unknown data is to determine the class. Different methods are used to access information in data mining. There are many algorithms that belong to these methods. Many studies have been conducted on which of these algorithms give better results and different results have been obtained from the studies. The main reasons for obtaining different results are the pre-processing on the data, the choice of the parameters of the selection of the parameters, the selection of the parameters of the algorithms used and the differentiation of the programs in which the classification methods are applied.

The aim of this study is to divide the data obtained from cancer cells into the correct classes in the targeted bioinformatics field. With the correct classification, important findings can be obtained. In the classification of microarray gene expression data, these statistical classification methods and Adaptive Neural-Fuzzy Inference Systems (ANFIS) are used.

Support Vector Machines, K-Nearest Neighbourhood, Naive Bayes statistical classification algorithms are used as methodology. The data to which the classification methods are applied are the microarray gene expressions. These gene expressions (Zhu, Ong, & Dash, 2007) are the data of patients with Breast Cancer and Central Nervous System Cancer from the site of Shenzhen University (Shenzhen University, 2018)

In this study, both statistical methods and recently started to be studied by researchers and artificial intelligence-based ANFIS method, which has recently started to enter the literature, has been classified by using as hybrid.

Apart from the methods used to classify the data commonly used in the literature, new approaches to classifying microarray gene expression data by using this ANFIS have been used to compare the performance of these new approaches with commonly used methods.

1.2. Structure of Thesis

The organization of the thesis is as follows; In the first chapter, the theoretical information about data mining is given and then it is emphasized that the data to be used in this thesis is microarray gene expression data and theoretical information is given about these data. It is revealed that the microarray data to be studied will be processed by the classification method which is one of the data mining methods after the normalization process. Besides, the theoretical information about the classification is given and what kind of classification methods will be used in this study is explained.

In the second part of the thesis, the classification tools used in the study are discussed and information about these programs is given. Then, microarray gene expression data to be used in the study is high-dimensional data to reduce the size of these data, Correlation-based feature selection method is given information about. Information about Naive Bayes, Support Vector Machines, K-Nearest Neighbour methods and Artificial Intelligence-based ANFIS, which will be used to classify diminished microarray data, has been given information about ANFIS. Then, the models to be applied and the appropriate arrangements are discussed.

In the third chapter, information about the parameters used in the study was given and information was given about similar studies. In the last chapter, the results were compared with the previous studies and recommendations were made.

CHAPTER 2

GENERAL INFORMATION

2.1.History of Data Mining

In the 1950s, mathematicians worked on data mining techniques to reveal artificial intelligence and machine learning areas in the fields of logic and computer science. In the 1960s, the statisticians discovered regression analysis and the greatest likelihood estimation, which were the first steps in data mining. In the following 20 years, firstly, the classification of the data and the establishment of the relational links between these classes and the concept of database have been revealed. In the 1990s, the first steps of the discovery of knowledge were created in the database and a data warehouse was developed for large databases. Data mining has been widely used at the same time with new technologies.

2.2.Data Mining

Data mining is the process of automatically extracting structured information from databases. This process is a special part of the general process called information discovery from databases (Fayyad, 1997).

Data mining is the process of discovering interesting information, such as models, patterns, relationships, deviations, and meaningful structures derived from databases where a large number of data is stored (Han, Cheng, Xin, & Yan, 2007).

Fayyad and friends. data mining is a step in the knowledge discovery process in databases that consists of applying data analysis and discovery algorithms that, under acceptable computational efficiency limitations, produce a particular enumeration of patterns (or models) over the data (Fayyad, Piatetsky-Shapiro, & Smyth, 1996).

In recent years, information collection has become much easier, but the effort needed to uncover the peculiarities of the information in hand has increased greatly in large-scale databases (De Falco, Della Cioppa, & Tarantino, 2002).

The rapid increase in data collection and storage technologies has led to an over-expansion of data in storage types such as data base, data or warehouse (Zhou, 2013). In contrast, the number of scientists, engineers and analysts remains unchanged (Vahaplar & İnceoğlu, 2001).

For example; In the field of clinical treatment, there are difficulties in discovering information from growing volume data. The continuous collection of the physiological parameters of the patients under observation now leads to the emergence of enormous volumes of information. Growing amounts of data prevent medical professionals performing manual analysis from performing their tasks. Many concealed and potentially beneficial relationships cannot be recognized by the analyst (Tan, Yu, Heng, & Lee, Evolutionary computing for knowledge discovery in medical diagnosis, 2003).

Traditional techniques allow you to prove your own hypothesis. As shown in Figure 2.1, approximately 5% of all relationships can be found in this way. Data mining is a gateway to the remaining 95% relations (Distilleries, 1999).

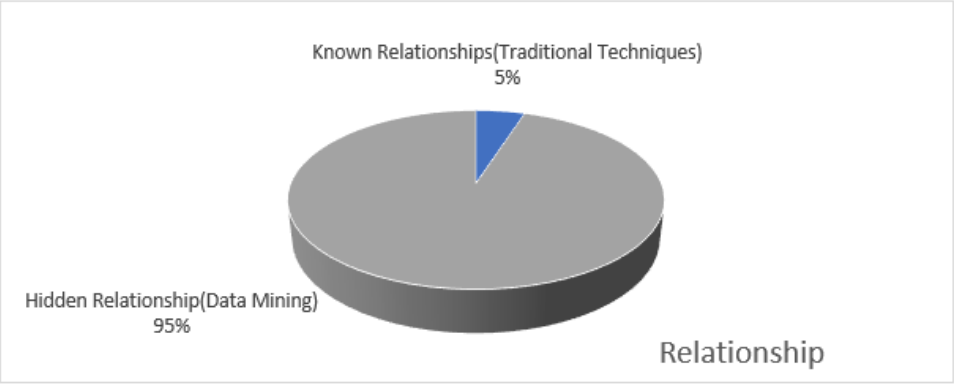

Figure 2.1. Relations (Distilleries, 1999)

Many of the time large databases are searched for unknown or unprecedented relationships, trends and patterns. These relationships or tendencies are generally assumed by engineers, analysts or market researchers, but these relationships need to be proved by the data from which they are obtained. New information it helps users do their job better (Kamrani, Rong, & Gonzalez, 2001). In general, the increase in interest in data mining can be explained by the following factors (Telcioğlu, 2007);

1. In the 1980s, companies created data bases for their customers, competitors and products. These databases are like potential goldmines. These data, which exceeds millions, contain confidential information and can easily be accessed using the SQL (Structured Query Language) database query language or other superficial query languages. SQL is just a query language and helps to find information under
2. previously known limitations. Data mining algorithms typically become evident in subgroups of the database or in appropriate clusters. In many instances, repeatable SQL queries are used and the average results are obtained. It is possible to do this manually, but it is quite tiring and long-lasting.
3. 2. Network usage on computers continues to improve. This makes it easier to connect to the database. Thus, a link can be made between the demographic data file and the client file and identification of the specific population groups can be established.
4. Over the last few years, the techniques of machine learning have improved. Neural networks, genetic algorithms, and other simple feasible learning techniques make it easy to create interesting links with databases.
5. The relationship between the customer and the service provider sends personal information from the computer at the service desk to the central information systems. Marketers and insurers also want to use these newly acquired techniques.

2.3.Data Mining and Interdisciplinary Relationship

The VM is a multi-disciplinary approach and incorporates many techniques. The close link between data mining and machine learning, statistics and database technologies

can be easily seen. These disciplines aim to find interesting associations and patterns within the data. Figure 2.2 shows the interdisciplinary relationship with data mining (Özbakır, Baykasoğlu, & Kulluk , 2008).

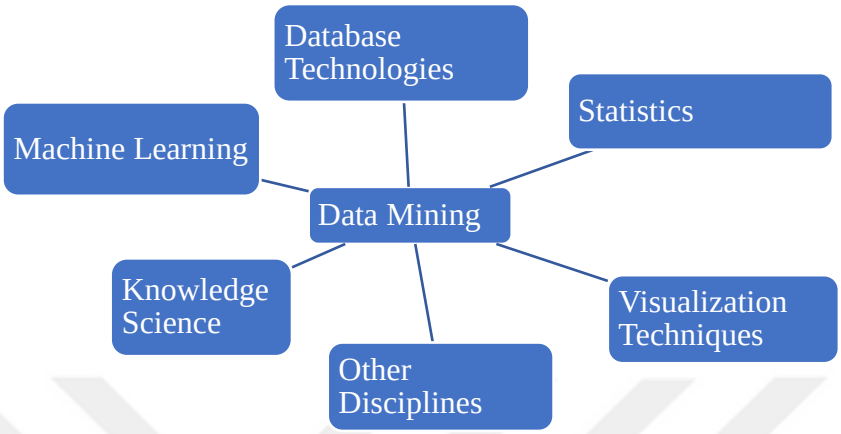

Figure 2.2. Interdisciplinary relationship with data mining

2.4.Applications of Data Mining

If the application areas of data mining are mentioned briefly (Han & Kamber, 2018);

- Marketing: Identifying customers' buying habits, demographic information, campaign products, new customers without losing existing customers earning, market basket analysis, customer relationship management and sales forecast areas are the most common data mining application areas.
- Banking and Insurance: In determining the correlation between different financial indicators; in determining credit card fraud, evaluating credit requests, determining customer profile according to credit card expenditures, determining insurance fraud and requesting new policy will be used intensively in the estimation of customers.
- Biology, Medicine and Genetics: Plant species breeding, gene map analysis and detection of genetic diseases, detection of cancer cells, the discovery and classification of new virus species, physiological parameters are used in the analysis and evaluation.

- Chemistry: The discovery and classification of new chemical molecules are used in the discovery of feed and drug species.
- Image Recognition and Robot Vision Systems: It is used in techniques such as obstacle recognition, road recognition, face recognition and fingerprint recognition by means of the images determined by various sensors.
- Space Science and Technology: Planet surface shapes and planetary settlements are used to group new galaxies and group stars according to their positions.
- Text Mining: It is used to obtain meaningful relationships between very large and meaningless text heaps.
- Scientific, Engineering and Health Care Data: Today, scientific data have become more complicated than job site data.

2.5.Data Mining Process

Data mining techniques and data discovery process consists of the following steps.

1. Collection of Data
2. Cleaning of Data
3. Integration of Data
4. Converting Data
5. Data Mining
6. Pattern Evaluation
7. Information Presentation (Han & Kamber, Data mining: concepts and techniques (the Morgan Kaufmann Series in data management systems), 2000).

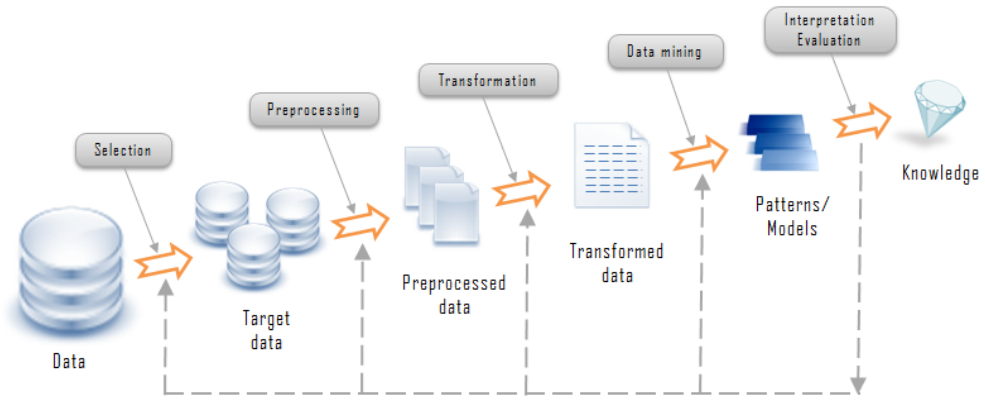

Figure 2.3. Knowledge discovery process

If we want to handle data mining in more detail, we need to follow the steps below.

- Analysis of the problem and understanding of the data
- Data selection
- Data analysis and preparation
- Data reduction and conversion
- Attributes selection
- Reducing the dataset size
- Normalization
- Combining
- Selection of data mining method
- Data mining process
- Visualization
- Evaluation
- Use of information and evaluation of results according to the target

The general experimental procedure, which is based on the narrow and broad meanings of data mining, consists of the 5 steps mentioned in Figure 2.4.

Figure 2.4. Data mining process steps

2.5.1. Determining the Problem and Understanding the Data

Identifying the problem and understanding the data constitute the first stage of the application of data mining. In order to determine the problem and to understand the data in the best way, the following steps can be reached by following the following rules:

- The problem is not clearly defined to provide tangible benefits.
- Possible result should be determined
- It should be determined how to use the result.
- Problems and data should be understood as much as possible.
- The problem should be transformed into model
- Assumptions must be determined.
- The model should be cyclically improved.
- The model should be simplified as much as possible.
- The instability of the model should be defined.
- The uncertainty of the model should be defined.

Depending on the nature of the problem and the data, expert support may be required in the relevant field after these rules are followed and the cooperation with the expert and the data mining workers can be more successful (Kantardzic, 2003).

2.5.2. Data Collection

This stage is about how data is collected. There are two different approaches in the process of data collection. Experiment designed if the process is carried out under expert supervision; if done without expert control, it is called an observational approach. The data used in the same known sample is important in terms of establishing, testing and implementing the model.

2.5.3. Preparation of Data

"In the data mining process, the preparation of the data, where the most time is spent, contributes between 75% and 90% of the success of the process. Preparation of data set from weak or non-existent data is responsible for the failure of the process" (Goldberg, Ohno, Esfarjani, & Kawazoe, 1989). The process of preparing the data is not independent of the other steps of the data mining process. "All the processes performed at each step of data mining together help us obtain a new and more advanced data set" (Kantardzic, 2003). Preparation of data; It consists of missing, wide-spread, conflicting data collected from real life, fusion and conversion of data, reduction of data volume, disruption of data.

2.5.3.1. Clearing data

The data clearing phase consists of filling in missing values, defining incompatible data, verifying conflicting data or removing it from the data set.

At this stage, the following methods can be used for defining and replacing the missing parents:

a) Records with missing values can be discarded from the data set.

b) A general constant can be used to replace the missing values.

c) The mean value of the variable is calculated using all data and this value can be used instead of the missing value.

d) It can be used instead of the missing value by calculating the variable average of the samples of only one class instead of all the data of the variable" (Özkan, 2008).

2.5.3.2. Data fusion and conversion

In order to integrate data obtained from different sources in a fusion process, some procedures have to be done. These processes include metadata, correlation analysis, data conflict detection, and the elimination of semantic mismatch. The conversion of

data is the process of introducing data into a form that will give better results in data mining application (Kantardzic, 2003).

2.5.3.3. Discrete data and conceptual hierarchy

Displacement of data is a condition that is applied on continuous variables. The following techniques are used for disrupting parents; integration, histogram analysis and entropy.

Continuous rendering of discrete data if the numerical based algorithm will be used in the conversion phase of the data; If categorical-based algorithms are to be used, continuous data should be discontinued.

2.5.4. Establishing the Model

At the stage of determining the appropriate model for the defined problem, as many models as possible are established and the models are tried. Therefore, the stages of data preparation and model building are a recurring phase until they reach the model that is considered to be the best.

2.5.5. Model Interpretation

When the model is sufficient to meet the expected objectives, a more broad-based process-based assessment is made. In the evaluation process, whether the model is established correctly, what can be used in the future, different data, such as the expansion of the model includes.

2.6.Data Mining Models

Data mining models are divided into two parts based on estimation and descriptive:

- Models based on estimation: classification, curve fitting, time series
- Descriptive models: bundling, summarizing, correlation rules

2.7.Learning Methods in Data Making

There are 3 different types of learning in data mining:

2.7.1. Supervised Learning

Purpose of supervised learning; learn the relationship between input and output values. Thus, an output value for a new input value can be estimated. Classification algorithms, especially traditional statistical techniques including regression, artificial neural networks (ANN), Decision Tree, Rule Induction, K-means Clustering, K Nearest-Neighbour and support vector machines (SVM) are examples of supervised learning.

2.7.2. Unsupervised Learning

The purpose of unsupervised learning is to find the most appropriate way of showing the input values (data). There are many different approaches. Information maximization, minimum cross entropy, minimum reconstruction error. Data compression, distribution estimation; Various applications of unsupervised learning, such as data source separation, data visualization, can be used as pre-processing for supervised learning. Clustering is the most basic uncontrolled learning method (Herbrich, Graepel, & Obermayer, 1999). Also, self-organized maps is another method of unsupervised learning.

2.7.3. Reinforcement Learning

Supported learning is a learning method that models the way animals learn. The learner takes action and only receives feedback from the changes in the environment due to his / her action. Dynamic resource allocation is an example of learning through game playing and temporal difference learning.

2.8.Data Mining Techniques

Data mining techniques are used successfully in various fields. His main areas of application are marketing, banking, insurance, stock market, telecommunication, health and medicine, industry, science and engineering applications. Some of the data mining techniques are as follows (Işık, 2006);

- Association Analysis
- Classification
- Cluster Analysis
- Identification and Isolation
- Uniform Analysis
- Evolutionary Analysis

Data mining, which is defined as the process of converting large size and fast data to meaningful information as a result of various analyses, is the most widely used technique for classification and clustering problems.

2.8.1. Classification Problem

The classification problem consists of assigning objects to each set of attributes and to the predefined class labels. For each data in the data set, the attribute class and the class label are included. Based on these data, the resulting classifier model derives short and meaningful veins that can be used to classify subsequent records. In classed

classification problems, class labels are available. The aim here is to derive models that fit a specific purpose on objects with a class label (Rastogi & Shim, 2000). The classification model is used for data mining purposes such as Identification, Estimation, Association Analysis, Cluster Analysis (Tan, Steinbach, & Kumar, 2006).

- **Description:** The classification model can serve as an explanatory tool to distinguish the objects of different classes. For example, body temperature for both biologists and others; it would be useful to have a descriptive model that explains that such characteristics as skin, fertility, and a vertebrate describe a vertebrate as a mammal, reptile, bird or fish (Tan, Steinbach, & Kumar, 2006).
- **Prediction:** Classification deals with discrete outputs such as yes/no, mammalian/reptile/ bird. The estimation deals with the outputs that receive continuous values. The estimation is used to find values for unknown continuous variables such as income level, number of votes, future sales forecast when some input data are given (Tan, Steinbach, & Kumar, 2006).
- **Association Analysis:** Association analysis is a model that defines certain types of data relationships. When a product is purchased, the purchase of another product alongside this product gives a rule of association. For example, in a supermarket, shopping is examined and determining which product is purchased with which product is related to the rules of association (Silahtaroğlu, 2008).

2.8.2. Clustering Problem

Clustering analysis is one of the most important areas of data mining; The aim is to collect objects that are similar to each other in a cluster, and those that do not. Uniqueness is determined on the basis of the properties that define objects. Clusters are created by marking groups of similar objects or by having differences with other groups. In terms of machine learning, each cluster represents a hidden pattern, and applied learning is an unsupervised learning. In statistics, multivariate statistical estimation is used in the areas of sound and picture recognition, DNA analysis, geographic information systems and related fields (Silahtaroğlu, 2008).

2.9. Classification Model

Classification is the basic mental that advances through complex events, such as a set of objects defined by high-level data, to small and descriptive units, classes, infrastructures, or parts that serve to better control or express, and by assigning new states to those thousands of classes from existing classes. it is a skill (Bock, 2002). The main algorithms used in the classification can be listed as follows;

- Decision trees,
- Artificial neural networks,
- Evolutionary Algorithms,
- K-closest neighbour,
- Bayesian classifiers,
- Swarm intelligence techniques,
- Dummy-based reasoning,
- Rough cluster approach,
- Fuzzy cluster approach

2.9.1. Decision Trees

Decision trees are the techniques used in data mining to describe the data and to estimate the tree and tree rules to be used in estimating. The decision tree is in a tree structure similar to the flow diagram, each branch representing the result of a test, and leaf nodes representing the classes. If you classify an unknown instance, the attribute values are tested against the decision tree. A new example enters the root section of the tree. This new sample tested in the root is sent to a lower node according to the test result. This process continues until the new sample reaches any leaf node. All Examples that come into a particular leaf of the tree are classified in the same way. There is only one path from root to each leaf. This path defines a rule used to classify samples (Han & Kamber, 2018). Figure 2.5 shows an exemplary decision tree structure.

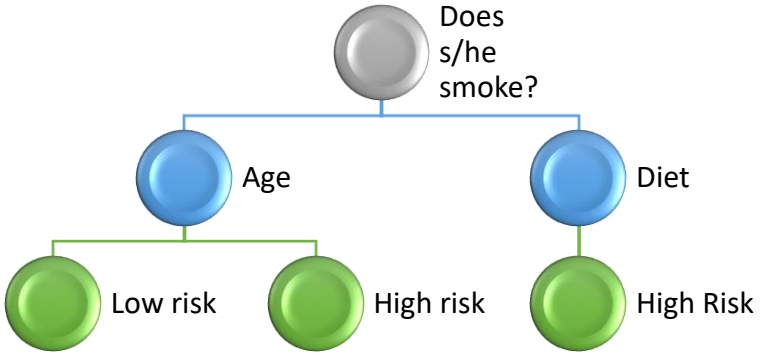

Figure 2.5. An example decision tree structure

A decision tree helps in the data discovery as follows (Murthy, 1998);

- It reduces the volume of data by making the data that is protected as a property and presenting an accurate summary more compacted.
- The discovery tree discovers whether or not it contains well-distributed class objects, so that classes can be interpreted correctly in the concept of significance theory.
- The data in the form of a tree in the form of maps, so you can go back from the branches of the tree to the root of the forecast values can be produced. These values can be used to estimate the output of a new data or query.

Basic decision trees are divided into two groups (Quinlan, 1993). These;

- Classifiers from the machine learning community: 1D3, C4.5, CART.
- Classifiers for large databases: SLIO, SPRINT, SONAR, Rainforest.

One of the biggest advantages is that decision trees can easily be converted into classification rules. Other advantages can be listed as follows;

- The establishment is cheap,
- Effective operation in noisy data,
- Exploring the distinctive features of classes,

- Easy integration into database systems,
- Easy to interpret,
- Good reliability.

2.9.2. Artificial Neural Networks

Another data mining technique used for estimation and classification is artificial neural networks. Neural networks are signal processing systems that attempt to mimic the biological nervous system by presenting a mathematical model of multiple neuronal compositions connected to a network (Haykin & Lippmann, 1994) (Horne, 1993). In ANN, the aim is to mimic the behaviour of the human brain (Setiono, Leow, & Thong, 2000). Neurons are found in the input and output layers and, if any, in hidden layers or layers. When a neuron is identified as significant, the weights associated with this neuron are changed. This means that the neuron will be more effective than other neurons that are at the same level as the neuron itself. Artificial neural networks learn by adjusting the weights between neurons. Figure 2.6 shows the structure of a simple neural network.

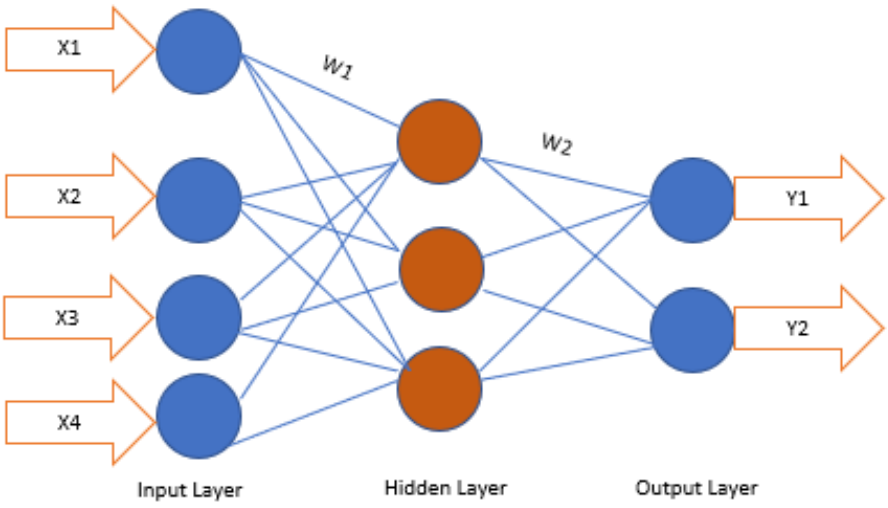

Figure 2.6. Structure of artificial neural networks

The positive and negative aspects of artificial neural networks in terms of data mining are as follows;

- Wide range of application areas,
- They produce better results in complex situations,
- Studying on continuous and categorical data,
- They cannot express their results clearly,
- There is no guarantee that the result is the best result.

2.9.3. Evolutionary Algorithms

Evolutionary algorithms, inspired by the natural evolution process, are widely used in classification and rule extraction in data mining. In contrast to radian-based techniques, evolutionary algorithms intelligently evaluate the performance of multiple candidate solutions and intelligently scan the search space and approach the global best (Michalewicz, 2013).

In general, evolutionary algorithms include all algorithms with population and selection-based genetic operators that generate a new search point in search space. These algorithms are genetic algorithms (GA), genetic programming (GP), evolutionary programming (EP), and evolutionary strategies (ES). These approaches differ from each other according to the operators they use, the models they are applied to, the selection methods and the compliance functions. GA and GP are evolutionary models at the genetic level. In optimization used in evolutionary strategies, the structures of individuals in the population are tried to be optimized. Various behavioural characteristics of individuals are made parametrically and these values are developed at the time of optimization. Evolutionary programming uses the highest level of abstraction, emphasizing the adaptation of the behavioural characteristics of various species (Wong & Leung, 2006).

2.9.4. k-Nearest Neighbour

It is a classifier algorithm based on the distance between points. In this algorithm, the number of points for each sample is taken into account and the process is performed accordingly. k nearest neighbour algorithm is a simpler method than other classifier algorithms. Although it is simple, it has proved its success in many studies and is one

of the most widely used classifier algorithms (Kuncheva, 1995) (Ho, Shu, & Chen, 1995).

When the studies are examined, it is seen that Euclidean distance is used although there are different distance types. The Euclidean distance is an algorithm based on the square root of the sum of the squares of the distance difference between the points. In this distance calculation, the values belonging to the attributes in the training set and the distance to the attributes in the test set are examined (Enas & Choi, 1986).

A separate value is calculated and the sample is labelled with the highest value class label. Another important point here is how many adjacent values are considered. Although k coefficient 3 is generally taken in the literature, this value may vary according to the data set. However, the coefficient n must be single numbers.

Euclidean Distance Calculation:

i = number of attributes;

X = a set of attributes belonging to the test set;

Y = a set of attributes belonging to the training set;

$$d = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_i - y_i)^2} \quad (2.1)$$

2.9.5. Bayes Classifiers

Bayesian classifiers are statistical classifiers. As a possibility given to a particular class, they can predict the possibility of class membership in advance (Han & Kamber, 2018). In addition, Bayes classifiers operating on large databases have very high performance in terms of speed and accuracy. Bayesian classifiers can work effectively with the lowest error rate when the probability distribution of the data is given (Sydow, 1977).

Bayes classifiers are simple classifiers that require a single scan of the data. Therefore, they can achieve high accuracy and speed in large data stacks. Their performances are sufficiently competitive to compete with decision trees and neural networks (Mitra & Acharya, 2005).

The following equations are used to estimate the data for a class for the Naive Bayes classifier:

$$g(x,\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.2}$$

$$P(X \mid C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i}) \tag{2.3}$$

In the equation, μ represents the mean, zrefers to the standard deviation, and x is the attribute.

$P(X|C_i)$ means the probability that the X attribute is in the C_i class.

2.9.6. Swarm Intelligence

Swarm intelligence is defined as the development of a collective intelligence of a group of simple individuals in an autonomous structure (Bonabeau & Theraulaz, 2000). The definition of herd intelligence was first used in 1989 by Gerardo Beni and Jing Wang (Beni & Wang, 1993) in the concept of cellular robotic systems. Flock Intelligence is based on collective behaviour work in decentralized, self-directed systems. Examples of such systems are available in nature. Examples include ant colonies, bird swarms, bacterial moulds, bee colonies, and a swarm of fish.

The concept of herd intelligence includes different algorithms, including ant colony optimization and track optimization. These algorithms are used successfully in optimization. Nowadays, geo-intelligence techniques have been used in the field of data mining and the applications show that these techniques can achieve good results in classification.

2.9.7. Status-Based Reasoning (SBR)

The SBR classifiers first proposed by Schank (Schank, 1982) are example-based classifiers. DTN learns through experience and benefits from the similarities and differences of the problems encountered before. Unlike the closest neighbour classifiers that store educational examples as dots in Euclidean space, the examples or situations stored by the DTN are complex symbolic definitions. Unlike artificial neural networks, status-based reasoning classifiers do not generalize.

New approaches to the DTN include finding a good measure of similarity, developing effective techniques for indexing educational situations and combining solutions.

2.9.8. Rough Set Approach

Initially, the rough set approach proposed by Pawlak (Pawlak, Rough sets-theoretical aspect of reasoning about data, 1991) is used to find structural relationships in noisy data that are not definitive in classification, and is applied to discrete variables. The rough set approach is based on the assumption that, in contrast to the approach in which the cluster is defined only with elements of the cluster and that no additional information is available about the elements of the cluster, there is a need for some information about the space at the beginning to define a cluster. The basis of the rough cluster approach is the inability to distinguish.

The cluster, which forms the basis of information and is a set of the same objects, is called the elementary cluster. Any combination of elementary sets is called a “definite” cluster, otherwise it is referred to as “rough” cluster. Each coarse cluster has elements that cannot be classified as elements of the cluster itself or as elements of the complementary set, which are called "boundary line elements" (Binay, 2002).

The coarse cluster approach is based on the establishment of equivalence classes in an educational data handled. All data samples that make up an equivalence class are indistinguishable. The rough set definition for a given class is estimated by two sets; these are “bottom approach” and “top approach”. The bottom approach is strictly composed of all objects that belong to the class. The upper approach includes all the

objects that belong to the class. The difference between the lower and upper approaches constitutes the boundary zone (Pawlak & Skowron, Rough set rudiments, 1999). Decision rules are created for each limit. In general, in the rough cluster approach, the decision tables are used to show the rules (Han & Kamber, Data mining: concepts and techniques, 2018).

2.9.9. Fuzzy Set Approach

The fuzzy set approach proposed by Zadeh and based on fuzzy logic is concerned with uncertainty. Fuzzy logic provides almost accurate and efficient methods for identifying the behaviour of complex, poorly defined or mathematically not easily analysed systems (Zadeh, 195). In other words, fuzzy logic creates a platform for the use of uncertain and indefinite information. They use an accuracy value of 0-1 rather than a definite distinction between categories.

2.10.Classification Data Mining

2.10.1. Classification Rules

Classification is the process of using a model set of models or functions that define data classes and concepts, in order to use the model obtained to estimate the classes of class label unknown objects [16]. The classification process generally uses consultative learning methods to create a classification model from databases. When the output class is given a known set of examples, the purpose of classification is to discover hidden relationships between variables and classes (Tan, Yu, Heng, & Lee, 2003). Decision limits in classification are established to distinguish samples from different classes (Mitra & Acharya, 2005).

Classification rules are one of the most preferred methods for the representation of the output in data mining applications. This is because it is easy for the user to understand and interpret the rules. A leading part of a rule, such as tests of decision trees in nodes,

includes the test series, the next part of which refers to the classes or classes covered by that rule (Witten & Frank, 2005).

The general form of a rule (also known as a ‘conditional expression’) is exemplified as:

IF antecedent THEN consequence

Two types of rules can be defined, namely characterisation rules and discriminant rules. With characterisation rules, the objective is to find rules that describe the properties of a concept. Characterisation rules have the form:

IF concept THEN characteristic

With discriminant rules, the objective is to find rules that allow the selection (discrimination) of the objects (data records), belonging to a given concept (class), from the rest of the objects (data records or classes). Discriminant rules have the form:

IF characteristic THEN concept

Note that the inverse implication of the characteristic rule is not a discriminant rule.

The concept attribute of a property is defined by a set of variables and their respective values, which are decisive for a given concept. When a dataset of n records, which contains a special dc variable, expressed as a class (decision) variable, is considered, the dc variable splits the records in the data set into parts that are expressed as classes and classify the records, separating them into discrete subsets that are defined by the value of the class variable. The number of discrete subsets here is equal to the number of classes available in the dataset.

2.10.2. Classification Rule Extraction

Classification rule extraction aims to discover a small set of rules in the database to create an accurate classifier (Nel, 2004) (Brieman, Friedman, Olshen, & Stone, Classification and regression trees, 1984). Rule extraction process, information from a data set, symbolic, continuous or discrete information stored in the data set (De Falco, Della Cioppa, & Tarantino, 2002) (Rouwhorst & Engelbrecht, 2000).

The main purpose of the rule is to reveal the hidden information in the data in a clear way, to reveal the previously unknown relationships, to provide the ability to identify and identify (Parpinelli, Lopes, & Freitas, 2002). The rules obtained by classification rule extraction are have characteristics (Nel, 2004);

- Must be understandable,
- Short, simple, clear and clean,
- Define the data correctly
- Rules should not be repeated,
- They should be useful,
- Summarize the data in the data.

Rule extraction from databases not only serves for rule-based classification, but also provides a better perspective to the problem being addressed with the linguistic knowledge discovered (Tan, Yu, & Ang, 2006).

A rule that performs the classification of samples in a dataset is often referred to as the classifier. Classification rule extraction techniques are divided into rule-based methods and non-rule-based methods (Tan , Yu, & Ang, 2006).

- **Rule-based methods:** Rule-based classification methods extract confidential information directly from the data and users can easily understand this information. C4.5 decision tree, decision tables and so on. examples of rule-based methods.

- **Non-rule-based methods:** Non-rule-based classification methods generally give more accurate results than rule-based classification methods, but they

cannot present the information they acquire because they behave like a black box. Generally, non-rule classifiers such as artificial neural networks can achieve very good classification accuracy, but they are not competitive in terms of intelligibility. Support vector machines, artificial neural networks, linear genetic programming are examples of non-rule-based methods.

When creating a rule set to classify a dataset, a local, global, or local-global hybrid algorithm is used in two stages: training and testing. In the first stage, classification rules are learned from a data set composed of classified records. The data set used in this step is called the training data set. This type of learning is called the concept. Therefore, this process is called concept learning (Uran, 2005). In the second step, the rules learned in the first stage are applied to a test dataset and the accuracy of the rules is evaluated. The test dataset also contains class values, such as a training dataset.

2.10.3. Basic Criteria in Evaluation of Classification Methods

It is perhaps the most widely used technical classification in data mining (Larose, 2005). There is a categorical target variable in the classification. The data mining model deals with a set of records that contain information about this target variable, as well as input variables. The following criteria are commonly used to evaluate classification methods (Uran, 2005).

- Predictive accuracy,
- Rule intelligibility,
- Speed and scalability,
- Time required to build the model,
- Time required to use the model,
- Sturdiness,
- Addressing noise and blank values,
- Interest

Predictive accuracy: Determining the predictive accuracy is very important because it will determine which accuracy the classifier will discover previously unseen samples. There are techniques used to estimate holdout, cross-validation, bootstrapping and leave-one-out accuracy (Han & Kamber, 2018).

Resistance and cross-validation are the most common methods for evaluating classifier accuracy. The data discussed in these techniques are randomly divided into pieces. In the method of resistance, the data is divided into two independent parts: training and test sets. Usually, two-thirds of the data is considered as a set of education and the remaining one as a test set. The training set is used to obtain the classifier and the classifier's accuracy is obtained by the test set. Because only a portion of the initial data is used to obtain the classifier, the resistivity technique is a pessimistic technique.

In the k-fold cross-validation technique, data is divided into random size pieces of equal size. Training and testing procedures are repeated k times. For each repetition, a different set of k is used as the test set, and the remaining k-1 is used as a training set. The accuracy estimate is calculated by dividing the total number of correct classifications from the iterations by the number of initial data samples. In general, 10-fold cross-validation is a recommended technique for determining the classifier accuracy due to its low threshold and variance. In the cross-validation technique, a maximum of 10 is used as a result of the detailed tests performed on a large number of datasets with different learning techniques. There is also theoretical knowledge to support it (Witten & Frank, 2005).

The preloading method is a statistical method based on modifiable sampling. In the methods described previously, the training and test are generated without changing the data sets, i.e. the same sample cannot be re-selected once after it has been selected, but it is an example of the data set by modifying to create the basic set of instruction in the boot.

The one-out-and-drop method simply refers to n-fold cross-characterization to indicate the number of samples in the dataset. In this method, each sample is excluded and the learning method is trained with the remaining samples. The test is performed on the excluded sample, the accuracy value will be either positive or negative. The result of the test on each sample, i.e. n is centred on the result of the test, and this mean value indicates the final accuracy value.

Rule comprehensibility: Another important criterion of classification rule extraction is the intelligibility of the model. The understanding of the model means that it can be easily understood and interpreted by users.

Speed and scalability: Speed and scalability are the criteria used in the evaluation of the rules obtained in the classification rule extraction.

Robustness: Robustness is another criterion used in the evaluation of classification methods, but it is not used as frequently as accuracy and clarity. Strength is a measure of the sensitivity to educational data or disturbances in the initial field information.

Interest: In addition to the criteria described above, another criterion that must be taken into account in the classification rule is interest. For end user (Kaur, Wasan, Al-Hegami, & Bhatnagar, 2006);

- If the rules contradict the user's knowledge and expectations (unexpected rules),
- If users can and can do something with the rules (available),
- It is interesting if it adds information(new) to the user's previous information.

2.10.4. Micro and Macro Evaluation of Classification Rules

Evaluation of the rules is of great importance in the classification process. Most of the existing rule learning algorithms are based on the individual rule evaluation criteria. However, the performance of the rule extraction system and the classification process are taken into account in the evaluation of the set of rules. This requires the combination of single rule and rule set evaluation criteria. Rule evaluation criteria are determined by analysing the relationship between the leading and subsequent part of the rule in the 2X2 contingency table. Table 2.1 shows an example contingency table.

Table 2.1. 2x2 Conditionality Table

	Class	Wrong Class	Total
Leading	TP	FP	Leading provided
Wrong Leading	FN	TN	Leading not Provided
Total	Class Provided	Class not Provided	Data Set

When a rule is used to classify a given example of training; four cases occur as positive true (TP), positive false (FP), negative true (TN) and negative false (FN) (Tan , Yu, & Ang, A dual-objective evolutionary algorithm for rules extraction in data mining, 2006). Positive true and negative true, true classifications; positive false and negative false, express false classifications.

- **Positive true (TP):** The rule predicts that the class is positive, and the given class of example is positive.
- **Negative true (TN):** The rule estimates that the class is negative and the class of the given example is negative.
- **Positive false (FP):** The rule estimates that the class is positive, but the class of the given example is negative.
- **Negative False (FN):** The rule predicts that the class is negative, but the class of the given example is positive.

Yao and Zhou (Yao & Zhou, 2008) divided the rule evaluation criteria into two as macro and micro evaluation according to the number of rules assessed. In the second stage, they classified the macro evaluation criteria into two groups as overlapping and non-overlapping rules according to the relationship between rules in a cluster.

Micro evaluation is based on singular rules. Many existing rules of assessment have been proposed for micro-assessment. These criteria are used to determine the stop criterion in rule production and to produce high quality rules for classification purposes. However, evaluation according to the individual rules may result in overlapping results.

Macro evaluation is based on a rule set. Because there are multiple rules for making a decision, it is more evident than micro-evaluation. In macro evaluation, if the object in the treated space provides a maximum rule in a rule set, the rules are rules that do not conflict, if they provide more than one rule, they are conflicting rules. They also divided the conflicting rules into two as consistent and contradictory rules. If an object provides one or more rules in the same class, the rules are consistent, if they provide at least two different classes with multiple rules, they are conflicting rules.

Micro evaluation criteria are designed to demonstrate the power of individual rules. The formulas for the most commonly used micro-performance assessment criteria are given in (2.4) - (2.7).

$$\text{Correctness} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.4}$$

Correctness is a measure of the extent to which the set rule reflects the data set and is found by dividing the total number of samples in the data set by the number of correctly classified samples.

$$\text{Reliability} = \frac{TP}{TP + FP} \tag{2.5}$$

Reliability in classification refers to the ratio of objects that provide both the class and the leading part within the objects that provide the leading part of the rule, and take values from 0 to 1.

$$\text{Support} = \frac{TP}{TP + FN} \tag{2.6}$$

A support criterion is a measure of the acceptability of a rule and takes values from 0 to 1. In classing problems, the support class gives the ratio of objects that provide the leading part in the objects that provide the correct one.

$$\text{Generality} = \frac{TP + FP}{N} \tag{2.7}$$

Generally, the entire data set refers to the part that provides the leading part of the rule. The generality evaluation criterion, such as reliability and support, also takes values from 0 to 1. The number of variables can be used as a measure of complexity in micro rules. This criterion is determined by the number of variables in the leading part of the rule.

Macro evaluation focuses on evaluating the performance of the whole rule inference system, rather than evaluating the performance of each individual rule of the system. Macro performance evaluation criteria can be calculated as follows. Correctness;

$$\text{Correctness} = \frac{\text{Number of samples classified by rule set correctly}}{\text{Number of samples in a dataset}} \quad (2.8)$$

In the macro evaluation, the truth expresses the accuracy of the rule set and is found by dividing the total number of samples in the dataset by the number of instances classified correctly. Reliability;

$$\text{Reliability} = \frac{\text{Number of samples classified by rule set correctly}}{\text{Number of instances classified by the rule set}} \quad (2.9)$$

Reliability returns the proportion of correctly classified objects within the objects classified by the rule set. Support gives the proportion of objects classified correctly by the set of rules within all objects (Equation 1.7);

$$\text{Support} = \frac{\text{Number of samples classfied by rule set correctly}}{\text{Number of samples in a dataset}} \quad (2.10)$$

The generality of a rule set indicates the ratio of objects covered by the rule set within the objects in the data set and is calculated by the following formula;

$$\text{Generality} = \frac{\text{Number of samples covered by rule set}}{\text{Number of instances in a rule set}} \quad (2.11)$$

In macro evaluation, the number of variables as a measure of complexity is calculated by calculating the number of variables in the rule set. The number of rules is expressed by the number of rules in the rule set.

2.10.5. Rule Representation

The first decision in classification rule extraction is how many rules will be coded in a solution (Bojarczuk, Lopes, Freitas, & Michalkiewicz, 2004). There are two approaches to this problem: the Michigan approach and the Pittsburgh approach. In

the Michigan approach, each set of solutions contains a rule, while a number of rules in the Pittsburgh approach make up a set.

The Pittsburgh approach is more suited to classifying the whole set of rules because of its ability to evaluate and, as a result, takes into account the rule interactions. The quality of the rules obtained by this approach is easy to assess.

The Michigan approach has the advantage of low processing time as it contains uncomplicated solutions. However, due to the fact that it treats a rule at once, rule interactions are taken into account, which is the main obstacle to the predictive accuracy of the rule set.

After deciding whether a single rule or multiple rules are to be encoded in the solution array, it must be decided how to encode the arrays. While high-level rule representation can be used, binary encoding is the most common and simplest form of coding (Uran, 2005). One approach to symbolic variables (variables with discrete values) is to use one bit per value. In this case, a variable with N values will be represented by a sub-bit of N bits. For example, if a color variable with {red, white, black, yellow} values is considered, subset k 1010, will represent the colour to be red or black. In this approach, if the bit values of all sub-sequences of a variable are 1, the test result of this variable will return continuously, in other words, this variable will not be part of the leading part of the rule. This is a critical factor considering the generality of the rule. The extraction of some variables creates simpler rules and prevents conflicts.

Another approach for binary coding is to display the index value of a variable in binary form. With this representation, short sub-sequences are obtained, especially for variables containing a large number of elements. In this approach, it is necessary to use a different technique such as trivial bit to extract the variable from the rule.

For continuous (numerical) variables, an intermittent technique is usually required. If the variable values are integers or decimal places, the variable can be converted directly into binary. Although there are very simple cutting techniques that can be used, the interruption process can be very complex and may affect the success of the rules. There are two types of cuts: consultant and non-consultant. In non-consulted interruptions, variable values are discontinued without using class information,

whereas in class-level interruptions, class values are also included in the interruption process and discontinued (Witten & Frank, 2005).

2.10.6. Compliance Function

The second most important decision that must be given in the classification is the fitness function, ie the objective function to be used. The eligibility function for classification rule extraction is often selected according to the purpose of the classification process and the rule-of-use structure used. The maximization of the predictive accuracy is often the first objective, and the predictive accuracy is simply calculated as the ratio of the correctly categorized samples to the samples in the training set. Spears and De Jong (De Jong, Spears, & Gordon, 1994) proposed the following conformity function, which presents a non-linear threshold to correctly classified samples.

$$\text{Eligibility (series i)} = \text{percentage of correctly classified samples}^2 \qquad (2.12)$$

However, this function cannot effectively penalize misclassifications caused by individual rules, which may cause a performance problem. This is especially true when rules are shown by the Michigan approach, because interactions between rules are not taken into account. To mitigate this problem, not only the positive correct and negative truth (correctly classified examples), but also the fit functions that take into account different variations of the positive false and negative false (misclassified examples) values can be used.

Measurement of interest is often more complex. For example, Noda (Romao, Freitas, & Gimenes, 2004) proposed a method that calculates the gain of information from each variable in the leading part of the rule, depending on the weights assigned by users.

2.11.Gene Expression

Although the cells in the human body all contain the same genetic material, the same genes are not active in each cell. The knowledge of which gene is active and which does not exist gives biologists information about how these cells normally function and how the cells will be affected when certain genes are not functioning properly. This activity is called gene expression of the cell (Özcan, 2014).

Each cell contains the same genetic information. However, skin cells, kidney, liver, blood, brain cells are different from each other. These differences are due to the different expression of genes in different cells (Lüleyap, 2008).

In the past, biologists have been able to measure the gene expression data of several genes at the same time, and the gene expression data of thousands of genes can be measured simultaneously with the development of DNA (Deoxyribonucleic acid) microchip technology (Özcan, 2014).

After the gene sequences produced with the human genome project, the new objective; Finding how these genes express their expression, ie removing mRNA profiles, was to show how they relate to other genes, and thus to determine which genes play a role in certain diseases. The type of a cell or phase it is in relates to the mRNA expression of that cell. By studying the expression levels of previously unidentified genes and comparing them with the mRNA expressions of other known genes, it is attempted to obtain information about the functions of those genes (Lüleyap, 2008).

2.12.Microarray Technology

The rapid development of computer technology in parallel with molecular biology brought the two disciplines closer together. Thus, microarray (gene chip), which is one of the end points that biotechnology can reach conceptually, has emerged. The first attempts of the microarray technique were performed by Shalon and Schena (Bal & Budak, 2012).

Microarray technology offers new analytical methods that allow the investigation of many genes at the same time, unlike traditional methods. Although this technique is

especially medicine; biology, microbiology and genetics. Many microarray types are available for different purposes (Şimşek, 2013).

DNA microarray technology; it is a method of simultaneously examining the levels of expression (expression level) and DNA changes of thousands of genes (Shakya, Ruskin, Kerr, Crane, & Becker, 2010).

In traditional methods of molecular biology, the principle "one gene in one experiment gen is generally valid. So; Seeing all of the gene functions in the same study is difficult with conventional methods. New methods, also known as gene chip technology, allow the entire genome to be viewed, which allows simultaneous interaction of thousands of genes. The first attempts of microarray technologies based on the study of thousands of genes in a single study were carried out by Schena in the early 1990s. This technology is a technology that allows for the examination of the expression level of multiple genes at a time and allows thousands of DNA to be analysed at the same time (Bal & Budak, 2012).

DNA microarray technology, monitoring the activity of many genes at the same time; be a fast method; the time-consuming analysis of all results, as well as the advantages of comparing the activities of genes in patients and healthy cells, and categorizing diseases as subgroups; the results may be too complex to interpret; it also has some disadvantages, such as the fact that the results are not sufficiently quantitative and are quite expensive (Liu, Bebu, & Li, 2010).

Measurement of gene expression using microarrays is feasible in many areas of biology and medicine. For example, microarrays can be used to identify disease-related genes by comparing gene expression in a diseased and normal cell. In another study, DNA microarray analysis was performed for major depression in postmortern prefrontal cortex from brain tissue. Gene expression models of patients with major depression and control group were compared and 99 genes were expressed differently in major depression (İpekdal, 2011).

Generate expression profile of DNA microarray (also known as gene chip, DNA chip or biochip). In other words, in order to monitor the expression level of thousands of genes at the same time, a solid surface such as glass, plastic or silicon chip is attached to the array.

Although there are some minor differences in methods and methods of expression, definitions such as DNA array, DNA chip and microchip are used to express similar applications (Bier, et al., 2008).

2.12.1. Production of Microarray and Working Logic

It is a new and powerful technology that has been started to be developed in the mid-90s and allows for the collective examination of gene expression in cells and tissues. All the genes of a microorganism can be placed on a microscope slide scale and the expression levels of thousands of genes can be studied simultaneously in a single experiment. DNA microarray analysis consists of the following steps (Figure 2.7):

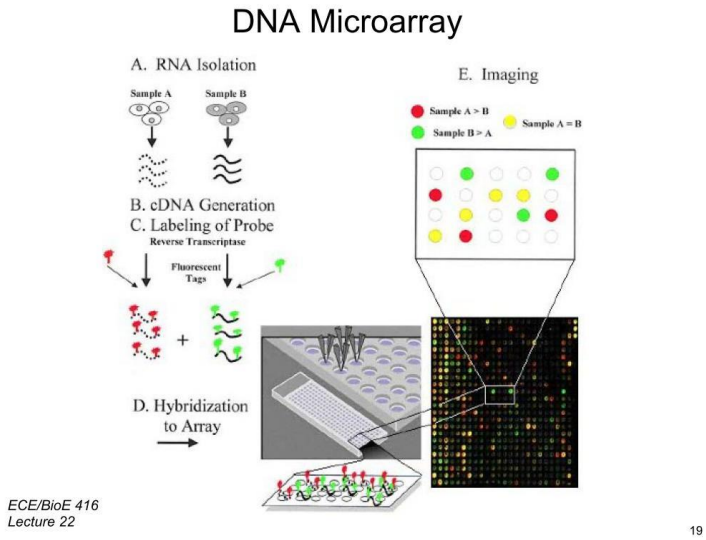

Figure 2.7. DNA microarray process steps (Bal & Budak, 2012)

1. Preparation of microarray
2. Preparation and labelling of samples to be tested
3. Hybridization
4. Washing
5. Stimulation of labels
6. Image scanning / Data processing-analysis

1. **Preparation of Chip:** Microarray as a support material usually glass, plastic silicone, etc. solid surfaces are used. These solid surfaces are treated prior to spotlighting in order to facilitate binding of the nucleic acids by increasing the electrostatic interaction. DNA microarray chips are cDNA (complement DNA) chips. cDNA chips are generated by spotting 500-2000 base pairs of cDNAs or Expressed Sequenced Tagged (EST) clones from the cDNA clone library into the slides by special printers (multi-end mechanical printers, ink-jet printers, etc.). cDNA chips are mostly used for expression analysis. No matter how the chips are obtained, the result is a platform in which there are a large number of homologous DNAs in each probe, where different probes contain different DNA chains and are ready for hybridization with suitably prepared samples (Figure 2.8).

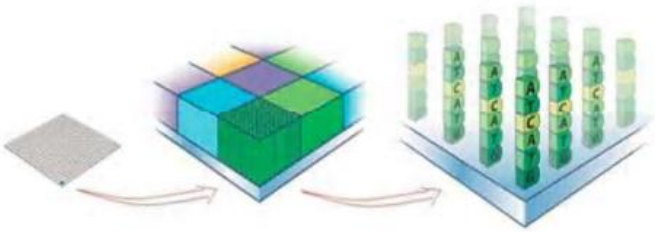

Figure 2.8. The appearance of DNA microarrays (Bal & Budak, 2012)

2. **Preparation and Labelling of Samples:** Sample preparation in the DNA microarray method is an important step, and the target mRNAs isolated from both state samples for the study are translated into cDNA by reverse transcriptase. The resulting cDNAs are labelled with radioactive or fluorescent markers. For radioactive marking, radioisotopes, such as 33P, are generally used for radioactive marking, while cyanine dyes such as Cyanine (Cy3, Cy5) are used for fluorescent marking. Cy3, which gives a green colour and Cy5 which gives red colour, are the most commonly used dyes due to various advantages.

3. **Hybridization:** The mixture of labelled cDNAs is incubated on the microarray to provide hybridization. If there are complementary sequences with microarray probes in the sample, they will hybridize to probes at the end of this process.

4. **Washing:** After the hybridization step, washing is required to remove structures such as nonspecific signal foci and non-probe linked media in the environment. It is an important stage for the correct evaluation of the reaction.
5. **Stimulation of labels:** Hybridization in the microarray, which is hybridized with the samples in the sample and removed from the non-washing-bound material, is made visible. The goal is to make the sequences that hybridize to the microarray appear visible or evaluable (Figure 2.10). The process is carried out using the warning resources appropriate to the nature of the label used and the type of scanners.

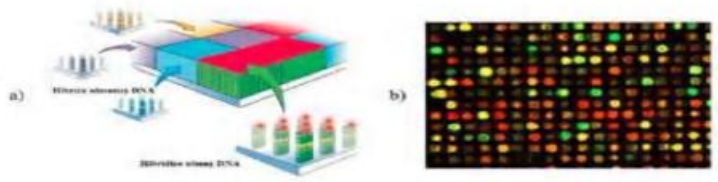

Figure 2.9. Stimulation and fluorescence of labels (Bal & Budak, 2012)

6. **Image Scanning / Data Processing-Analysis:** This is the stage where the fluorescent or radioactive signal on the microarray is collected. For this process, fluorescent signal detectors such as confocal or charge coupled device (CCD) or phosphor imager detectors for radioactive signals are used which measure the light intensity in the microarray spots. All detectors are connected to a special computer, the program of the software and the detectors. A large number of incoming data is evaluated by these software. During scanning, the detectors determine the signal intensity in each microarray probe generated by the hybridized samples. The scanning results are processed by software to make meaningful data. Indexes marked with Cy3 in the example; If the probe is hybridized to the cDNA in the target probe, then that probe will emit red colour if the green colour is hybridized with Cy5-labeled sequences, and the probe will emit yellow if both Cy3 and Cy5 labelled sequences hybridize evenly (Figure 2.11). The processing and analysis of data includes various processes such as normalization, filtering, clustering, and pattern identification (Bal & Budak, 2012).

2.12.2. Usage Areas of Microarrays

- Removal of gene expression profiles
- Polymorphism analysis
- Mutation Analysis
- DNA sequence analysis
- Evolutionary studies
- The presence, development, optimization and clinical evaluation of potential therapeutic agents

The most conspicuous use of the Microarray technique is to measure differences in gene expression. All of the genes transcribed from genomic DNA are called transcriptome or gene expression profiles. Although the genome is cell-to-cell, the gene expression profile is rapidly changing according to the conditions in which the cell is present. Following the changes in the expression levels of genes under various conditions, important clues about the functions of the proteins encoded by these genes can be obtained.

DNA microarray is used extensively to characterize gene expression differentiation in cancer cells. For example, approximately 5500 genes expressed by human lung epithelial cells can be compared with lung cancer tissue genes. Thus, it is possible to obtain information about the genes that play a role in the process of cancerization. Another important role in cancer treatment is the ability to classify cancerous cells according to their gene expression status.

2.12.3. Advantages of Microarray Technology

Generate a general view of gene expression models. A gene expression profile can be determined for a given environment of a particular cell type, and this profile can be compared to this method by different cell types and gene expression profiles under different environmental conditions. It is possible to analyse several thousand genes in a short time and quite practically. Since there is an automation-based system, the likelihood of human error is very low (Bal & Budak, 2012).

2.13.Literature Review

Many studies have been done for microarray analysis in the field of bioinformatics, one of which is the work of D.H Tran et al. (Tran, Ho, Pham, & Satou, 2011). Tran and his colleagues conducted the classification and analysis studies on tumour samples of microRNA (miRNA) expression profiles and thousands of miRNAs can be studied simultaneously thanks to microarray technology. Tran and his colleagues used 223 samples with 151 miRNA attributes in their studies, which were used by Gloub and his colleagues (Lu, et al., 2005). SVM was applied as a classification algorithm and the samples were divided into 2 classes (tumour-normal). As a result, the values in Table 2.2. were obtained.

Table 2.2. Accuracy, sensitivity and AUC values of classification algorithms applied by Tran et al.

SVM Kernel Function	Accuracy	Sensitivity	AUC
RBF	0,92	0,98	0,98
Linear	0,95	0,95	0,97
Polynomial	0,93	0,95	0,96

In another study, X.Fan et al. (Huang, Fang, & Fan, 2010) applied the microarray analysis in the liver, hepatitis, colon, leukaemia and lymphatic cancer cells. In the study in which hepatotox data set was used, classification was performed using 318 samples with 20500 gene attribute (Lobenhofer, et al., 2008). The colon cancer data set (Alon, et al., 1999) consists of 2000 genes and 62 samples, leukaemia dataset consists (Golub, et al., 1999) of 7129 genes and 72 samples, and lymphatic cancer data set (Alizadeh, et al., 2000) consists of 4026 genes and 96 samples. The results of the study are as in Table 2.3.

Table 2.3. Accuracy of classification algorithms applied to hepatatox, colon, leukaemia and lymph cells by X. Fan et al.

Applied Classification Algorithm	Hepatatoks	Colon Cancer	Lymph Cancer	Leukaemia
Decision Trees	0,901	0,817	0,943	0,966
SVM	0,886	0,813	0,961	0,945
KNN	0,873	0,795	0,892	0,926

D. Liu and his friends (Chen, Yang, Liu, & Liu, 2011) were divided into 2 classes with 999 samples of 699 samples with his research in the Wisconsin Diagnostic Breast Cancer (WDBC) cell database (Uci Machine Learning Reporsitory, 2018). SVM was used as the classification algorithm and the training and test data were divided into 50-50%, 70-30%, 80-20%, respectively. The results of the study are as in Table 2.4.

Table 2.4. The accuracy of the SVM algorithm applied to the WDBC breast cancer cells by D.Liu et al.

Applied Classification Algorithm	Training-Test Cluster Percentages	Accuracy Values
SVM	%50-50	0,95
	%70-30	0,96
	%80-20	0,96

C. Chakraborty and colleagues (Krishnan, Banerjee, Chakraborty, & Ajoy, 2010) have also analysed the microarray of WDBC cells in their studies. The aim of this study was to determine the high accuracy of SVM in breast cancer. Two types of data sets were used in the study. The first of these data sets consists of 699 examples, with 9 attributes and 2 classes. In the other dataset, 569 sample is divided into 2 classes with 10 attributes. While SVM algorithm was applied to these data sets, the polynomial and Gaussian functions of SVM were used. The results obtained are as in Table 2.5.

Table 2.5. C. Chakraborty et al. WDBC on breast cancer cells SVM algorithm applied with polynomial and Gauss kernels sensitivity and specificity values

	Dataset-1		Dataset-2	
	Polynomial	Gauss	Polynomial	Gauss
Sensitivity	0,9775	1	0,9269	0,945
Specificity	0,9762	0,9879	0,9256	0,9298

In another study, M. Acı and M. Avcı (Acı & Avcı, 2011) tried the K-ENK algorithm in the WDBC cells database. As previously noted, the WDBC dataset consists of 699 examples of 9 attributes and 2 classes. The distance values which are important in K-ENK algorithm are taken as Manhattan, Euclid and Minkowski respectively. The values in Table 2.6 show the results of the study.

Table 2.6. M. Acı and M. Avcı's false sample numbers classified in WDBC cells by K-ENK algorithm for different distances

*	Manhattan	Euclidean	Minkowski
k=1	14	10	17
k=2	16	17	15
k=3	15	18	15
k=4	19	16	21
k=5	17	19	20
* The above values show incorrectly classified sample numbers.			

B. Han et al. (Han, Li, Chen, Zhu, & Dai, 2011) have studied microarray analysis studies in leukaemia, brain tumours, colon and prostate cancer cells. Sample datasets attribute selection methods, respectively 1, 5, 10, 20, 50, 100 it is divided into genes and then classifying algorithms were used. SVM, KNN, RF, NB were used as classification algorithms. Table 2.7 shows the classification accuracy values for 5, 10 and 20 attributes of colon cancer cell.

Table 2.7. Accuracy values of different classification algorithms applied in colon cancer cells using 5, 10, 20 attributes by B. Han

Applied Classification Algorithm	Number of genes		
	5	10	20
SVM	0,840	0,863	0,851
KNN	0,853	0,903	0,886
RF	0,846	0,866	0,886
NB	0,887	0,911	0,866

In the study (Li & Liu, 2010) conducted by D. Li et al., Data sets in the University of California's machine learning data warehouse were used. Data sets were used as Echocardiogram, WDBC, BUPA liver and PIMA diabetes data (Uci Machine Learning Reporsitory, 2018). As a classification algorithm, Gauss and polynomial kernels and SVM algorithm was applied. Data sets are divided into 5, 10, 20, 30, 50, 100 educational examples, respectively. Table 2.8 shows the results of the studies.

Table 2.8. Accuracy values of the SVM algorithm in which Echocardiogram, WDBC, Bupa and Pima are applied using Gauss and polynomial kernels on data sets by D. Li and friends.

Echocardiogram Dataset				
SVM Kernel Function	Education Sample Numbers			
	5	10	20	30
SVM-Gauss	67,17	74,33	84,17	86,50
SVM-Polynomial	64,50	67,67	73,00	74,17
WDBC Dataset				
SVM Kernel Function	Education Sample Numbers			
	5	10	20	30
SVM-Gauss	52,73	57,60	62,90	74,10
SVM-Polynomial	52,73	53,56	57,88	55,38
BUPA Liver Dataset				
SVM Kernel Function	Education Sample Numbers			
	5	10	20	30
SVM-Gauss	49,32	51,53	54,57	56,35
SVM-Polynomial	51,27	51,92	54,50	56,42
PIMA Diabetes Dataset				
SVM Kernel Function	Education Sample Numbers			
	5	10	20	30
SVM-Gauss	55,00	59,22	60,87	60,03
SVM-Polynomial	55,03	60,05	61,68	61,68

In the microarray based cancer classification study, X.Wang and O.Gotoh (Wang & Gotoh, 2010) tried different classification algorithms on the nervous system, colon, lung, prostate, breast cancer cells and leukaemia samples. SVM, NB, KNN and KA were applied as classification algorithms. Table 2.9 shows the classification results of colon cancer with 5, 10, 20, 50 and 100 attributes.

Table 2.9. Accuracy values of different classification algorithms used by Gotoh using a set of 5, 10, 20, 50 and 100 attributes by X.Wang and O.Gotoh

Colon Cancer Dataset					
Applied Classification Algorithm	Number of genes				
	5	10	20	50	100
SVM	59,68	82,26	88,71	83,87	87,10
NB	75,81	80,65	79,03	77,42	74,19
KNN	67,74	82,26	85,48	85,48	88,71
KA	61,29	79,03	83,87	74,19	88,71

CHAPTER 3

METHOD

In the study, the open-source Java-based Weka program program for the classification of low-dimensional microarray gene expression data was used so that the dimensions of the microarray gene expression data were reduced by using the attribute selection methods. Thanks to this program, statistical and artificial intelligence-based methods were used to classify microarray data and their performance on microarray gene data were compared with each other.

Thus, artificial intelligence-based classification methods, which are among the new methods used in the literature, have been used in this study.

3.1. Weka Program

Weka is an open-source data mining program developed by the University of Waikato in New Zealand, which incorporates machine learning algorithms, has a functional graphical interface, and is developed with the Java programming language (Witten & Frank, 2005). Weka various data pre-processing, classification; Includes regression, association rules, clustering and visualization tools. Algorithms can be applied to the data set directly or by calling from the Java code (Hall, et al., 2009) (Patterson, Liu, Turner, Concepcion, & Lynch, 2008). It is also suitable for developing new machine learning algorithms.

Weka supports all steps of data mining, such as the processing of raw data, statistical evaluation of learning methods on data, and visual monitoring of raw data and the model extracted from raw data. It includes many data pre-processing filters as it has a wide range of learning algorithms. Weka hosts 4 basic applications called Explorer, Experimenter, Knowledge Flow and Simple CLI.

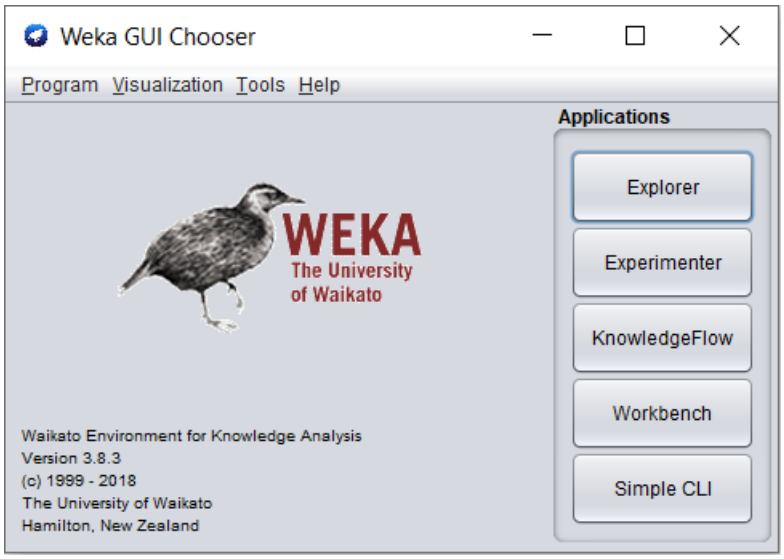

Figure 3.1. Weka user interface

When the program is executed, the user interface in Figure 3.1 is displayed. This interface screen contains the main menu consisting of the “Program”, “Visualization”, “Tools” and “Help” menus and the "Applications" sections, which are “Explorer”, “Experimenter”, “Knowledge Flow” and “Simple CLI”. The Explorer option in the “Applications” section contains a general graphical user interface that contains instructions that can be made on existing data. The Experimenter option is a user interface that allows one or more algorithms to be applied and monitored on one or more datasets. The "Knowledge Flow" option works like Simulink in Matlab, or the Explorer Window, which has drag-and-drop functionality like LabVIEW to National Instruments. User can use “Explorer” or “Knowledge Flow” options depending on preference. The last option, “Simple CLI”, allows you to process through the command screen!

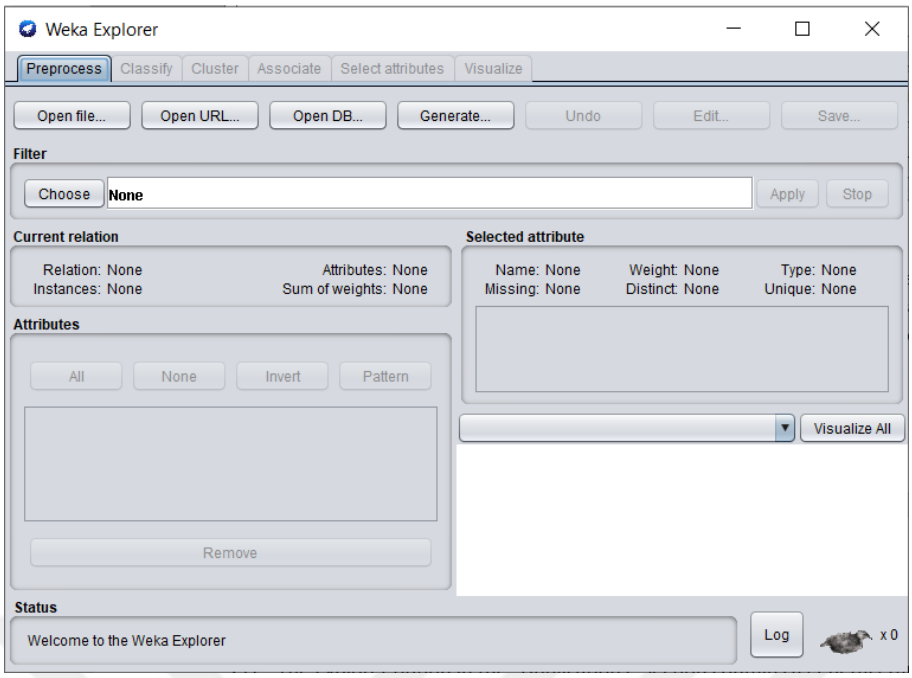

Figure 3.2. Tabs in Explorer

Figure 3.2 shows the tabs in the "Explorer" option of the Weka data mining program. Under this tab, there are menus such as association, classification, clustering, property visualization, selection and also provides information about the attributes and classes of the data on this page.

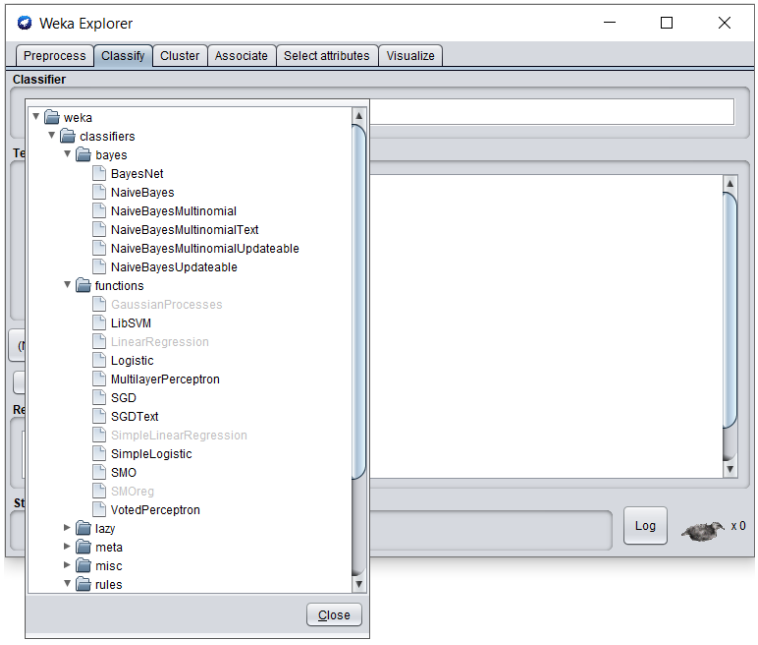

Figure 3.3. Weka "Classify" tab

In Figure 3.3, in the "Classify" tab, a user interface screen with various classification algorithms is displayed. One of the many classifier algorithms such as "Bayes", "SMO", "lBk", "J48" is selected under this interface.

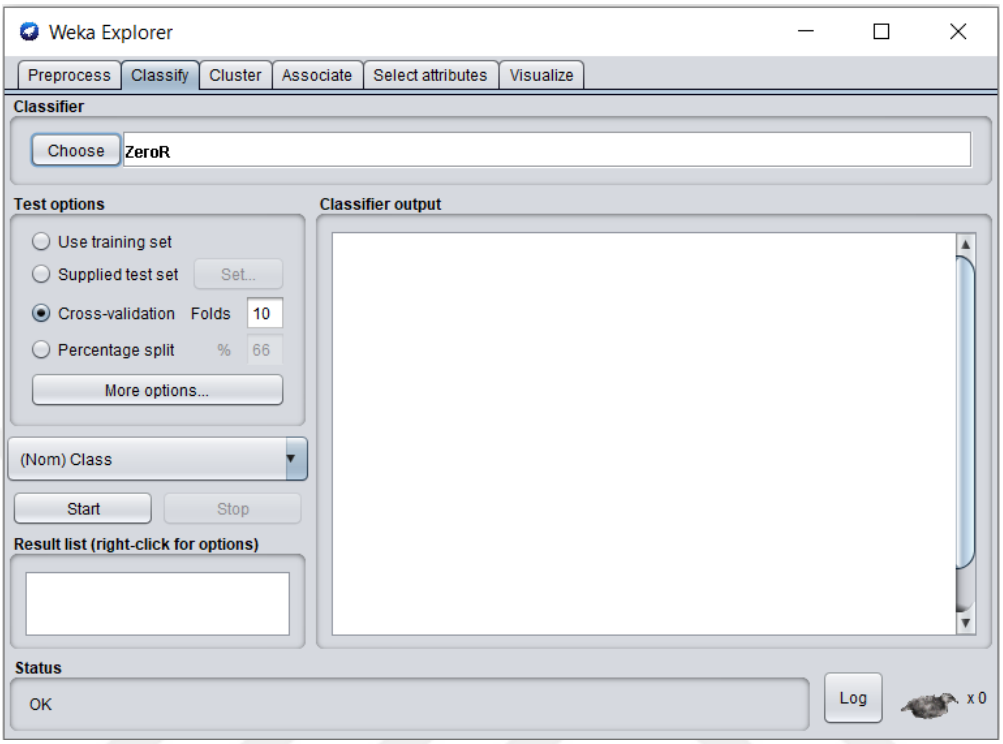

Figure 3.4. "Test Options" title

In Figure 3.4, there are options on how to use the training set and test set in the "Test Options" section.

- **Use training set:** Assesses the classifier on how well it predicts the class of the occasions it was prepared on.
- **Supplied test set:** Assesses the classifier on how well it predicts the class of a lot of examples stacked from a record. Tapping on the 'Set… ' secure carries an exchange enabling you to pick the document to test on.
- **Cross-validation:** Assesses the classifier by cross-approval, utilizing the quantity of folds that are entered in the 'Folds' content field.
- **Percentage split:** Assesses the classifier on how well it predicts a specific level of the information, which is waited for testing. The measure of information held out relies upon the worth entered in the '%' field (Svetlana)

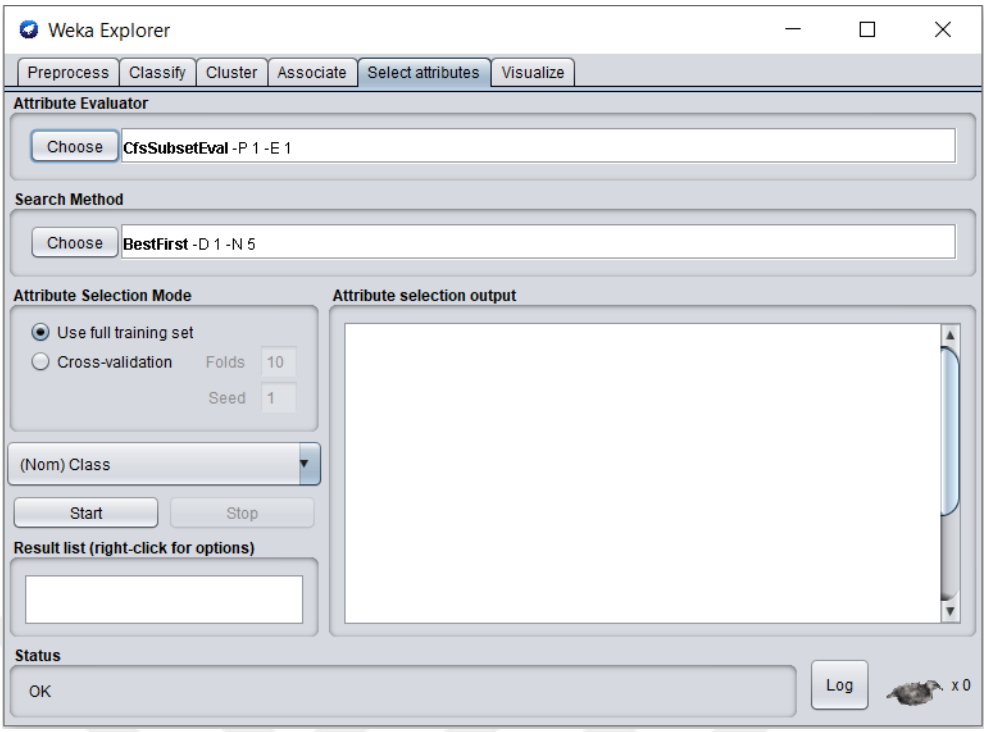

Figure 3.5. The "Select Attributes" tab 1

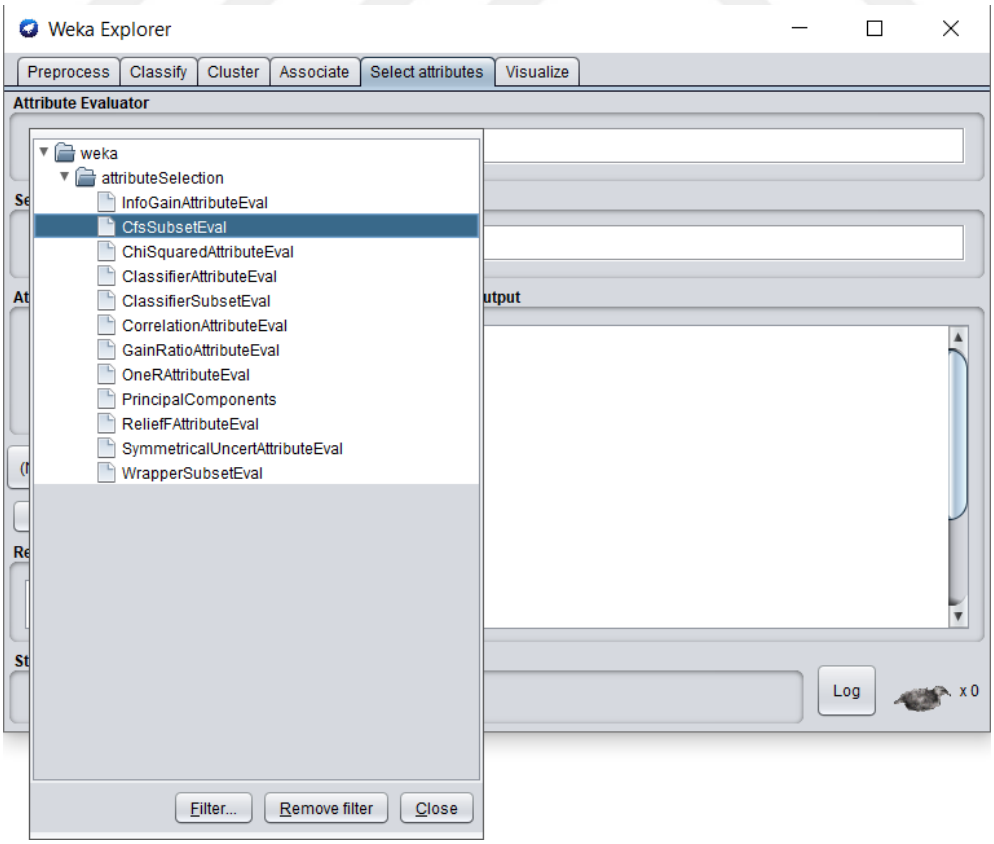

Figure 3.6. The "Select Attributes" tab 2

In Figure 3.5 and Figure 3.6, dimension reduction is performed using one of the attribute selection methods in the "Select attributes" tab.

3.2. Attribute Selection

The attribute selection is a subset of attributes that has been reduced from all attributes, and possibly have better classification performance. Qualification selection has proved to be a critical requirement when it comes to achieving accurate and reliable cancer classification results using mRNA information. WrapperSubsetEval, CfsSubsetEval (correlation based) and ChiSquareSubsetEval algorithms have been used in this study.

- **WrapperSubsetEval:** Assesses property sets by utilizing a learning plan. Cross approval is utilized to assess the exactness of the learning plan for a lot of traits.
- **CfsSubsetEval:** Assesses the value of a subset of properties by considering the individual prescient capacity of each component alongside the level of excess between them. Subsets of highlights that are profoundly associated with the class while having low intercorrelation are liked.
- **ChiSquareSubsetEval:** Assesses the value of a quality by registering the estimation of the chi-squared measurement as for the class.

3.3. Arff File Format

An ARFF (Attribute-Relation File Format) file is an ASCII content document that portrays a rundown of occurrences sharing a lot of characteristics. ARFF records were created by the Machine Learning Project at the Department of Computer Science of The University of Waikato for use with the Weka AI programming.

ARFF files have two distinct sections. The first section is the Header information, which is followed the Data information. (Weka, 2018)

The Header of the ARFF record contains the name of the connection, a rundown of the characteristics (the sections in the information), and their sorts. A model header on the standard IRIS dataset looks like as Figure 3.7

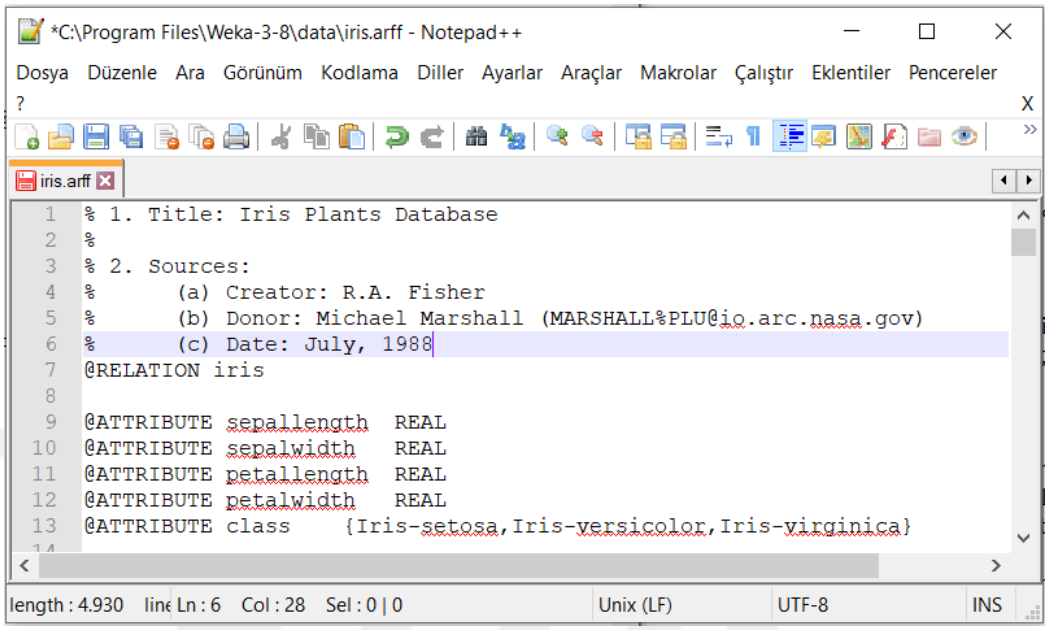

Figure 3.7. The header information of standard IRIS dataset

The Data of the ARFF file looks like as Figure 3.8

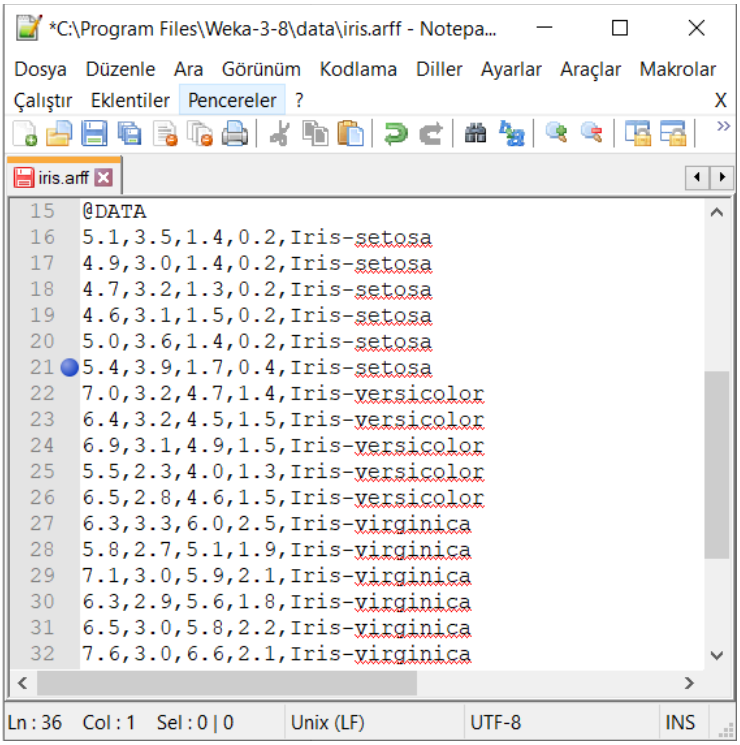

Figure 3.8. The data information of standard IRIS dataset

Lines that begin with a % are comments. The @RELATION, @ATTRIBUTE and @DATA declarations are case insensitive.

3.3.1.The ARFF Header Section

The ARFF Header area of the document contains the connection revelation and trait presentations.

@relation Declaration: The relation name is defined as the first line in the ARFF file. The format is:

@relation <relation-name>

where <relation-name> is a string. The string must be quoted if the name includes spaces.

The @attribute Declarations: Attribute declarations take the form of an ordered sequence of @attribute statements. Each attribute in the data set has its own @attribute statement which uniquely defines the name of that attribute and its data type. The order the attributes are declared indicates the column position in the data section of the file. For example, if an attribute is the third one declared then Weka expects that all that attributes values will be found in the third comma delimited column. The format for the @attribute statement is:

@attribute <attribute-name> <datatype>

where the <attribute-name> must start with an alphabetic character. If spaces are to be included in the name then the entire name must be quoted. (Weka, 2018)

3.3.2.ARFF Data Section

The ARFF Data section of the file contains the data declaration line and the actual instance lines.

@data Declaration: The @data declaration is a single line denoting the start of the data segment in the file. The format is:

@data

Instance data: Each instance is represented on a single line, with carriage returns denoting the end of the instance. Attribute values for each instance are delimited by commas. They must appear in the order that they were declared in the header section (i.e. the data corresponding to the nth @attribute declaration is always the nth field of the attribute).

3.4. Classification Methods

3.4.1.Naive Bayes (NB)

It is a classification system dependent on Bayes' Theorem with a suspicion of autonomy among indicators. In basic terms, a Naive Bayes classifier expect that the nearness of a specific component in a class is irrelevant to the nearness of some other element. For instance, an organic product might be viewed as an apple in the event that it is red, round, and around 3 creeps in width. Regardless of whether these highlights rely upon one another or upon the presence of different highlights, these properties autonomously add to the likelihood that this organic product is an apple and that is the reason it is known as 'Naive'

Naive Bayes model is anything but difficult to manufacture and especially valuable for exceptionally enormous informational collections. Alongside straightforwardness, Naive Bayes is known to beat even exceptionally refined order strategies. (Naive Bayes, 2018)

Bayes theorem provides a way of calculating posterior probability $P(c \mid x)$ from $P(c)$, $P(x)$ and $P(x \mid c)$. Look at the equation below:

$$\underbrace{P(c \mid x)}_{\text{Posterior Probability}} = \frac{\overbrace{P(x \mid c)}^{\text{Likelihood}} \quad \overbrace{P(c)}^{\text{Class Prior Probability}}}{\underbrace{P(x)}_{\text{Predictor Prior Probability}}} \tag{3.1}$$

$$P(c \mid X) = P(x_1 \mid c) * P(x_2 \mid c) * \ldots * P(x_n \mid c) * P(c)$$

Above,

- $P(c \mid x)$ is the posterior probability of class (c, target) given predictor (x, attributes).
- $P(c)$ is the prior probability of class.
- $P(x \mid c)$ is the probability which is the likelihood of indicator given class.
- $P(x)$ is the prior probability of predictor. (Naive Bayes, 2018)

How Naive Bayes algorithm works?

Let's understand it using an example. Beneath I have a preparation informational index of climate and relating target variable 'Play' (recommending conceivable outcomes of playing). Presently, we have to group whether players will play or not founded on climate condition. How about we pursue the underneath ventures to perform it.

Step 1: Convert the informational index into a recurrence table

Step 2: Make Likelihood table by finding the probabilities like Overcast likelihood = 0.29 and likelihood of playing is 0.64..

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

Figure 3.9. Naive Bayes likelihood table

Step 3: Presently, utilize Naive Bayesian condition to compute the back likelihood for each class. The class with the most noteworthy back likelihood is the result of forecast.

Problem: Players will play if climate is radiant. Is this announcement is right?

We can solve it using above discussed method of posterior probability.

$P(Yes \mid Sunny) = P(Sunny \mid Yes) * P(Yes) / P(Sunny)$

Here we have $P(Sunny \mid Yes) = 3/) = 0,33$, $P(Sunny) = 5/14 = 0,36$, $P(Yes) = 9/14 = 0,64$.

Now, $P(Yes \mid Sunny) = 0,33 * 0,64 / 0,36 = 0,60$, which has higher probability.

Naive Bayes utilizes a comparable technique to foresee the likelihood of various class dependent on different qualities. This calculation is for the most part utilized in content characterization and with issues having different classes.

3.4.2.Support Vector Machines (SVM)

Support Vector Machine (SVM) is a supervised machine learning algorithm which can utilized for both characterization or relapse difficulties. Be that as it may, it is for the most part utilized in characterization issues. In this calculation, we plot every datum thing as a point in n-dimensional space (where n is number of highlights you have) with the estimation of each component being the estimation of a specific arrange. At that point, we perform order by finding the hyper-plane that separate the two classes great (look at the below figures).

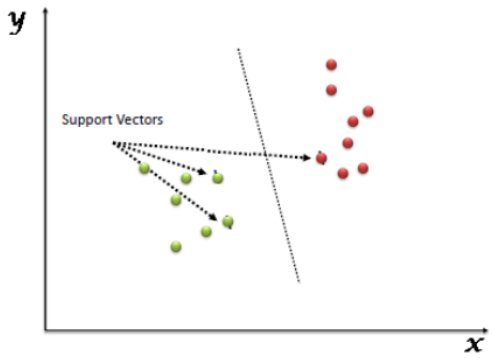

Figure 3.10. Support Vector Machine

Support Vectors are basically the co-ordinates of individual perception. Support Vector Machine is a boondocks which best isolates the two classes (hyper-plane/line).

How does Support Vector Machine work?

Above, we got acquainted with the way toward isolating the two classes with a hyper-plane. Presently the consuming inquiry is "How might we recognize the privilege hyper-plane?".

Let’s understand:

- **Identify the right hyper-plane (Scenario-1):** Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify star and circle.

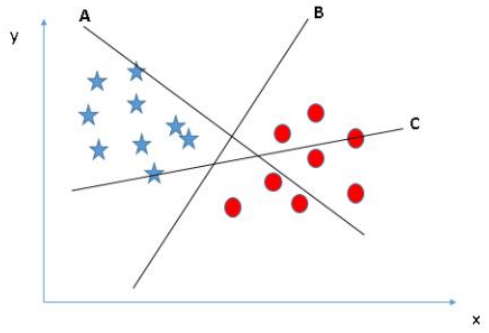

Figure 3.11. Support Vector Machine Scenario 1

You need to remember a thumb rule to identify the right hyper-plane: “Select the hyper-plane which segregates the two classes better”. In this scenario, hyper-plane “B” has excellently performed this job.

- **Identify the right hyper-plane (Scenario-2):** Here, we have three hyper-planes (A, B and C) and all are segregating the classes well. Now, how can we identify the right hyper-plane?

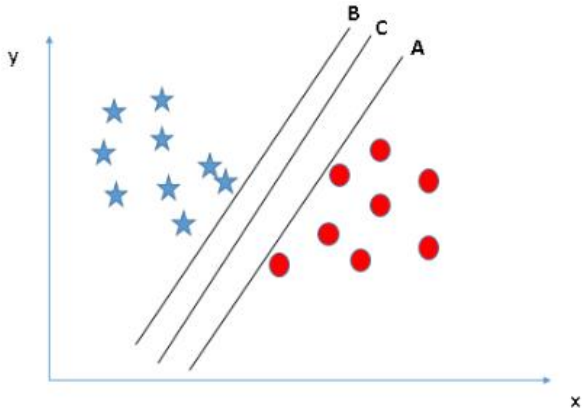

Figure 3.12. Support Vector Machine Scenario 2-a

Here, expanding the separations between closest information point (either class) and hyper-plane will assist us with deciding the privilege hyper-plane. This separation is called as Margin. Let’s look at the Figure 3.13:

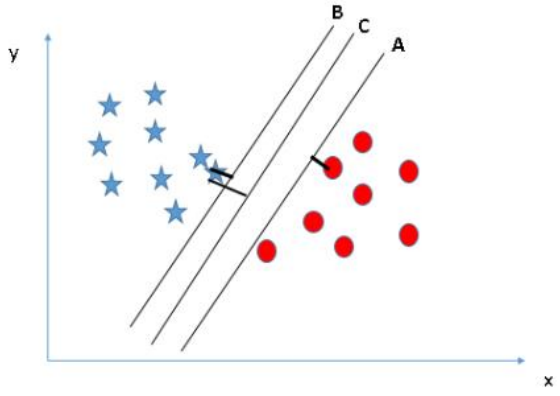

Figure 3.13. Support Vector Machine Scenario 2-b

Above, you can see that the margin for hyper-plane C is high when contrasted with both An and B. Thus, we name the privilege hyper-plane as C. Another lightning purpose behind choosing the hyper-plane with higher edge is strength. On the off chance that we select a hyper-plane having low edge, at that point there is high chance of miss-classification

- **Identify the right hyper-plane (Scenario-3):** Hint: Use the rules as discussed in previous section to identify the right hyper-plane

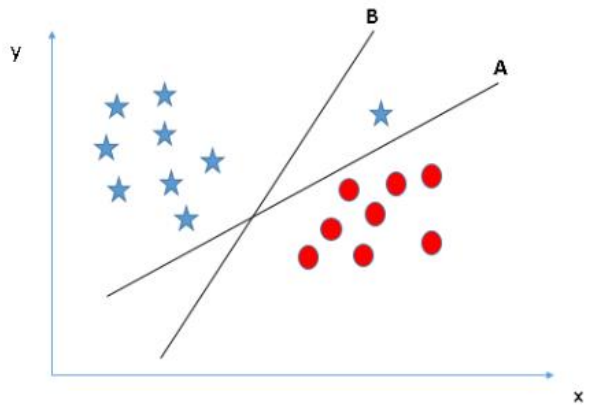

Figure 3.14. Support Vector Machine Scenario 3

Some of you may have chosen the hyper-plane B as it has higher edge contrasted with A. Be that as it may, here is the trick, SVM chooses the hyper-plane which orders the classes precisely preceding augmenting edge. Here, hyper-plane B has an arrangement mistake and A has ordered all effectively. Along these lines, the privilege hyper-plane is A.

- **Can we classify two classes (Scenario-4)?** : Below, we can't isolate the two classes utilizing a straight line, as one of star lies in the region of other(circle) class as an anomaly.

Figure 3.15. Support Vector Machine Scenario 4-a

As we have just referenced, one star at opposite end resembles an anomaly for star class. SVM has an element to disregard exceptions and discover the hyper-plane that has most extreme edge. Thus, we can say, SVM is vigorous to anomalies.

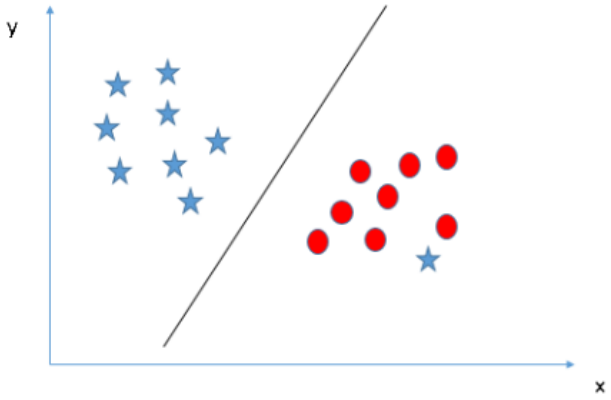

Figure 3.16. Support Vector Machine Scenario 4-b

- **Find the hyper-plane to segregate to classes (Scenario-5):** In the situation beneath, we can't have direct hyper-plane between the two classes, so how does SVM order these two classes? Till now, we have just taken a gander at the linear hyper-plane.

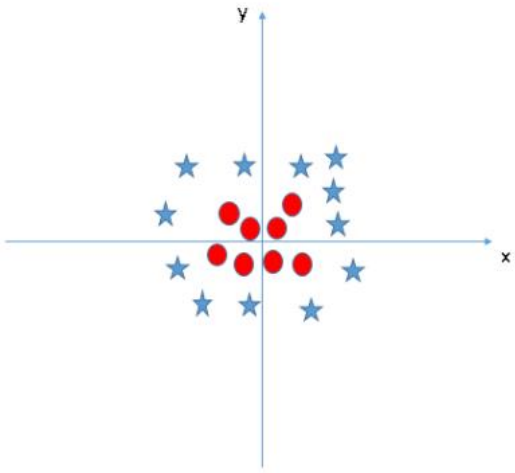

Figure 3.17. Support Vector Machine Scenario 5-a

VM can take care of this issue. Effectively! It takes care of this issue by presenting extra element. Here, we will include another element $z = x^2 + y^2$. Presently, how about we plot the information focuses on pivot x and z:

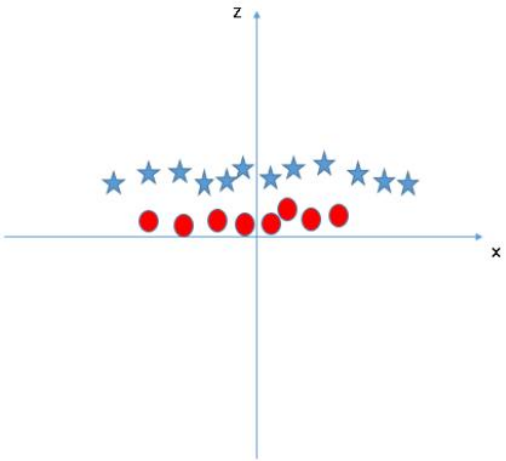

Figure 3.18. Support Vector Machine Scenario 5-b

In above plot, points to consider are:

- All values for z would be positive always because z is the squared sum of both x and y,
- In the original plot, red circles appear close to the origin of x and y axes, leading to lower value of z and star relatively away from the origin result to higher value of z.

In SVM, it is anything but difficult to have a straight hyper-plane between these two classes. Be that as it may, another consuming inquiry which emerges is, should we have to add this element physically to have a hyper-plane. No, SVM has a method called the part trap. These are capacities which takes low dimensional info space and change it to a higher dimensional space for example it changes over not detachable issue to distinguishable issue, these capacities are called portions. It is for the most part valuable in non-direct detachment issue. Simply put, it does some extremely complex data transformations, then find out the process to separate the data based on the labels or outputs you've defined. (Understaing Support Vector Machine, 2018).

When we look at the hyper-plane in original input space it looks like a circle:

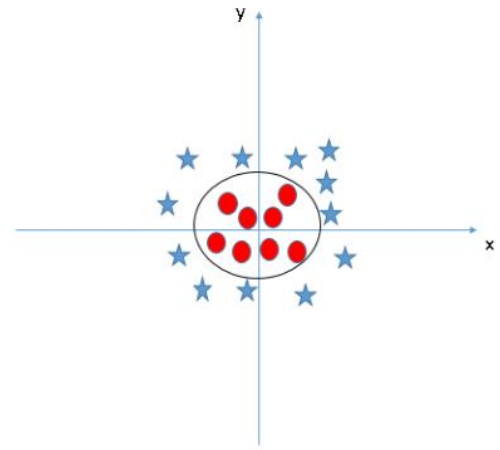

Figure 3.19. Support Vector Machine Scenario 5-c

3.4.3.Bagging

Bagging is a method for generating multiple versions of the predictor and uses them to obtain a clustered estimator. Clustering averages these versions while estimating the numerical output and applies the plurality vote principle when predicting the class. Bagging can increase accuracy if the mix of the set of instruction leads to significant changes in the configured predictor (Brieiman, 1996).

3.4.4.One-R

One-R or "One Rule" is a simple algorithm proposed by R. C. Holt. This algorithm generates a rule for each feature in the training data, and then the rule with the smallest error rate is selected according to the One-R (Novakovic, Minic, & Veljovic, 2010).

3.4.5.Decision Tree

Decision trees are a popular and powerful tool used for classification and prediction purposes. Decision trees provide a convenient alternative for viewing and managing large sets of business rules, allowing them be translated in a way that allows humans to understand them and apply the rules constraints in a database so that records falling into a specific category are sure to be retrieved. (What is a Decision Tree, 2018)

Decision trees generally consist of the following four steps:

1. Structuring the issue as a tree by making end hubs of the branches, which are related with a particular way or situation along the tree.
2. Assigning subject probabilities to each represented event on the tree.
3. Assigning adjustments for outcomes. This could be a particular dollar sum or utility worth that is related with a specific situation.
4. Identifying and choosing the proper course(s) of activity dependent on investigations.

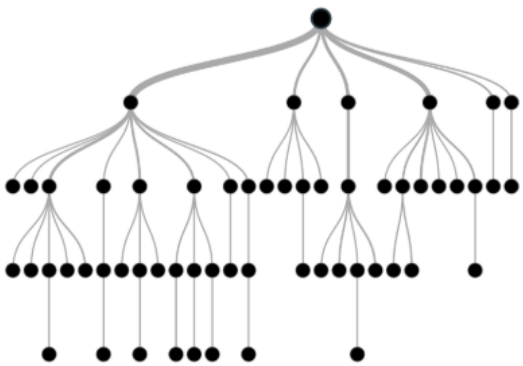

Figure 3.20. Decision Tree

3.4.6.Fuzzy k- Nearest Neighbour (Fuzzy k-NN)

Fuzzy logic concept was first introduced in 1965 by Prof. Lotfı Asker Zadeh. This logic is based on the principle of mathematical modelling of computer-based expressions that people use in their daily life. In classical methods, the sharp transitions of decisions made with 1 or 0 are softened with fuzzy logic and can be expressed with interim values.

The fuzzy k-closest neighbouring method is a classification algorithm such as the k-nearest neighbour method but it is separated from the k-nearest neighbour method by the expression of its results. The fuzzy k nearest neighbour algorithm assigns class membership to a sample vector instead of assigning the vector to a specific class. The fact that an element belongs to a cluster or a class belongs to the concept of a classic set (membership = 1) or is not (membership = 0).

In fact, it is not clear whether an element belongs to a cluster completely or not. So this element must have a degree of belonging (membership value) for that cluster or class. This membership value can be infinite between 0 and 1. In fuzzy algorithms, while classifying the sample to be tested, information is given on how much it belongs to that class, as well as to identify its class. This information, for example, is the membership value for that class. The advantage of the fuzzy k-closest neighbouring method over the k-closest neighbouring method is that the fuzzy k-closest neighbouring method contains more information.

In the fuzzy k-closest neighbouring method, the membership values specified for the sample to be tested provide a level of assurance for the resulting classification. For example; If we consider that there are two classes, we assume that the membership values for a sample to be tested are calculated with the membership value of 0.89 to a class and the membership value with the value of 0.11 to the other class; We can easily decide by looking at the number of membership values in the decision that the class with 0.89 membership value is the class of the sample to be tested.

For a different example; If we assume that there are three classes, then it may not be possible to make a definitive decision in the classification of the sample to be tested if the membership to be tested is calculated as 0.55 membership value, the membership value of the second class is 0.44 and the membership to the third class is calculated as

0.01 membership value. But we can be sure that it does not belong to the third class. In this case, different methods can be examined by trying the sample to be tested in order to understand the class. because the sample to be tested shows a high degree of membership in both classes.

As a result; The fuzzy k-closest neighbour algorithm is the answer to the question of how much the test sample belongs to a particular class, rather than classifying a test sample unknown to the nearest neighbour algorithm. In the fuzzy k-closest neighbour algorithm, the value of how much it belongs to class 0 is calculated, in addition to the knowledge of belonging to a class for the sample or not. This value is used to classify, for example, (Eren, 2008).

3.4.7.Single Layer Perceptron

The perceptron machine learning algorithm was devised in 1957 by Frank Rosenblatt at Cornell University. It is executed here as a supervised learning algorithm, meaning a desired, or known, output exists. We train the perceptron to do our bidding based on how closely its guesses at each iteration correspond to the known output. We can think of the perceptron as a group of n input neurons that communicate at n synapses with a single output neuron. Each input neuron receives an input x_i, and the affect of this stimulation on the output neuron depends on the strength w_i of the synaptic connection between them. Training the perceptron involves changing these synaptic weights over many iterations to arrive at the set of weights $w_1 ... w_n$ producing an output o that matches our desired output, y.

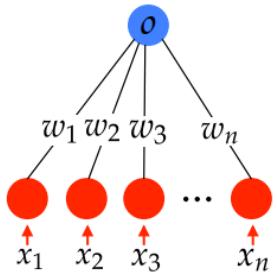

Figure 3.21. Synaptic weights updated upon every iteration determine how much each input ( ) contributes to the output (o) of the perceptron.

Our training routine will consider binary inputs and outputs. Specifically, each input neuron is either stimulated ($x_i = 1$) or not ($x_i = 0$), and the output (a weighted sum of binary inputs) is either 1 or 0.

You may be wondering how an output that is a weighted sum of 1's and 0's is restricted to being either 1 or 0. We appeal to basic principles of neuroscience and realize that neural firing is an all-or-none event. In other words, a neuron fires (o=1) if the weighted sum of inputs exceeds a given threshold, and doesn't fire (o=0) if it does not reach threshold. To simplify matters, we set this threshold to 0 and pass our weighted sum θ to a threshold function $f(\theta)$ which returns a 1 if $\theta > 0$ and a 0 if $\theta \leq 0$. To train a perceptron, you are coding a learning algorithm that governs the evolution of synaptic weights over time. How do these weights change? (Perceptron, 2018)

The Perceptron Learning Algorithm

At each iteration j, the perceptron calculates the output based on the current input pattern (a vector of $x_1 ... x_n$ values) and weights (a vector of $w_1 ... w_n$ values), then updates the weights based on how much its output o differs from the desired output y .The algorithm involves three basic steps at each j: The current output is given by

$$o^j = f(\sum_{i=1}^{n} w_i x_i) \text{ where } f(\theta) = \begin{cases} 1, if\ \theta > 0. \\ 0, if\ \theta \leq 0. \end{cases} \quad (3.2)$$

The difference between o^j and desired output y can be calculated as

$$d^j = y - o^j \quad (3.3)$$

Each weight is then updated by ΔW_i where

$$\Delta W_i = \ell d^j x_i^j \quad for\ \ i = 1...n, w_i = w_i + \Delta w_i \quad (3.4)$$

with l as the learning rate that scales changes in weight. (Perceptron, 2018)

3.4.8.Multilayer Perceptron

A multilayer perceptron (MLP) is a perceptron that teams up with additional perceptrons, stacked in several layers, to solve complex problems.

The diagram below shows an MLP with three layers. Each perceptron in the first layer on the left (the input layer), sends outputs to all the perceptrons in the second layer (the hidden layer), and all perceptrons in the second layer send outputs to the final layer on the right (the output layer).

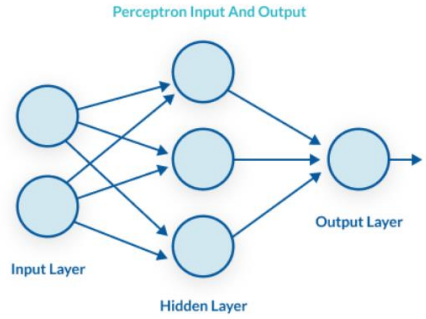

Figure 3.22. Multilayer Perceptron

Each perceptron sends multiple signals, one signal going to each perceptron in the next layer. For each signal, the perceptron uses different weights. In the diagram above, every line going from a perceptron in one layer to the next layer represents a different output. Each layer can have a large number of perceptrons, and there can be multiple layers, so the multilayer perceptron can quickly become a very complex system.

The multilayer perceptron has another, more common name—a neural network. A three-layer MLP, like the diagram above, is called a Non-Deep or Shallow Neural Network. An MLP with four or more layers is called a Deep Neural Network.

One difference between an MLP and a neural network is that in the classic perceptron, the decision function is a step function and the output is binary. In neural networks that evolved from MLPs, other activation functions can be used which result in outputs of real values, usually between 0 and 1 or between -1 and 1. This allows for probability-based predictions or classification of items into multiple labels. (Perceptrons and Multi-layer Perceptrons, 2018)

3.4.9.Radial Based Artificial Neural Networks

(RBF) systems are feed-forward systems prepared utilizing a directed preparing calculation. They are commonly arranged with a solitary shrouded layer of units whose initiation capacity is chosen from a class of capacities called premise capacities. While like back engendering in numerous regards, spiral premise capacity systems have a few focal points. They as a rule train a lot quicker than back engendering systems. They are less helpless to issues with non-stationary data sources on account of the conduct of the spiral premise capacity concealed units.

Promoted by Moody and Darken (1989), RBF systems have demonstrated to be a helpful neural system design. The real contrast between RBF arranges and back proliferation organizes (that is, multi-layer perceptron prepared by Back Propagation calculation) is the conduct of the single concealed layer. As opposed to utilizing the sigmoidal or S-moulded actuation work as in back spread, the shrouded units in RBF systems utilize a Gaussian or some different premise piece work. Each shrouded unit goes about as a privately tuned processor that figures a score for the match between the information vector and its association loads or focuses. As a result, the premise units are very particular example locators. The loads associating the premise units to the yields are utilized to take straight blends of the shrouded units to item the last grouping or yield.

The Structure of the RBF Networks

Radial Basis Functions are first presented in the arrangement of the genuine multivariable insertion issues. Broomhead and Lowe (1988), and Moody and Darken (1989) were the first to misuse the utilization of outspread premise works in the structure of neural systems. The structure of a RBF arranges in its most essential structure includes three completely various layers (Figure 3.23.).

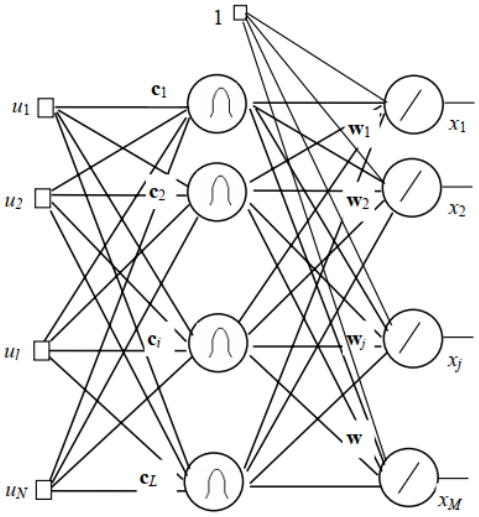

Figure 3.23. Structure of the Standard RBF network

The input layer is made up of source nodes (sensory units) whose number is equal to the dimension p of the input vector **u**.

Hidden layer

The second layer is the concealed layer which is made out of nonlinear units that are associated straightforwardly to the majority of the hubs in the info layer. It is of sufficiently high measurement, which fills an alternate need from that in a multilayer perceptron.

Each hidden unit takes its input from all the nodes at the components at the input layer. As mentioned above the hidden units contains a basis function, which has the parameters centre and width. The centre of the basis function for a node *i* at the hidden layer is a vector **c**i whose size is the as the input vector **u** and there is normally a different centre for each unit in the network. First, the radial distance *di*, between the input vector **u** and the centre of the basis function **c**i is computed for each unit *i* in the hidden layer as $d_i = \|u - c_i\|$ using the Euclidean distance.

The output *hi* of each hidden unit *i* is then computed by applying the basis function G to this distance $h_i = G(d_i, \sigma_i)$.

As it is shown in Figure 0.25, the basis function is a curve (typically a Gaussian function, the width corresponding to the variance, σ_i) which has a peak at zero distance and it decreases as the distance from the centre increases.

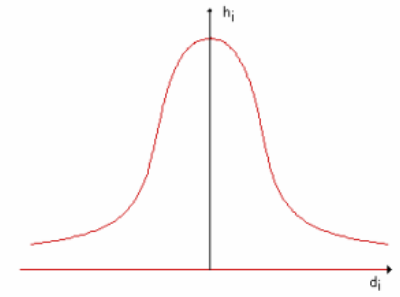

Figure 3.24. The response region of an RBF hidden node around its centre as a function of the distance from this centre.

For an input space $u \in R^2$, that is $M = 2$, this corresponds to the two dimensional Gaussian centred at c_i on the input space, where also $c_i \in R^2$, as it is shown in Figure 3.25.

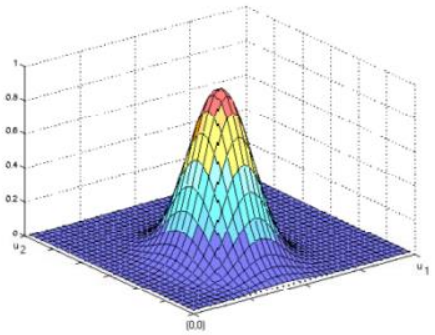

Figure 3.25. Response of a hidden unit on the input space for $u \in R^2$

Output layer

The transformation from the input space to the hidden unit space is nonlinear, whereas the transformation to the hidden unit space to the output space is linear. The j^{th} output is computed as $x_j = f_j(u) = w_{0j} + \sum_{i=1}^{L} w_{ij} h_i \; j = 1, 2, .., M$.

Mathematical model

In summary, the mathematical model of the RBF network can be expressed as:

$$x = f(u),\ f : R^N \rightarrow R^M$$

$$x_j = f_j(u) = w_{0j} + \sum_{i=1}^{L} w_{ij} G(\|u - c_i\|) \quad j = 1, 2, ..., M$$

where is the Euclidean distance between u and c_i .

3.4.10.Adaptive Neuro Fuzzy Inference System (ANFIS)

An adaptive neuro-fuzzy inference system or versatile system based fluffy deduction framework (ANFIS) is a sort of counterfeit neural system that depends on Takagi–Sugeno fluffy induction framework. The procedure was created in the mid-1990s. Since it coordinates both neural systems and fluffy rationale standards, it can possibly catch the advantages of both in a solitary structure. Its derivation framework relates to a lot of fuzzy IF–THEN rules that have learning ability to inexact nonlinear capacities. (Adaptive Neuro Fuzzy Inference System, 2019)

This section describes a class of Neuro-Fuzzy alongside the models and learning methods of versatile systems. The fundamental system structure is a superset of a wide range of neural system standards with directed learning ability. Neuro-Fuzzy frameworks, is the mix of ANN with fluffy frameworks, as a rule have the benefit of permitting a simple interpretation of the last framework into a lot of on the off chance that rules, and the fuzzy system can be seen as a neural system structure with learning conveyed all through association qualities (19011). Research and applications on neuro-fuzzy deduction methodology clarified that neural and fluffy half breed frameworks are useful in fields, for example, the appropriateness of existing calculations for fake neural systems (ANNs), and direct adjustment of learning verbalized as a lot of fuzzy etymological standards. A versatile system, as its name suggests, is a system structure comprising of hubs and directional connections, by and large info yield conduct is controlled by the estimations of an accumulation of modifiable parameters through which the hubs are associated (Jang, Neuro-Fuzzy

Modelling procedures of the IEEE, 1995) The versatile framework utilizes a half and half learning calculation to distinguish parameters of Sugeno-type fuzzy derivation frameworks. It applies a blend of the least-squares technique and the back-engendering inclination drop strategy for preparing FIS participation work parameters to imitate a given preparing informational collection (Rezaei, Hosseini, & Mazinani, 2014). The system learns in two primary stages. In the forward period of the learning calculation, resulting parameters distinguish the least squares gauge. In the retrogressive stage, the blunder signals, which are the subsidiaries of the squared mistake regarding every hub yield, engender in reverse from the yield layer to the info layer. In this regressive pass, the reason parameters are refreshed by the angle plunge calculation. Learning or preparing period of the neural system is a procedure to decide parameter esteems to adequately fit the preparation information. ANFIS preparing can utilize elective calculations to decrease the blunder of the preparation. A blend of the slope plunge calculation and a least squares calculation is utilized for a compelling quest for the ideal parameters. The principle advantage of such a cross breed approach is, that it combines a lot quicker, since it decreases the inquiry space measurements of the backpropagation technique utilized in neural systems (Hamdan & Garibaldi, 2010). ANFIS are the fuzzy Sugeno model put in structure of the versatile framework which serves in model structure and approval of created model to encourage preparing and adjustment (Roy, 2005).

Architecture of ANFIS

An adaptive network is a multilayer feed-forward system made out of hubs associated by coordinated connections, in which every hub plays out a specific capacity on its approaching sign to produce a solitary hub yield. Each connection in a versatile system indicates the heading of sign stream starting with one hub then onto the next; no loads is related with the connection. All the more explicitly, the setup of a versatile system plays out a static hub work on its approaching sign to produce a solitary hub yield and every hub capacity is a parameterized work with modifiable parameters; by changing these parameters, the hub capacities just as the general conduct of the versatile system, are changed. Figure 3.25 demonstrates whole framework engineering comprises of five layers, to be specific fuzzy layer, item layer, standardized layer, de-fuzzy layer

and all out yield layer. With info/yield information for given arrangement of parameters, the ANFIS strategy models a fuzzy induction framework (FIS) whose enrolment work parameters are tuned (balanced) utilizing either a backpropagation calculation alone, or in change with a least squares sort of technique. The primary target of the ANFIS is to decide the ideal estimations of the proportional fuzzy induction framework parameters by applying a learning calculation. The parameter improvement is done in such a route during the instructional course that the mistake between the objective and the genuine yield is limited. A cross breed calculation is utilized for enhancement, which is the mix of least square gauge and inclination plummet strategy. The parameters to be enhanced in ANFIS are the reason parameters. These parameters characterize the state of the participation capacities (Patel & Parekh, 2014). So as to diminish the blunder measure, any of a few advancement schedules can be connected in the wake of establishing MFs. The parameter set of a versatile system enables fuzzy frameworks to gain from the information they are demonstrating. This paper expects that versatile framework under thought has two data sources V1 and V2 and one yield f. Give us a chance to investigate a first request Takagi, Sugeno and Kang (TSK) fuzzy derivation framework containing two rules:

$$Rule\ 1: If(v\ is\ V_1)\ and\ (d\ is\ D_1)\ then\ f_1 = p_1 v + q_1 d + r_1$$
$$Rule\ 2: If(v\ is\ V_2)\ and\ (d\ is\ D_2)\ then\ f_2 = p_2 v + q_2 d + r_2$$

Where p_1 , p_2 , q_1 , q_2 , r_1 and r_2 are linear parameters and V_1 , V_2 , D_1 and D_2 are non linear parameters, in which V1 and D1 are the membership functions of ANFIS (antecedent). p_1, q_1 , r_1 are the consequent parameters (Pratama, Rajab, & Joo, 2011). To reflect versatile capacities, we utilize both circle and square. A circle demonstrates fixed hub while square shows versatile hub for example the parameter can be changed during adjusting or preparing. ANFIS is made from mix of fluffy rationale and neural system.

While structuring of ANFIS model, it is critical that the quantity of preparing ages, the quantity of enrolment capacities and the quantity of fuzzy guidelines ought to be tuned precisely. Mapping of those parameters is exceedingly critical for the framework since it might lead framework to over fit the information or won't almost certainly fit the

information. This modifying can be acquired by utilizing a mixture calculation joining the least-squares strategy and the inclination plummet technique with a mean square mistake strategy. The lesser distinction between ANFIS yield and the ideal target implies a superior (increasingly exact) ANFIS framework. So, we will in general decrease the preparation blunder in the preparation procedure (Uc, Karahoca, & Karahoca, 2013). The incorporation between fluffy rationale and neural system to be specific fuzzy neural system (FNN) has been normal and grown; by and large the course of action of fuzzy rationale and the neural system is called as ANFIS. Neural framework has numerous data sources and furthermore has different yields, yet the fluffy rationale has abundant information sources and single yield, so the blend of this two is known as ANFIS.

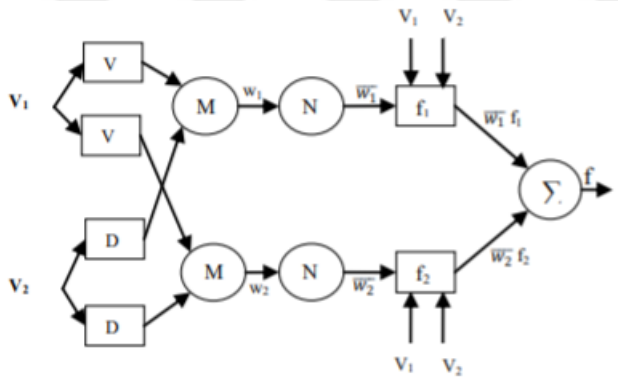

Figure 3.26. Basic architecture of ANFIS

Layers of ANFIS

For simplicity, the fuzzy inference system is under consideration of two inputs v, d and one output f. A brief summary of five layers of the ANFIS algorithm is shown below.

Layer 1

Each input node i in this layer is an adaptive node which produce membership grade of linguistic label. It is a fuzzy layer, in which and d are input of system. $O_{1,i}$ is the

output of the i^{th} node of layer l. Each adaptive node is a square node with square function represented using Eq. (3.5):

$$O_{1,i} = \mu_{v,i}(v) \quad for \;\; i = 1,2$$
$$O_{1,j} = \mu_{d,j}(v) \quad for \;\; j = 1,2 \tag{3.5}$$

Where $O_{1,i}$ and $O_{1,j}$ denote output function $\mu_{v,i}$ and $\mu_{d,j}$ denote membership function. For example, if we choose triangular membership function, $\mu_{v,i}(v)$ is given by:

$$\mu_{vi}(v) = \max\left[\min\left(\frac{v-a_i}{b_i-a_i}, \frac{c_i-v}{c_i-b_i}\right), 0\right] \tag{3.6}$$

Where $\{a_i, b_i, c_i\}$ are the parameter of triangular membership function? In another example, if we choose $\mu_{v,i}(v)$ to be bell shaped is given by:

$$\mu(v) = \frac{1}{1+\left\{\left(\frac{v-c_i}{a_i}\right)^2\right\}^{b_i}} \tag{3.7}$$

Where $\{a_i, b_i, c_i\}$ are the parameter set that changes shapes of M.F accordingly? Value of a_i and c_i that can be adjusted to vary the centre and width of membership function and then b_i is used to control slopes at crossover points of next membership function. Parameters in this layer are referred to as "premise parameter".

Layer 2

This layer checks weights of each membership function, it receives input values v_i from first layer and acts as a membership function to represent fuzzy sets of respective input variables. Every node in this layer is fixed node labelled with M and output is

calculated via product of all incoming signals. The output in this layer can be represented using Eq. (3.9):

$$o_{2,i} = w_i = \mu_{v,i}(v).\mu_{Dj}(d),\ i = 1,2 \tag{3.9}$$

Which are the firing strengths of the rules. In general, any T-norm operator that performs fuzzy AND can be used as a node function in this layer.

Layer 3

Every node in this layer is fixed marked with circle labelled with N, indicating normalization to the firing strength from previous layer. This layer performs pre-condition matching of fuzzy rules, i.e. they compute activation level of each rule, the number of layers being equal to number of fuzzy rules. The i^{th} node in this layer calculate ratio of i^{th} rule's strength to the sum of all rules firing strength. The output of this layer can be expressed as *wi* using Eq. (2.10):

$$O_{3,i} = \overline{w_i} = \frac{w_i}{w_1 + w_2}, \quad i = 1,2 \tag{3.10}$$

For convenience, outputs of this layer will be called as normalized firing strengths.

Layer 4

This layer provides output values y, resulting from the inference of rules. The resultant output is simply a product of normalized firing rule strength and first order polynomial. Weighted output of rule represented by node function as:

$$O_{4,i} = \overline{w_i} f_i = \overline{w_i}(p_i v + q_i d + r_i), \quad i = 1,2 \tag{3.11}$$

Where $O_{4,i}$ represents layer 4 output. In this layer, $p_{i,}\, q_i$ and r_i are linear parameter or consequent parameter.

Layer 5

This layer is called output layer which sums up all the inputs coming from layer 4 and transforms fuzzy classification results into crisp values. This layer consists of single fixed node labelled as "∑". This node computes summation of all incoming signals calculated using Eq. (2.12).

$$O_{5,i} = \sum_i \overline{w_i} f_i = \frac{\sum_i w_i f_i}{w_1 + w_2}, \quad i = 1,2 \tag{3.12}$$

Thus, it is observed that when the values of premise parameter are fixed, the overall output of the adaptive network can be expressed as linear combination of a consequent parameter. Constructed network has exactly the same function as a Sugeno fuzzy model. Overall output of a system (z) can be expressed as in Eq. (2.13). It can be observed that ANFIS architecture consists of two adaptive layers, namely the first layer and the fourth layer. The three modifiable parameters $\{a_i, b_i, c_i\}$ are so-called premise parameter in first layer and in the fourth layer, there are also three modifiable parameters $\{p_i, q_i, r_i\}$ pertaining to the first order polynomial. These parameters are so-called consequent parameters (Efosa & Akwukwuma, 2013).

$$\begin{aligned} z &= \frac{w_1}{w_1 + w_2} f_1 + \frac{w_2}{w_1 + w_2} f_2 + \ldots + \frac{w_n}{w_{n-1} + w_n} f_n \\ z &= \overline{w_1}(p_1 Q + q_1 M + \ldots + m_1 F + r_1) \\ &\quad + \ldots + \overline{w_n}(p_n Q + q_n M + \ldots + m_n F + r_n) \end{aligned} \tag{3.13}$$

Learning Algorithm of ANFIS

Neuro-adaptive learning techniques bless with a strategy for the fuzzy demonstrating method to learn data about an informational collection. It processes the enrolment work parameters that best enable the related fuzzy surmising framework to follow the

given information/yield information. The parameters related with the participation capacities changes through the learning procedure (Power). So as to all the more effectively adapt to true issues, the errand of the learning calculation for this engineering is to tune all the modifiable parameters, to plan the ANFIS yield coordinate the preparation information. To improve the rate of combination, the half breed system can be prepared by a cross breed learning calculation joining least square technique and inclination plunge strategy can be utilized. The least squares technique can be utilized to recognize the ideal estimations of the subsequent parameter on the layer 4 with reason parameter fixed. Angle vector gives a proportion of how well the fluffy deduction framework is displaying the information/yield information for a given arrangement of parameters. At the point when the slope vector is acquired, any of a few streamlining schedules can be connected so as to modify the parameters to diminish some mistake measure. At the point when the reason parameters are not fixed, at that point the hunt space increases and the combination of the preparation turns out to be slower. The cross-breed calculation is made out of a forward pass (LSM) and a backward pass (GDM). When the ideal ensuing parameters are discovered, in reverse pass begins. In the regressive pass, mistakes are spread in reverse and the reason parameters relating to the fluffy sets in the info area refreshed by slope drop strategy (Roy, 2005). ANFIS utilizes a blend of least squares estimation and back-spread for participation work parameter estimation. Two goes in the half and half learning calculation for ANFIS appeared in Table 3.1.

Table 3.1. Passes of Hybrid learning algorithm

	Forward pass	Backward pass
Premise parameters	Fixed	Gradient descent
Consequent parameters	Least square	Fixed
Signals	Node outputs	Error signals

The output error is utilized to adjust the reason parameters by methods for a standard back-engendering calculation to limit the mean square blunder capacity characterized

by Eq. (2.14). It has been demonstrated that this cross-breed calculation exceptionally effective in preparing the ANFIS (Jang, ANFIS :Adaptive-Network-Based Fuzzy Inference System, 1993).

$$\mathrm{E}(\theta)=\sum_{i=1}^{m}(z_i - a_i^T\theta)^2 = e^T e = (z - A\theta)^T(z - A\theta) \tag{3.14}$$

Where $e = z - A\theta$ is the error vector produced by a specific choice of θ? In Eq. (3.14) the squared error is minimized and is called the least squares estimator (LSE) [7]. Therefore, the hybrid learning algorithm can be applied directly. More specifically, the error signals proliferate backward and the premise parameters are updated by Gradient Descent (GD) and node outputs go forward until layer 3 and the consequent parameters are identified by the Least Squares (LS) method. This hybrid learning is structured as by defining, linear and nonlinear parameters are illustrious each iteration (epoch) of GD update the nonlinear parameters, LS follows to identify the linear parameters (Power).

3.5. Methods of Evaluation of Classification Results

3.5.1.Confusion Matrix

A confusion matrix is a table that outlines different predictions and test results and contrasts them with real-world values. Perplexity grids are utilized in measurements, information mining, AI models and other AI applications. A confusion matrix can likewise be called an error matrix.

Confusion matrices are utilized to make the top to bottom examination of factual information quicker and the outcomes simpler to peruse clear data visualization. The tables can help investigate blames in insights, information mining, legal sciences and therapeutic tests. An intensive investigation enables clients to choose what results show how blunders are made as opposed to just evaluating execution.

Confusion matrices use a simple format to log predictions. In the columns of a confusion matrix for an AI model, the potential forecasts are adjusted on the right-hand side and the facts along the top. In the columns underneath the realities, expectations or results are recorded. Results can incorporate the right sign of a positive as a true positive or a negative as a true negative as well as an incorrect positive as a false positive or an incorrect negative as a false negative.

3.5.2.MAE and RMSE

MAE measures the average magnitude of the errors in a set of predictions, without considering their direction. It's the average over the test sample of the absolute differences between prediction and actual observation where all individual differences have equal weight.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|y_i - \overline{y_i}\right| \tag{3.15}$$

If the absolute value is not taken (the signs of the errors are not removed), the average error becomes the Mean Bias Error (MBE) and is usually intended to measure average model bias. MBE can convey useful information, but should be interpreted cautiously because positive and negative errors will cancel out.

RMSE is a quadratic scoring rule that also measures the average magnitude of the error. It's the square root of the average of squared differences between prediction and actual observation.

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \overline{y_j})^2} \tag{3.16}$$

Both MAE and RMSE express average model forecast mistake in units of the variable of intrigue. The two measurements can extend from 0 to ∞ and are not interested toward mistakes. They are adversely arranged scores, which means lower esteems are better.

Taking the square foundation of the normal squared blunders makes them intrigue suggestions for RMSE. Since the mistakes are squared before they are arrived at the midpoint of, the RMSE gives a moderately high weight to huge blunders. This implies the RMSE should be progressively helpful when huge mistakes are especially unfortunate. The three tables underneath show models where MAE is unfaltering and RMSE increments as the change related with the recurrence conveyance of blunder extents likewise increments (MAE and RMSE, 2019).

3.5.3.AUC and ROC Curve

AUC - ROC curve is a performance measurement for classification problem at various thresholds settings. ROC is a probability curve and AUC represent degree or measure of separability. It tells how much model is capable of distinguishing between classes. Higher the AUC, better the model is at predicting 0s as 0s and 1s as 1s. By analogy, Higher the AUC, better the model is at distinguishing between patients with disease and no disease.

The ROC curve is plotted with TPR against the FPR where TPR is on y-axis and FPR is on the x-axis.

$$TPR / Recall / Sensitivity = \frac{TP}{TP + FN} \tag{3.17}$$

$$Specificity = \frac{TN}{TN + FP} \tag{3.18}$$

$$FPR = 1 - Specificity = \frac{FP}{TN + FP} \tag{3.19}$$

An excellent model has AUC near to the 1 which means it has good measure of separability. A poor model has AUC near to the 0 which means it has worst measure of separability. In fact, it means it is reciprocating the result. It is predicting 0s as 1s and 1s as 0s. And when AUC is 0.5, it means model has no class separation capacity whatsoever (Understanding AUC, ROC Curve, 2019).

CHAPTER 4

EXPERIMENTAL STUDIES

4.1. Datasets Used, Microarray Expression Profiles and Results Taken for Cancer Cell Gene

The classification technique, which is one of the data mining functions, is now widely used in many fields. Previously, classical statistical classification methods (logistic regression, variance analysis, linear regression analysis) were used to classify data in many areas. However, with the inadequate classification of these methods over time, statistical methods have been developed and started to be used. Some of these statistical methods developed are Support Vector Machines, Decision Trees, Bayesian Networks, Relationship-based classifiers, and k-nearest neighbouring method.

These statistical classification methods have been used in many fields recently, and the most used fields are engineering and especially medicine. Because of the recent microarray gene studies in the field of medicine, there has been a large increase in the size of the data and the current statistical methods have been used to classify these data.

In addition to the statistical methods currently used extensively, researchers have recently been interested in the field of artificial intelligence in order to be able to rank higher in data mining. The reason why researchers are seeking artificial intelligence-based classification methods is that they think they can be an alternative to statistical methods and whether these methods can yield better results than statistical methods in classification.

This study includes both classification by statistical methods and classification by ANFIS based on artificial intelligence and comparison of these two methods. As far as the researches in the literature are found, statistical methods and ANFIS methods have been used for classification and studies have been done to compare the performance of these classification methods with each other!

The gene expression data set used is the data of Breast Cancer and CNS Cancer patients from the University of Shenzhen [a1]. Using these data sets, classification successes were compared.

The breast cancer data set consists of 97 samples and 24482 genes. 51 of the samples were normal and 46 of them were tumour patients. The central nervous system cancer data set consists of 60 samples and 7130 genes. 39 of the samples were normal and 21 of them were tumour patients. Table 4.1 shows it.

Table 4.1. Breast cancer and CNS cancer data set information

	Number of Sample	Number of Patients with Cancer	Number of Normal Patients	Number of Genes
Breast Cancer	97	46	51	24482
CNS Cancer	60	21	39	7130

Classification algorithms; open source, Java programming language written with the Weka program was implemented. As a classification algorithm, frequently used statistical classification algorithms such as Support Vector Machines, K-Nearest Neighbourhood and Naive Bayes were used. 3 different attribute selection methods were applied for all classification methods. These attribute selection methods are; WrapperSubsetEval, CfsSubsetEval (correlation based) and ChiSquareSubsetEval. In Section 3.2, these methods are mentioned in detail. For the accuracy values resulting from the classification, the disturbance matrices are calculated and the AUC value below the drawn ROC curve is calculated.

While the K-nearest neighbour classifier is used, 4 different values are based on the K value of 1,3,5,7. At the same time, while the nearest neighbour classifier is used in Weka program, the research algorithm is used LinearNNSearch by default. LinearNNSearch applies the brute force search algorithm for the nearest neighbour search.

In the Weka program, when using the Support vector machine (SMO) classifier, PolyKernel is used as the default because it has better performance.

Two different microarray gene expression cancer data sets were used in the study. First of all, classification algorithms were applied on this cancer data without selection of features. 3 different classification algorithms were used in this classification process. Two different methods, k-fold cross validation and split percentage, were used to determine the training and test sets during this classification process. The k-fold cross validation, which is the first of these methods, has a value of k 5, with 2,4,6,8,10; the other method, the split percentage, the education and test set rate 66 percent and 70 percent were used as 2 different values.

4.1.1. Classification without Attribute Selection

a) Results from Breast Cancer dataset in Table 4.2, Table 4.3. The AUC results of the classification algorithms applied are shown in Figure 4.1, Figure 4.2, Figure 4.3 and the graphical representation of the accuracy values is expressed as in Figure 4.4.

Table 4.2. Success rate of classification of breast cancer data set without attribute selection

Breast Cancer		Cross Validation					Percentage Split	
		2	4	6	8	10	66%	70%
KNN	K1	57.732	55.670	59.794	58.763	60.825	57.576	48.276
	K3	59.794	57.732	58.763	58.763	58.763	63.636	65.517
	K5	71.134	64.949	59.794	61.856	62.887	60.606	65.517
	K7	69.072	65.979	61.856	67.010	67.010	60.606	62.069
NB		51.546	53.608	53.608	53.608	54.639	57.576	62.069
SMO		65.979	74.227	65.979	71.134	68.041	60.606	58.621

Table 4.3. Confusion matrix values after classification of Breast Cancer data set without attribute selection

Breast Cancer Data Set Confusion Matrix								
		KNN(k=3)		Naive Bayes		SMO		Class
Cross Validation	2-Fold	24	22	10	36	31	15	relapse
		17	34	11	40	18	33	non-relapse
	4-Fold	21	25	1	45	36	10	relapse
		16	35	0	51	15	36	non-relapse
	6-Fold	22	24	1	45	32	14	relapse
		16	35	0	51	19	32	non-relapse
	8-Fold	21	25	2	44	35	11	relapse
		15	36	1	50	17	34	non-relapse
	10-Fold	21	25	2	44	32	14	relapse
		16	35	0	51	17	34	non-relapse
Percentage Split	%66	6	8	0	14	7	7	relapse
		4	15	0	19	6	13	non-relapse
	%70	5	6	0	11	5	6	relapse
		4	14	0	18	6	12	non-relapse

Table 4.4. MAE and RMSE values after classification of Breast Cancer data set without attribute selection

Breast Cancer Dataset MAE and RMSE							
		KNN(k=3)		Naive Bayes		SMO	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
Cross Validation	2-Fold	0.4	0.5072	0.4845	0.6961	0.3402	0.5833
	4-Fold	0.4064	0.5231	0.4639	0.6811	0.2577	0.5077
	6-Fold	0.4199	0.5465	0.4639	0.6811	0.3402	0.5833
	8-Fold	0.3994	0.536	0.4639	0.6811	0.2887	0.5373
	10-Fold	0.4233	0.5578	0.4536	0.6735	0.3196	0.5653
Percentage Split	%66	0.3950	0.5270	0.4242	0.6513	0.3939	0.6276
	%70	0.426	0.5417	0.3793	0.6159	0.4138	0.6433

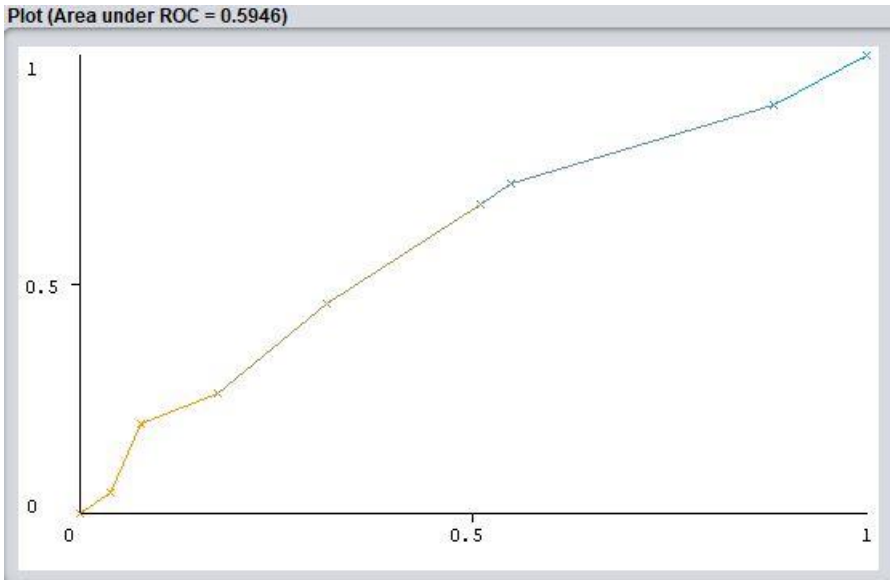

Figure 4.1. ROC curve and AUC value after classification by applying KNN to breast cancer dataset without attribute selection (For 10-folds Cross Validation)

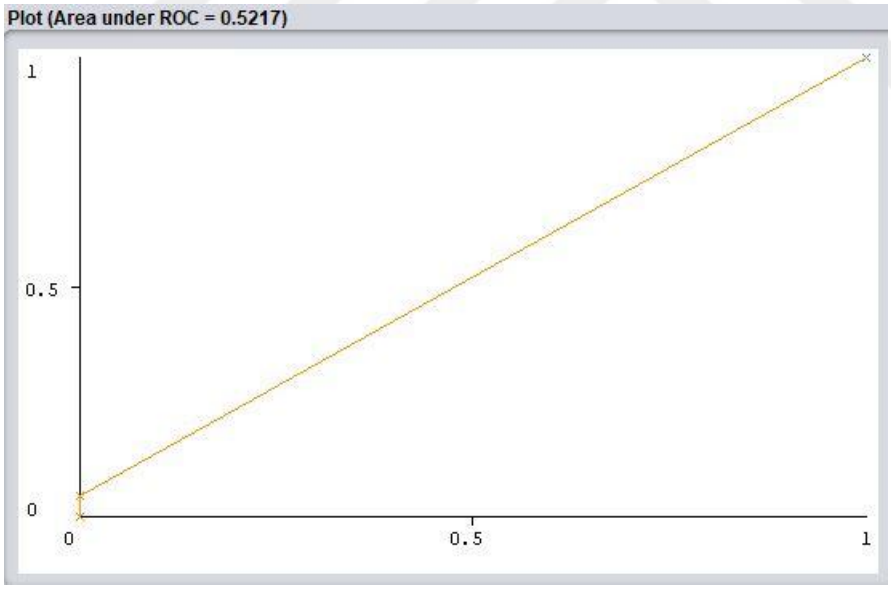

Figure 4.2. ROC curve and AUC value after classification by applying NB to breast cancer dataset without attribute selection (For 10-folds Cross Validation)

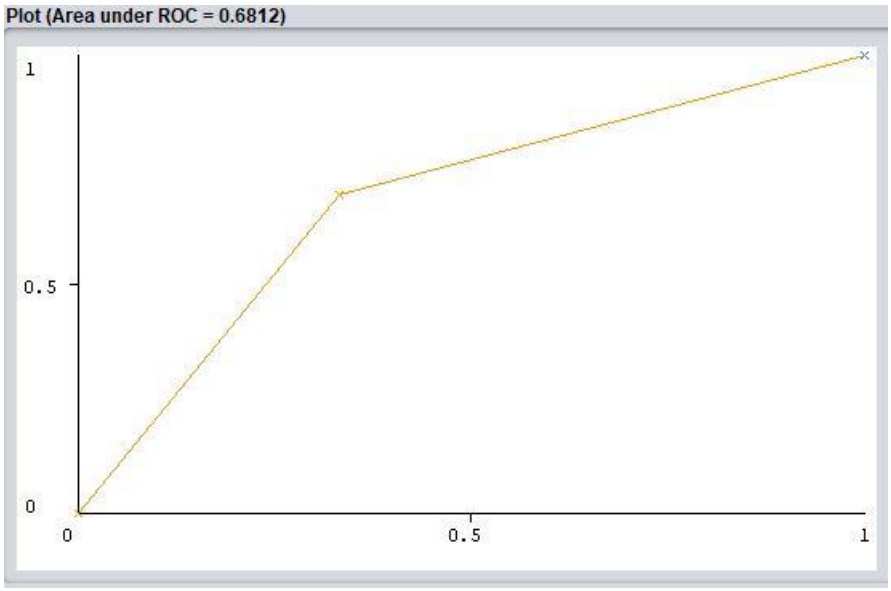

Figure 4.3. ROC curve and AUC value after classification by applying SMO to breast cancer dataset without attribute selection (For 10-folds Cross Validation)

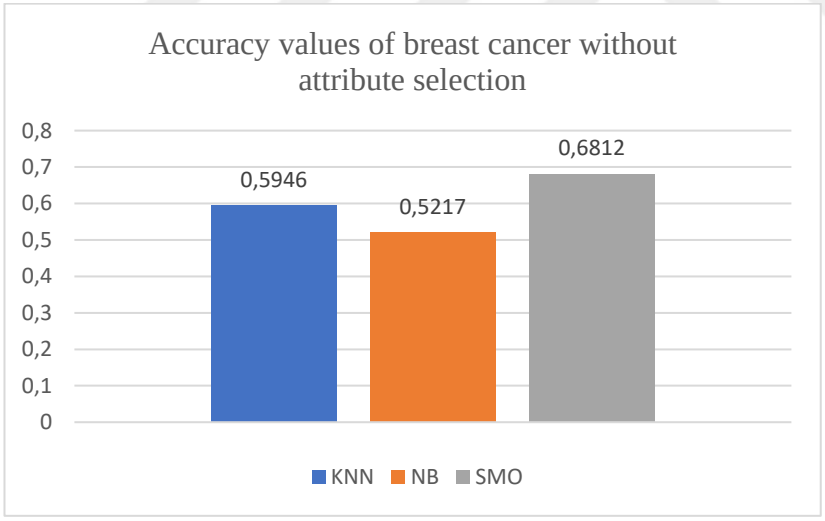

Figure 4.4. The graphical representation of the accuracy values of Breast Cancer without attribute selection (For 10-folds Cross Validation)

b) Results from CNS Cancer dataset in Table 4.4, Table 4.5. The AUC results of the classification algorithms applied are shown in Figure 4.5, Figure 4.6, Figure 4.7 and the graphical representation of the accuracy values is expressed as in Figure 4.8.

Table 4.5. Success rate of classification of CNS cancer data set without attribute selection

CNS Cancer		Cross Validation					Percentage Split	
		2	4	6	8	10	66%	70%
KNN	K1	51.667	51.667	60.000	60.000	56.667	40.000	38.889
	K3	61.667	61.667	61.667	61.667	56.667	40.000	33.333
	K5	63.333	58.333	65.000	60.000	65.000	40.000	27.778
	K7	63.333	58.333	63.333	56.667	63.333	55.000	38.889
NB		68.333	56.667	63.333	61.667	61.667	60.000	55.556
SMO		65.000	60.000	70.000	68.333	68.333	55.000	44.444

Table 4.6. Confusion matrix values after classification of CNS Cancer data set without attribute selection

CNS Cancer Data Set Confusion Matrix								
		KNN(k=3)		Naive Bayes		SMO		Class
Cross Validation	2-Fold	10	11	6	15	5	16	relapse
		12	27	4	35	5	34	non-relapse
	4-Fold	9	12	8	13	5	16	relapse
		11	28	13	26	8	31	non-relapse
	6-Fold	10	11	11	10	10	11	relapse
		12	27	12	27	7	32	non-relapse
	8-Fold	9	12	11	10	9	12	relapse
		11	28	13	26	7	32	non-relapse
	10-Fold	9	12	11	10	10	11	relapse
		14	25	13	26	8	31	non-relapse
Percentage Split	%66	2	6	1	7	2	6	relapse
		6	6	1	11	3	9	non-relapse
	%70	2	6	1	7	1	7	relapse
		6	4	1	9	3	7	non-relapse

Table 4.7. MAE and RMSE values after classification of CNS Cancer data set without attribute selection

CNS Cancer Dataset MAE and RMSE							
		KNN(k=3)		Naive Bayes		SMO	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
Cross Validation	2-Fold	0.4348	0.5279	0.3167	0.5627	0.35	0.5916
	4-Fold	0.4672	0.5525	0.4334	0.6583	0.4	0.6325
	6-Fold	0.4507	0.5342	0.3667	0.6055	0.3	0.5477
	8-fold	0.4397	0.517	0.3833	0.6191	0.3167	0.5627
	10-Fold	0.4616	0.5345	0.3833	0.6191	0.3167	0.5627
Percentage Split	%66	0.5328	0.5939	0.4	0.6325	0.45	0.6708
	%70	0.5547	0.616	0.4444	0.6667	0.5556	0.7454

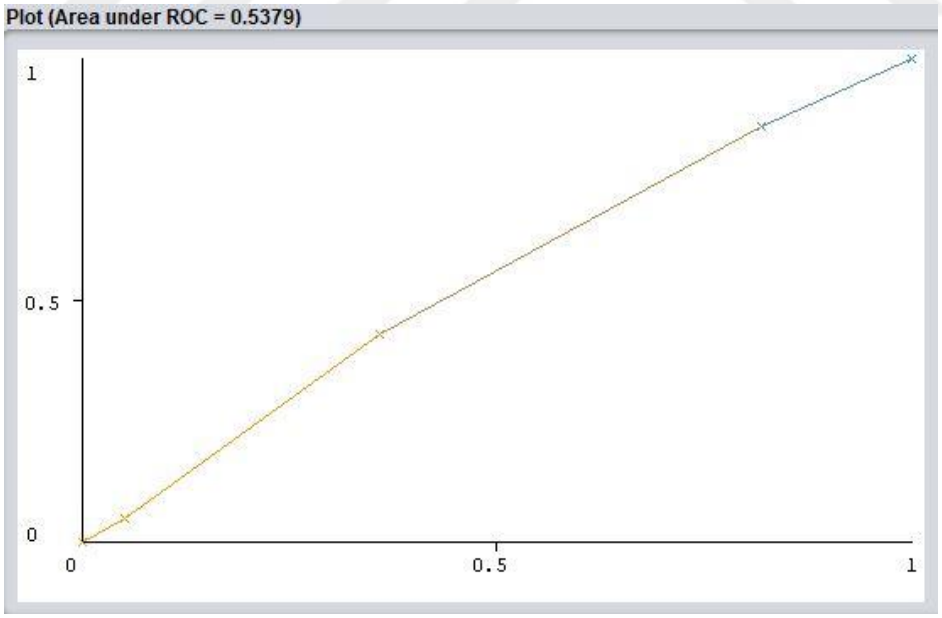

Figure 4.5. ROC curve and AUC value after classification by applying KNN to CNN Cancer dataset without attribute selection (For 10-folds Cross Validation)

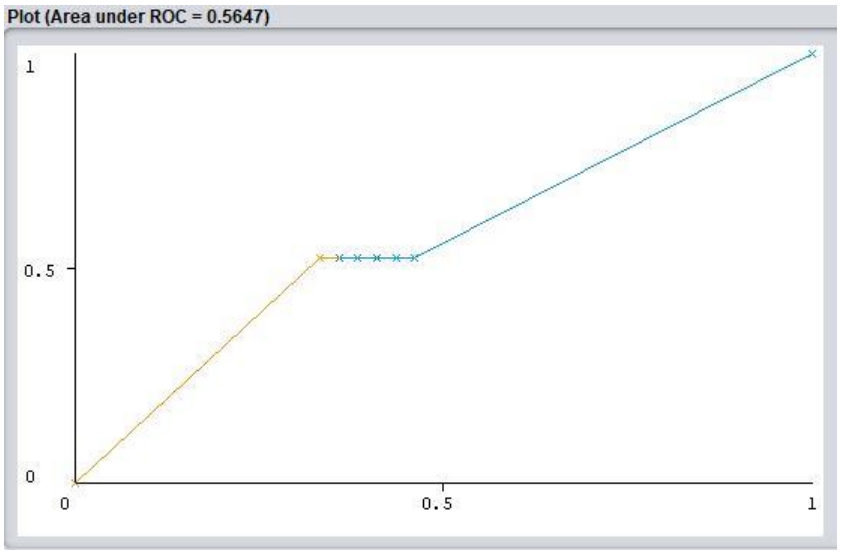

Figure 4.6. ROC curve and AUC value after classification by applying NB to CNN Cancer dataset without attribute selection (For 10-folds Cross Validation)

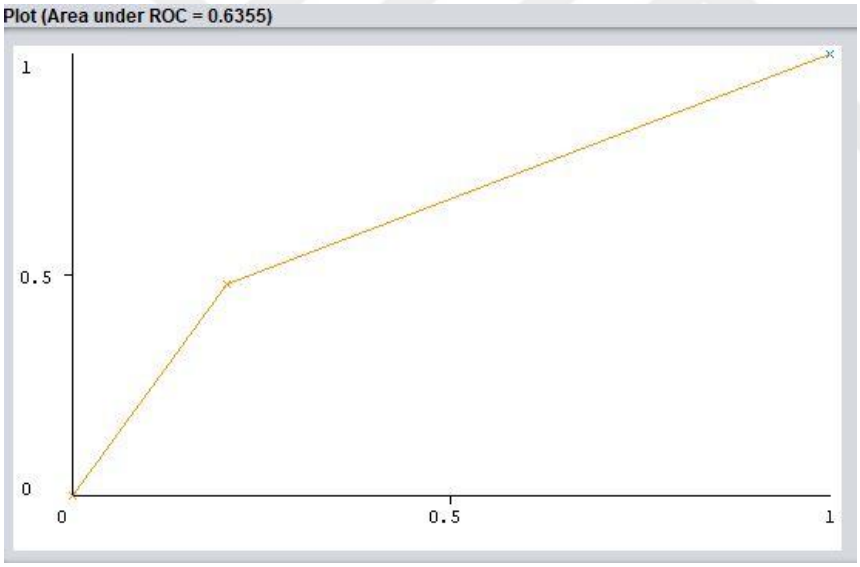

Figure 4.7. ROC curve and AUC value after classification by applying SMO to CNN Cancer dataset without attribute selection (For 10-folds Cross Validation)

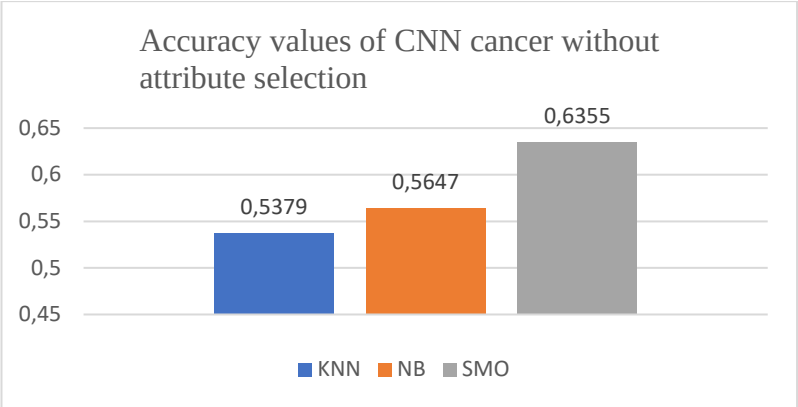

Figure 4.8. The graphical representation of the accuracy values of CNN Cancer without attribute selection (For 10-folds Cross Validation)

4.1.2. Classification with Attribute Selection

4.1.2.1. CfsSubsetEval

Weka CfsSubsetEval evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them. Subsets of features that are highly correlated with the class while having low intercorrelation are preferred.

In this attribute selection scenario, BestFirst Search is used as the research method. It searches the space of attribute subsets by greedy hill climbing augmented with a backtracking facility. In this way, it is targeted to find genes that are thought to be more likely to give information about the disease. And then these reduced attributes and cancer data are subjected to classification.

In this feature selection, the breast cancer attribute was reduced from 24482 to 138 and the CNS cancer was reduced from 7130 to 39.

Table 4.8. Number of attributes with/without CfsSubsetEval

Number of attributes	No Attribute Selection	With Attribute Selection by CfsSubsetEval
Breast Cancer	24482	138
CNS	7130	39

a) The Breast Cancer classification results of the data set being reduced after the attribute selection made using CfsSubsetEval are in Table 4.7, Table 4.8. The AUC results of the classification algorithms applied are shown in Figure 4.9, Figure 4.10, Figure 4.11 and the graphical representation of the accuracy values is expressed as in Figure 4.12.

Table 4.9. Breast cancer dataset classification success rate after attribute selection with CfsSubsetEval

Breast Cancer		Cross Validation					Percentage Split	
		2	4	6	8	10	66%	70%
KNN	K1	84.536	82.474	78.351	80.412	81.443	78.788	75.862
	K3	81.443	81.443	80.412	84.536	82.474	78.788	79.310
	K5	76.289	79.381	78.351	79.381	81.443	81.818	82.759
	K7	78.351	76.289	76.289	77.320	76.289	75.758	75.862
NB		71.134	61.856	59.794	57.732	56.701	57.576	62.069
SMO		87.629	81.443	83.505	85.567	84.536	84.849	75.862

Table 4.10. Confusion matrix values after classification of Breast Cancer dataset with CfsSubsetEval

Breast Cancer Data Set Confusion Matrix								
		KNN(k=3)		Naive Bayes		SMO		Class
Cross Validation	2-Fold	35	11	21	25	40	6	relapse
		7	44	3	48	6	45	non-relapse
	4-Fold	35	11	9	37	37	9	relapse
		7	44	0	51	9	42	non-relapse
	6-Fold	33	13	7	39	38	8	relapse
		6	45	0	51	8	43	non-relapse
	8-Fold	37	9	5	41	38	8	relapse
		6	45	0	51	6	45	non-relapse
	10-Fold	35	25	5	41	37	9	relapse
		6	45	1	50	6	45	non-relapse
Percentage Split	%66	9	5	0	14	13	1	relapse
		2	17	0	19	4	15	non-relapse
	%70	7	4	0	11	8	3	relapse
		2	16	0	18	4	14	non-relapse

Table 4.11. MAE and RMSE values after classification of Breast Cancer dataset with CfsSubsetEval

Breast Cancer Dataset MAE and RMSE							
		KNN(k=3)		Naive Bayes		SMO	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
Cross Validation	2-Fold	0.2339	0.3562	0.2866	0.5323	0.1237	0.3517
	4-Fold	0.2259	0.3777	0.3865	0.6196	0.1856	0.4308
	6-Fold	0.2495	0.3854	0.4037	0.6343	0.1649	0.4061
	8-Fold	0.2085	0.3608	0.4249	0.6505	0.1443	0.3799
	10-Fold	0.2289	0.3763	0.4297	0.6537	0.1546	0.3932
Percentage Split	%66	0.2451	0.4091	0.4242	0.6513	0.1515	0.3892
	%70	0.2325	0.3905	0.3793	0.6159	0.2414	0.4913

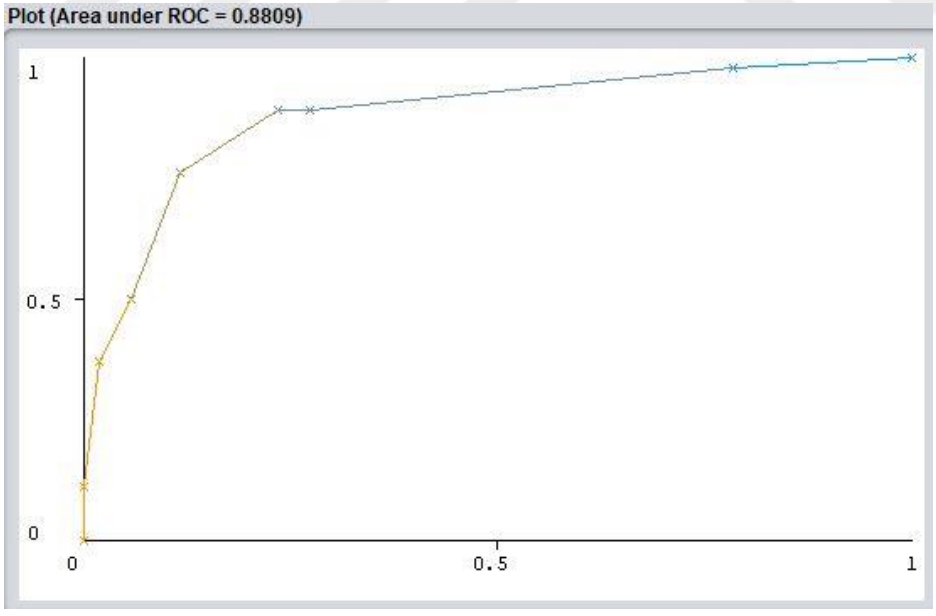

Figure 4.9. ROC curve and AUC value after classification by applying KNN to breast cancer dataset with CfsSubsetEval attribute selection (For 10-folds Cross Validation)

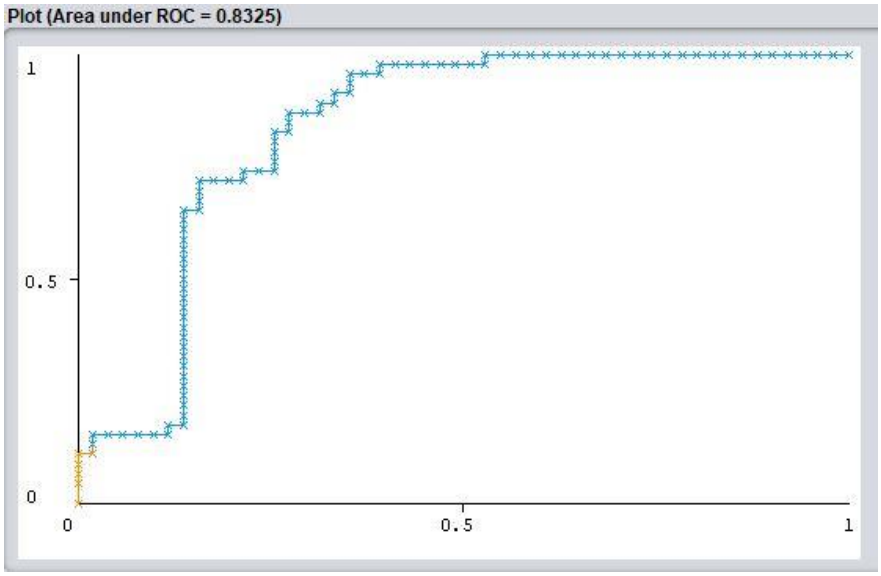

Figure 4.10. ROC curve and AUC value after classification by applying NB to breast cancer dataset with CfsSubsetEval attribute selection (For 10-folds Cross Validation)

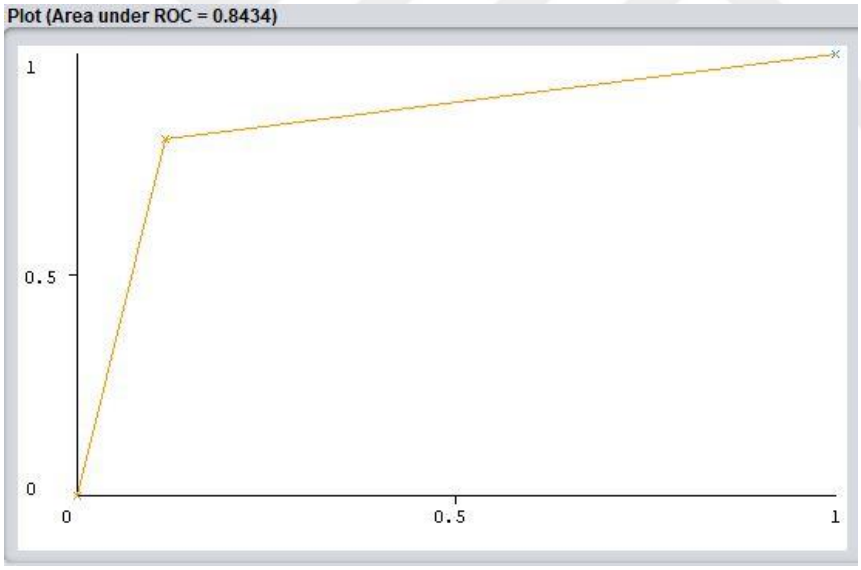

Figure 4.11. ROC curve and AUC value after classification by applying SMO to breast cancer dataset with CfsSubsetEval attribute selection (For 10-folds Cross Validation)

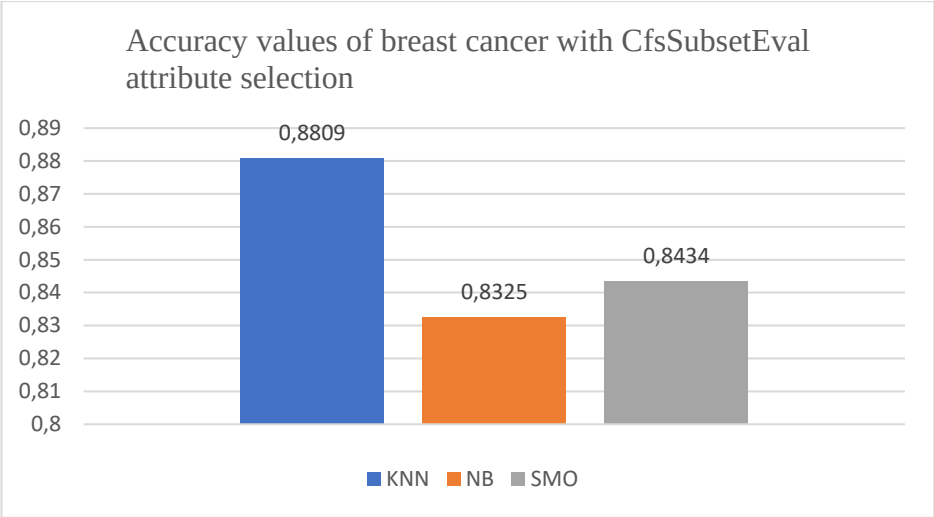

Figure 4.12. The graphical representation of the accuracy values of Breast Cancer with CfsSubsetEval attribute selection (For 10-folds Cross Validation)

b) The CNS Cancer classification results of the data set being reduced after the attribute selection made using CfsSubsetEval are in Table 4.9, Table 4.10. The AUC results of the classification algorithms applied are shown in Figure 4.13, Figure 4.14, Figure 4.15 and the graphical representation of the accuracy values is expressed as in Figure 4.16.

Table 4.12. CNS Cancer dataset classification success rate after attribute selection with CfsSubsetEval

CNS Cancer		Cross Validation					Percentage Split	
		2	4	6	8	10	66%	70%
KNN	K1	68.333	73.333	76.667	76.667	80.000	60.000	55.556
	K3	68.333	71.667	76.667	76.667	78.333	60.000	55.556
	K5	76.667	76.667	75.000	80.000	76.667	65.000	61.111
	K7	68.333	75.000	75.000	76.667	76.667	65.000	61.111
NB		73.333	76.667	78.333	73.333	75.000	70.000	66.667
SMO		85.000	88.333	93.333	88.333	88.333	75.000	77.778

Table 4.13. Confusion matrix values after classification of CNS Cancer dataset with CfsSubsetEval

CNS Cancer Dataset Confusion Matrix								
		KNN(k=3)		Naive Bayes		SMO		Class
Cross Validation	2-Fold	16	5	11	10	17	4	relapse
		14	25	6	33	5	34	non-relapse
	4-Fold	15	6	14	7	17	4	relapse
		11	28	7	32	3	36	non-relapse
	6-Fold	16	5	15	6	20	1	relapse
		9	30	7	32	3	36	non-relapse
	8-Fold	18	3	15	6	17	4	relapse
		11	28	10	29	3	36	non-relapse
	10-Fold	18	3	16	5	17	4	relapse
		10	29	10	29	3	36	non-relapse
Percentage Split	%66	5	3	4	4	5	3	relapse
		5	7	2	10	2	10	non-relapse
	%70	5	3	4	4	5	3	relapse
		5	5	2	8	1	9	non-relapse

Table 4.14. MAE and RMSE values after classification of CNS Cancer dataset with CfsSubsetEval

CNS Cancer Dataset MAE and RMSE							
		KNN(k=3)		Naive Bayes		SMO	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
Cross Validation	2-Fold	0.3043	0.4605	0.2681	0.5061	0.15	0.3873
	4-Fold	0.2755	0.4181	0.2332	0.474	0.1167	0.3416
	6-Fold	0.2533	0.377	0.2156	0.4482	0.0667	0.2582
	8-Fold	0.2641	0.3913	0.2497	0.4841	0.1167	0.3416
	10-Fold	0.2585	0.3936	0.2466	0.4808	0.1167	0.3416
Percentage Split	%66	0.3689	0.5032	0.2774	0.5113	0.25	0.5
	%70	0.3906	0.5246	0.3143	0.5462	0.2222	0.4714

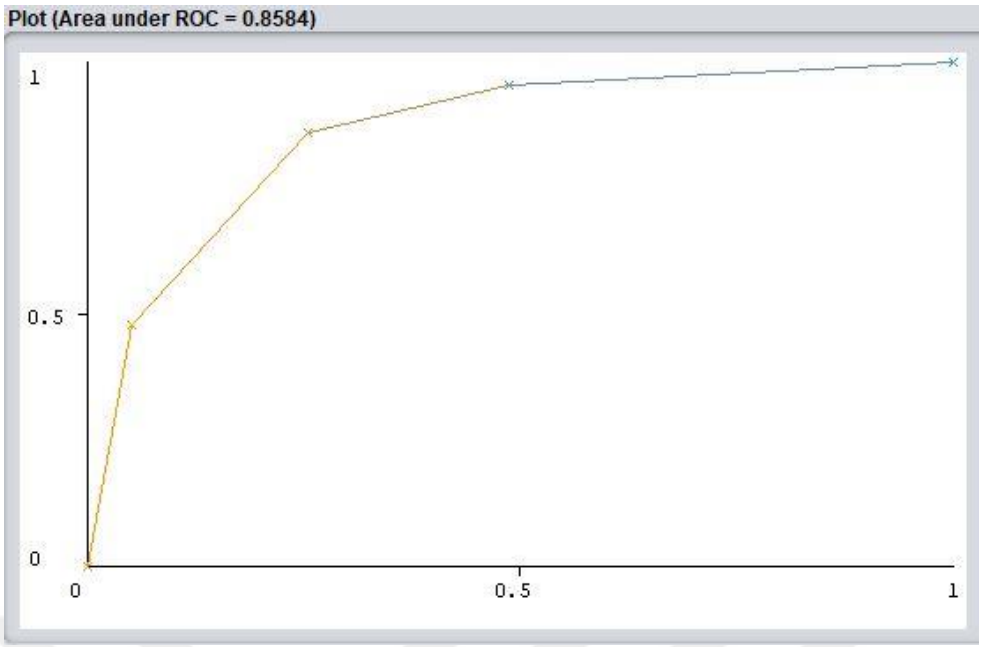

Figure 4.13. ROC curve and AUC value after classification by applying KNN to CNS Cancer dataset with CfsSubsetEval attribute selection (For 10-folds Cross Validation)

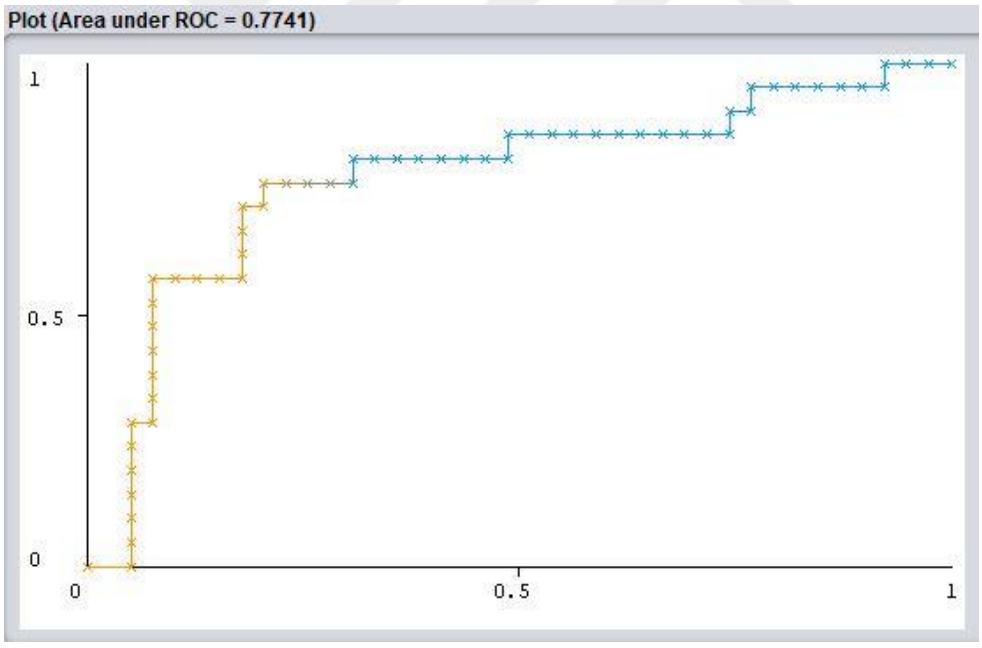

Figure 4.14. ROC curve and AUC value after classification by applying NB to CNS Cancer dataset with CfsSubsetEval attribute selection (For 10-folds Cross Validation)

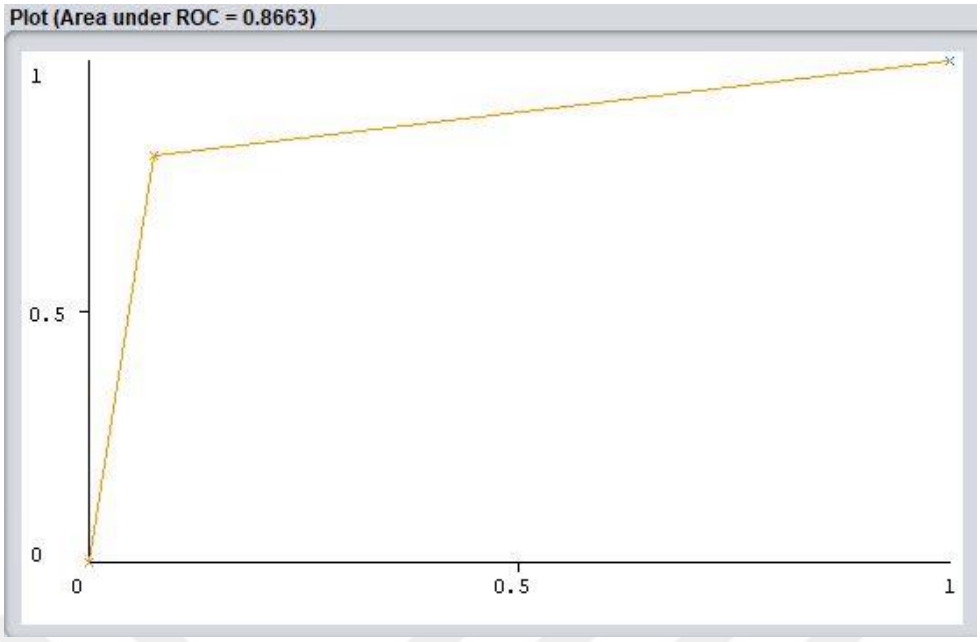

Figure 4.15. ROC curve and AUC value after classification by applying SMO to CNS Cancer dataset with CfsSubsetEval attribute selection (For 10-folds Cross Validation)

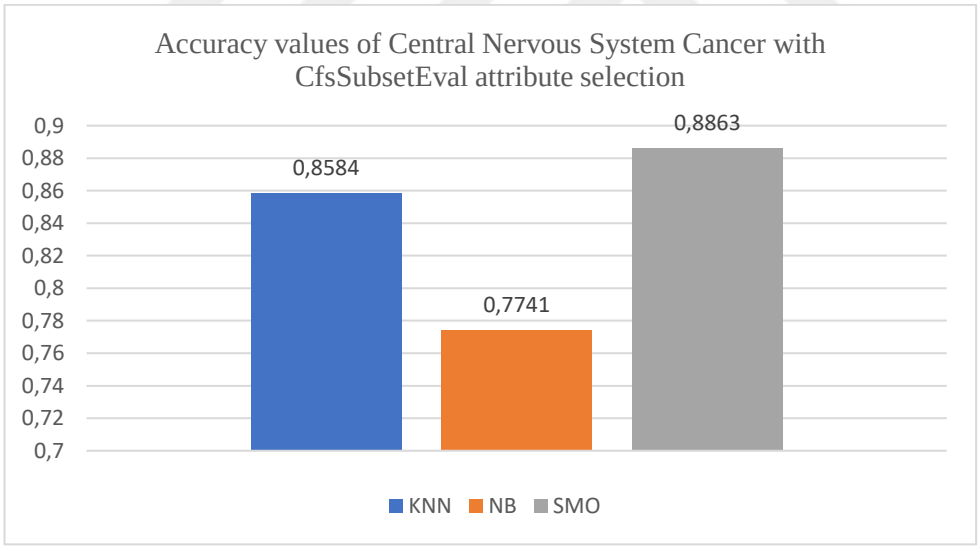

Figure 4.16. The graphical representation of the accuracy values of CNS Cancer with CfsSubsetEval attribute selection (For 10-folds Cross Validation)

4.1.2.2.ChiSquaredAttributeEval

ChiSquaredAttributeEval evaluates the worth of an attribute by computing the value of the chi-squared statistic with respect to the class. In this scenario, when selecting attributes, gene expressions whose rank values are different from zero are preferred. In this way, it is targeted to find genes that are thought to be more likely to give information about the disease. And then these reduced attributes and cancer data are subjected to classification. In this feature selection, the breast cancer attribute was reduced from 24482 to 819 and the CNS Cancer was reduced from 7130 to 73.

Table 4.15. Number of attributes with/without ChiSquaredAttributeEval

Number of attributes	No Attribute Selection	With Attribute Selection by ChiSquaredAttributeEval
Breast Cancer	24482	819
CNS Cancer	7130	73

a) The Breast Cancer classification results of the data set being reduced after the attribute selection made using ChiSquaredAttributeEval are in Table 4.12, Table 4.13. The AUC results of the classification algorithms applied are shown in Figure 4.17, Figure 4.18, Figure 4.19 and the graphical representation of the accuracy values is expressed as in Figure 4.20.

Table 4.16. Breast Cancer dataset classification success rate after attribute selection with ChiSquaredAttributeEval

Breast Cancer		Cross Validation					Percentage Split	
		2	4	6	8	10	66%	70%
KNN	K1	79.381	80.412	79.381	81.443	79.381	81.818	75.862
	K3	78.351	79.381	78.351	79.381	77.320	75.758	72.414
	K5	76.289	76.289	79.381	78.351	80.412	75.758	68.966
	K7	77.320	73.196	76.289	78.351	78.351	75.758	72.414
NB		67.010	59.794	59.794	57.732	58.763	60.606	65.517
SMO		82.474	82.474	81.443	83.505	83.505	69.697	75.862

Table 4.17. Confusion matrix values after classification of Breast Cancer data set with ChiSquaredAttributeEval

Breast Cancer Data Set Confusion Matrix								
		KNN(k=3)		Naive Bayes		SMO		Class
Cross Validation	2-Fold	38	8	19	27	39	7	relapse
		13	38	5	46	10	41	non-relapse
	4-Fold	37	9	8	38	38	8	relapse
		11	40	1	50	9	42	non-relapse
	6-Fold	37	9	7	39	36	10	relapse
		12	39	0	51	8	43	non-relapse
	8-Fold	36	10	5	41	38	8	relapse
		10	41	0	51	8	43	non-relapse
	10-Fold	36	10	6	40	36	10	relapse
		12	39	0	51	6	45	non-relapse
Percentage Split	%66	11	3	1	13	8	6	relapse
		5	14	0	19	4	15	non-relapse
	%70	9	2	1	10	7	4	relapse
		6	12	0	18	3	15	non-relapse

Table 4.18. MAE and RMSE values after classification of Breast Cancer dataset with ChiSquaredAttributeEval

Breast Cancer Dataset MAE and RMSE							
		KNN(k=3)		Naive Bayes		SMO	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
Cross Validation	2-Fold	0.2475	0.3893	0.3299	0.5744	0.1753	0.4186
	4-Fold	0.2395	0.3837	0.4021	0.6341	0.1753	0.4186
	6-Fold	0.2393	0.3867	0.3999	0.631	0.1856	0.4308
	8-Fold	0.2187	0.3625	0.4227	0.6501	0.1649	0.4061
	10-Fold	0.2459	0.3868	0.4124	0.6422	0.1649	0.4061
Percentage Split	%66	0.2751	0.4135	0.3939	0.6276	0.303	0.5505
	%70	0.2894	0.4146	0.3448	0.5872	0.2414	0.4913

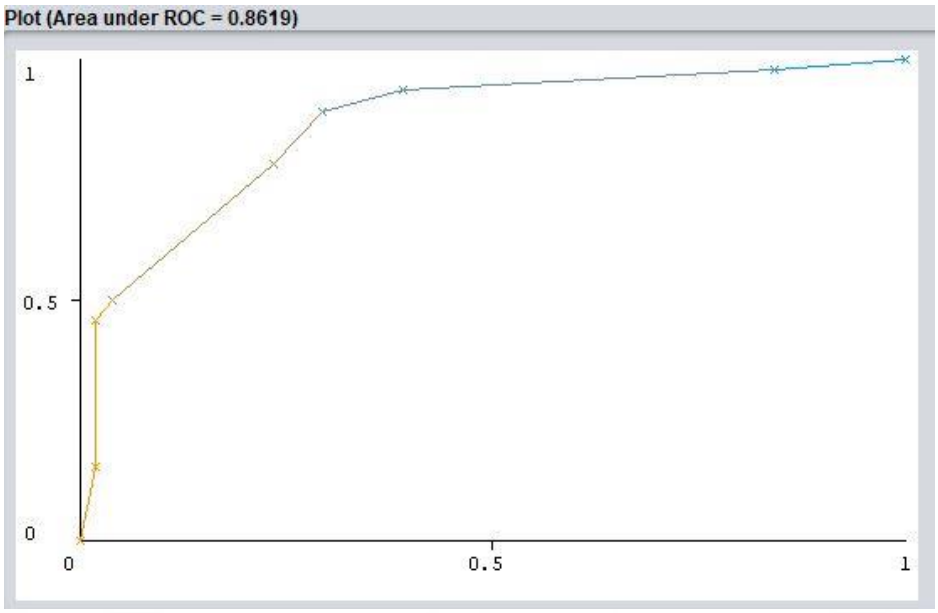

Figure 4.17. ROC curve and AUC value after classification by applying KNN to breast cancer dataset with ChiSquaredAttributeEval attribute selection (For 10-folds Cross Validation)

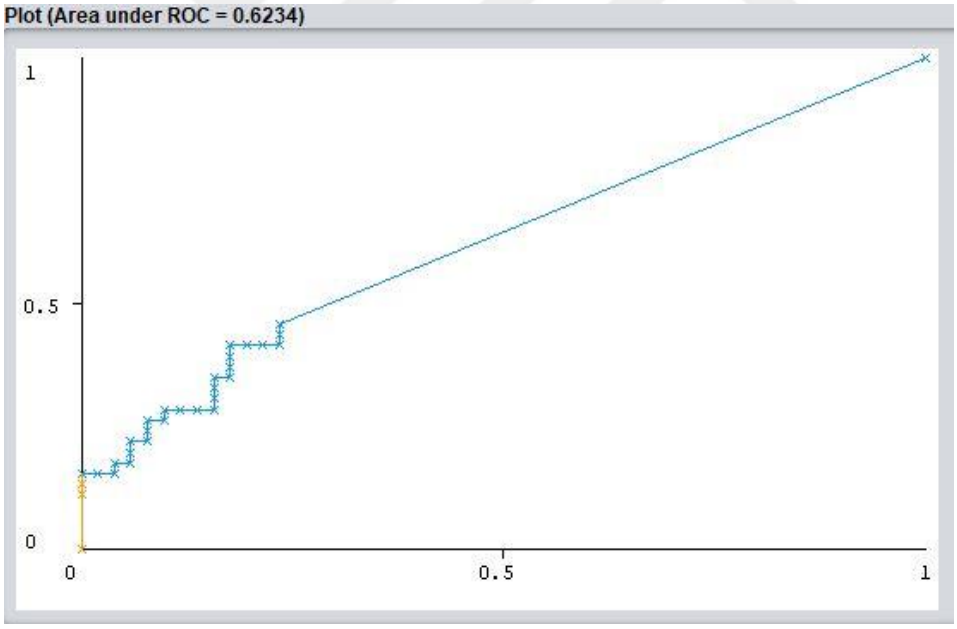

Figure 4.18. ROC curve and AUC value after classification by applying NB to breast cancer dataset with ChiSquaredAttributeEval attribute selection (For 10-folds Cross Validation)

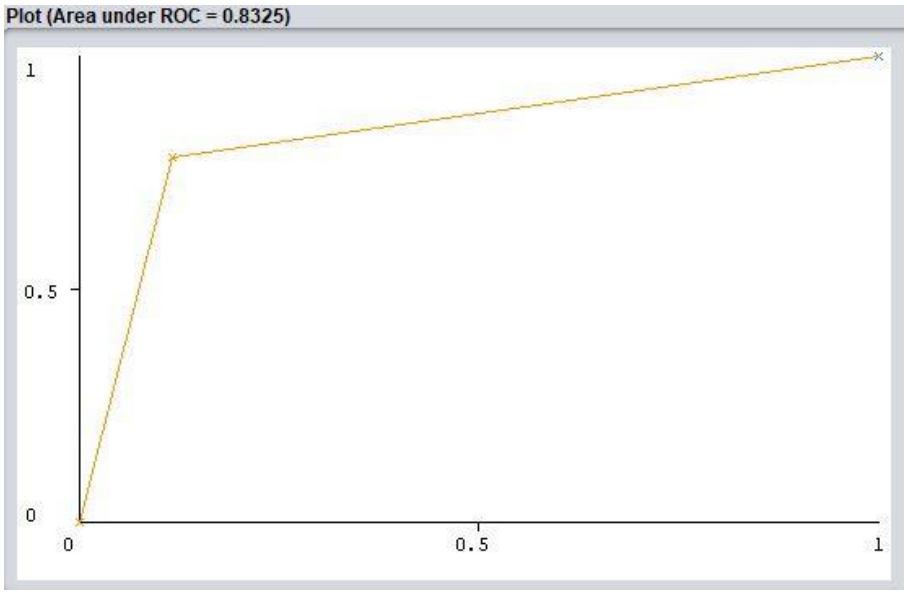

Figure 4.19. ROC curve and AUC value after classification by applying SMO to Breast Cancer dataset with ChiSquaredAttributeEval attribute selection (For 10-folds Cross Validation)

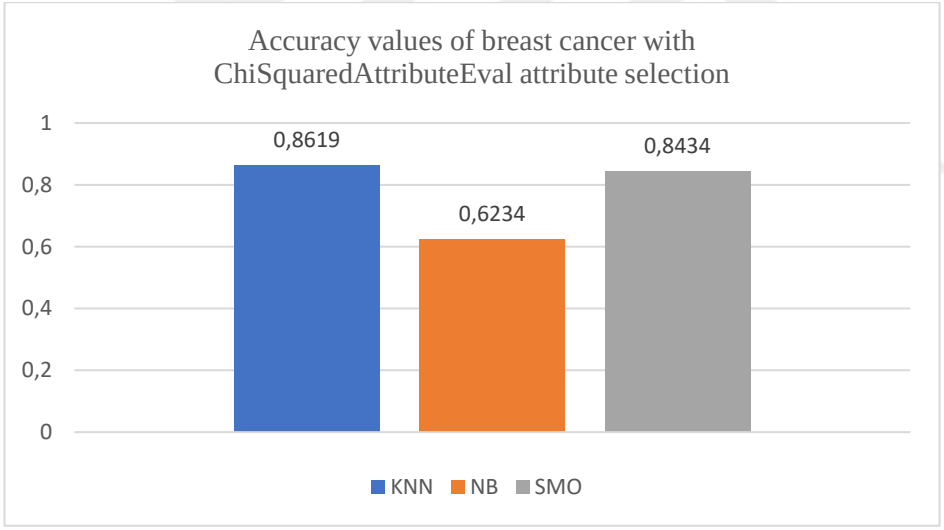

Figure 4.20. The graphical representation of the accuracy values of Breast Cancer with ChiSquaredAttributeEval attribute selection (For 10-folds Cross Validation)

b) The CNS Cancer classification results of the data set being reduced after the attribute selection made using CfsSubsetEval are in Table 4.9, Table 4.10. The AUC results of the classification algorithms applied are shown in Figure 4.13, Figure 4.14, Figure 4.15 and the graphical representation of the accuracy values is expressed as in Figure 4.16.

Table 4.19. CNS Cancer dataset classification success rate after attribute selection with ChiSquaredAttributeEval

CNS Cancer		Cross Validation					Percentage Split	
		2	4	6	8	10	66%	70%
KNN	K1	75.000	75.000	76.667	78.333	76.667	60.000	55.556
	K3	73.333	73.333	75.000	75.000	73.333	55.000	50.000
	K5	71.667	73.333	73.333	76.667	76.667	60.000	55.556
	K7	73.333	76.667	78.333	78.333	75.000	55.000	50.000
NB		75.000	75.000	75.000	71.667	71.667	65.000	61.111
SMO		81.667	86.667	90.000	91.667	91.667	70.000	66.667

Table 4.20. Confusion matrix values after classification of CNS Cancer data set with ChiSquaredAttributeEval

CNS Cancer Dataset Confusion Matrix								
		KNN(k=3)		Naive Bayes		SMO		Class
Cross Validation	2-Fold	16	5	11	10	15	6	relapse
		11	28	5	34	5	34	non-relapse
	4-Fold	16	5	13	8	15	6	relapse
		11	28	7	32	2	37	non-relapse
	6-Fold	17	4	14	7	18	3	relapse
		11	28	8	31	3	36	non-relapse
	8-Fold	17	4	13	8	19	2	relapse
		11	28	9	30	3	36	non-relapse
	10-Fold	16	5	13	8	19	2	relapse
		11	28	9	30	3	36	non-relapse
Percentage Split	%66	4	4	3	5	4	4	relapse
		5	7	2	10	2	10	non-relapse
	%70	4	4	3	5	4	4	relapse
		5	5	2	8	2	8	non-relapse

Table 4.21. MAE and RMSE values after classification of CNS Cancer dataset with ChiSquaredAttributeEval

CNS Cancer Dataset MAE and RMSE							
		KNN(k=3)		Naive Bayes		SMO	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
Cross Validation	2-Fold	0.288	0.4425	0.2488	0.4977	0.1833	0.4282
	4-Fold	0.2536	0.4304	0.2548	0.4927	0.1333	0.3651
	6-Fold	0.2368	0.4022	0.2346	0.4704	0.1	0.3162
	8-Fold	0.2257	0.3976	0.2824	0.5299	0.0833	0.2887
	10-Fold	0.2256	0.3931	0.28	0.5241	0.0833	0.2887
Percentage Split	%66	0.3852	0.5396	0.3459	0.5848	0.3	0.5477
	%70	0.4271	0.5689	0.3826	0.614	0.3333	0.5774

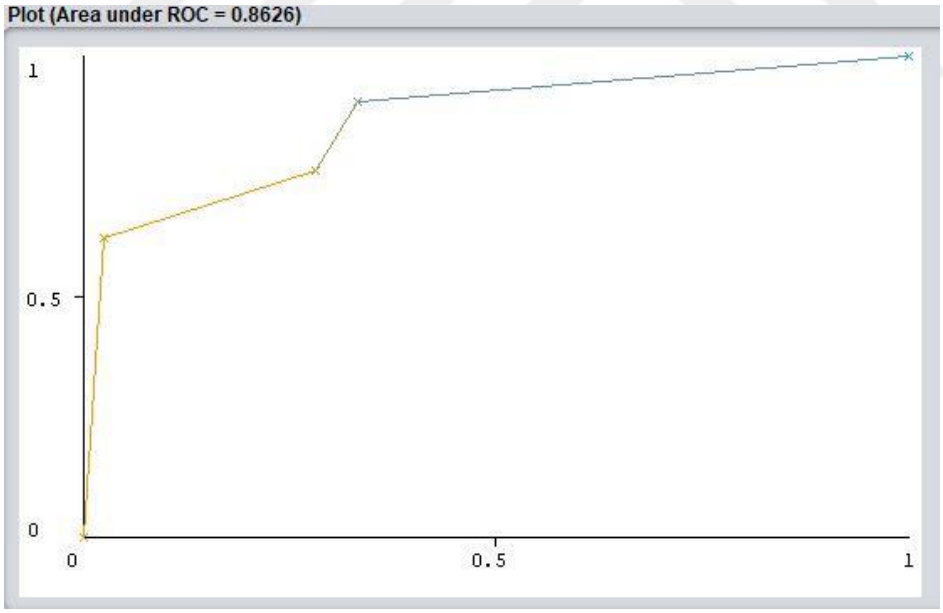

Figure 4.21. ROC curve and AUC value after classification by applying KNN to CNS Cancer dataset with ChiSquaredAttributeEval attribute selection (For 10-folds Cross Validation)

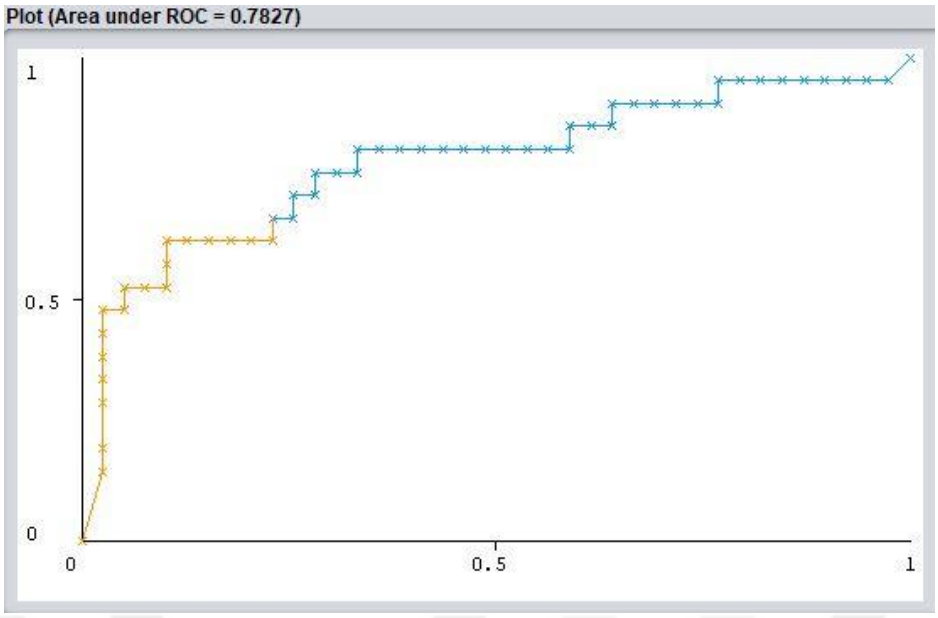

Figure 4.22. ROC curve and AUC value after classification by applying NB to CNS Cancer dataset with ChiSquaredAttributeEval attribute selection (For 10-folds Cross Validation)

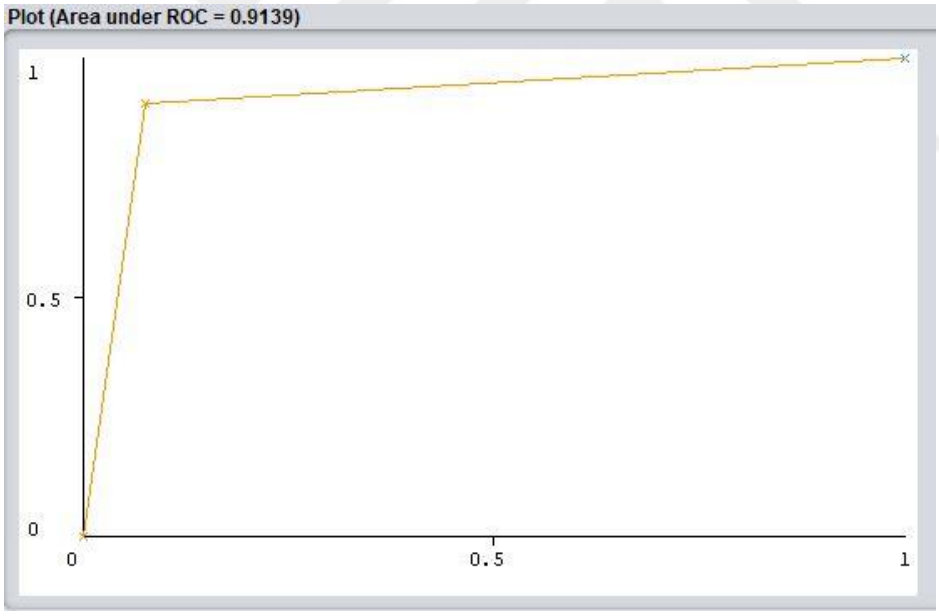

Figure 4.23. ROC curve and AUC value after classification by applying SMO to CNS Cancer dataset with ChiSquaredAttributeEval attribute selection (For 10-folds Cross Validation)

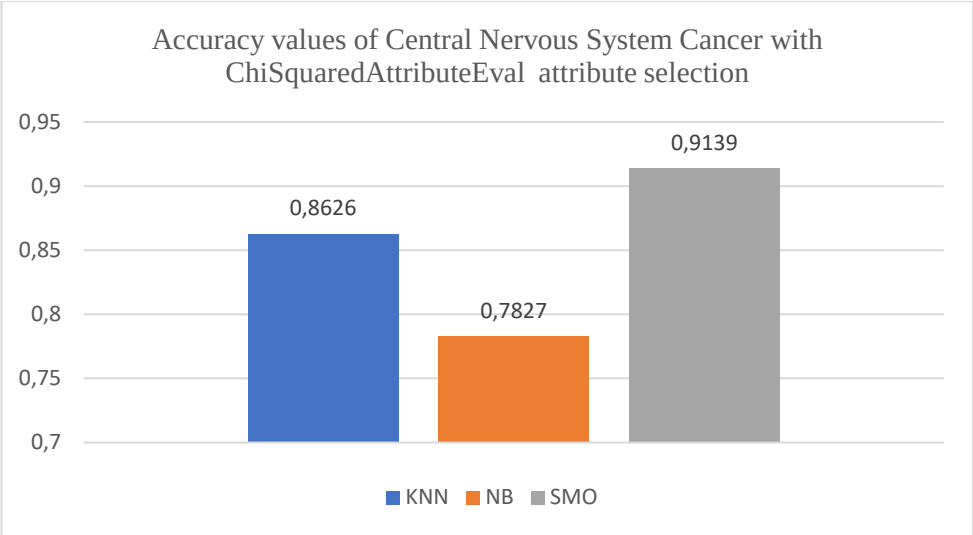

Figure 4.24. The graphical representation of the accuracy values of CNS Cancer with ChiSquaredAttributeEval attribute selection (For 10-folds Cross Validation)

4.1.2.3. WrapperSubsetEval

WrapperSubsetEval evaluates attribute sets by using a learning scheme. The "wrapper" method wraps a classifier in a cross-validation loop: it searches through the attribute space and uses the classifier to find a good attribute set. In this scenario the wrapper method uses a classifier in itself. SMO was accepted as a classifier with better performance in previous classifications.

In this way, it is targeted to find genes that are thought to be more likely to give information about the disease. And then these reduced attributes and cancer data are subjected to classification.

In this feature selection, the breast cancer attribute was reduced from 24482 to 7 and the CNS cancer was reduced from 7130 to 15.

Table 4.22. Number of attributes with/without WrapperSubsetEval

Number of attributes	No Attribute Selection	With Attribute Selection by WrapperSubsetEval
Breast Cancer	24482	7
CNS Cancer	7130	15

a) The Breast Cancer classification results of the data set being reduced after the attribute selection made using WrapperSubsetEval are in Table 4.17, Table 4.18. The AUC results of the classification algorithms applied are shown in Figure 4.25, Figure 4.26, Figure 4.27 and the graphical representation of the accuracy values is expressed as in Figure 4.28.

Table 4.23. Breast Cancer dataset classification success rate after attribute selection with WrapperSubsetEval

Breast Cancer		Cross Validation					Percentage Split	
		2	4	6	8	10	66%	70%
KNN	K1	75.258	78.351	77.320	73.196	75.258	78.788	75.862
	K3	80.412	78.351	77.320	75.258	77.320	66.667	68.966
	K5	79.381	81.443	80.412	84.536	81.443	69.697	72.414
	K7	76.289	81.443	76.289	78.351	80.412	75.758	75.862
NB		77.320	78.351	78.351	79.381	79.381	78.788	79.310
SMO		80.412	85.567	88.660	89.691	89.691	84.849	82.759

Table 4.24. Confusion matrix values after classification of Breast Cancer data set with WrapperSubsetEval

Breast Cancer Data Set Confusion Matrix								
		KNN(k=3)		Naive Bayes		SMO		Class
Cross Validation	2-Fold	33	13	32	14	31	15	relapse
		6	45	8	43	4	47	non-relapse
	4-Fold	35	11	34	12	36	10	relapse
		10	41	9	42	4	47	non-relapse
	6-Fold	33	13	34	12	39	7	relapse
		9	42	9	42	4	47	non-relapse
	8-Fold	33	13	34	12	40	6	relapse
		11	40	8	43	4	47	non-relapse
	10-Fold	35	11	35	11	40	6	relapse
		11	40	9	42	4	47	non-relapse
Percentage Split	%66	10	4	11	3	12	2	relapse
		7	12	4	15	3	16	non-relapse
	%70	9	2	9	2	9	2	relapse
		7	11	4	14	3	15	non-relapse

Table 4.25. MAE and RMSE values after classification of Breast Cancer dataset with WrapperSubsetEval

Breast Cancer Dataset MAE and RMSE							
		KNN(k=3)		Naive Bayes		SMO	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
Cross Validation	2-Fold	0.3085	0.4044	0.2777	0.4245	0.1959	0.4426
	4-Fold	0.277	0.3914	0.2698	0.4193	0.1443	0.3799
	6-Fold	0.2938	0.4127	0.2763	0.4211	0.1134	0.3368
	8-Fold	0.2835	0.4029	0.2649	0.4083	0.1031	0.3211
	10-Fold	0.2937	0.4072	0.2676	0.4077	0.1031	0.3211
Percentage Split	%66	0.3251	0.4411	0.2879	0.44	0.1515	0.3892
	%70	0.3122	0.4496	0.3041	0.4574	0.1724	0.4152

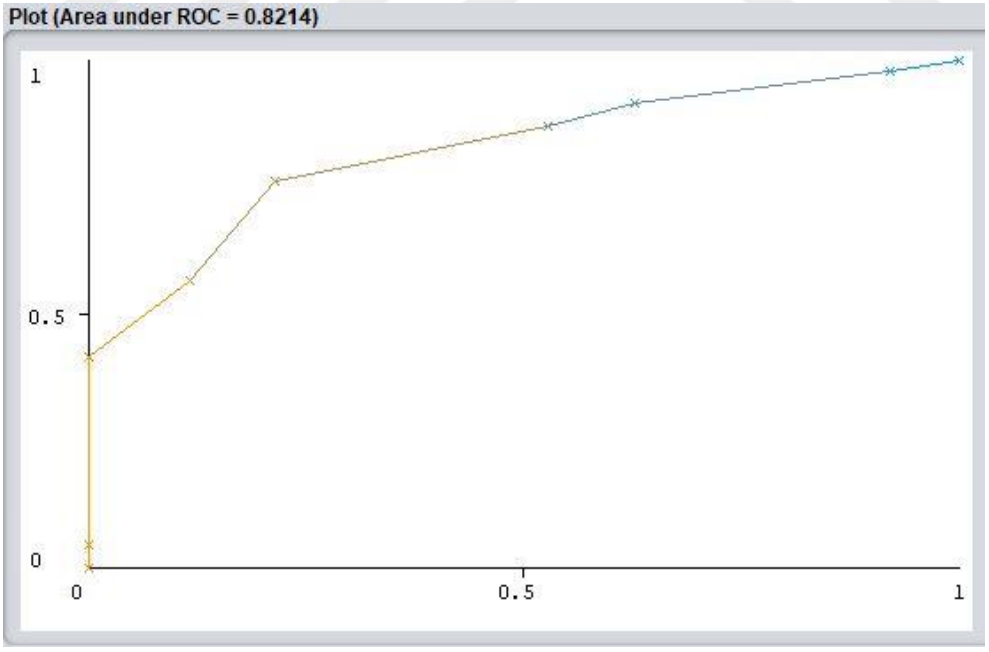

Figure 4.25. ROC curve and AUC value after classification by applying KNN to breast cancer dataset with WrapperSubsetEval attribute selection (For 10-folds Cross Validation)

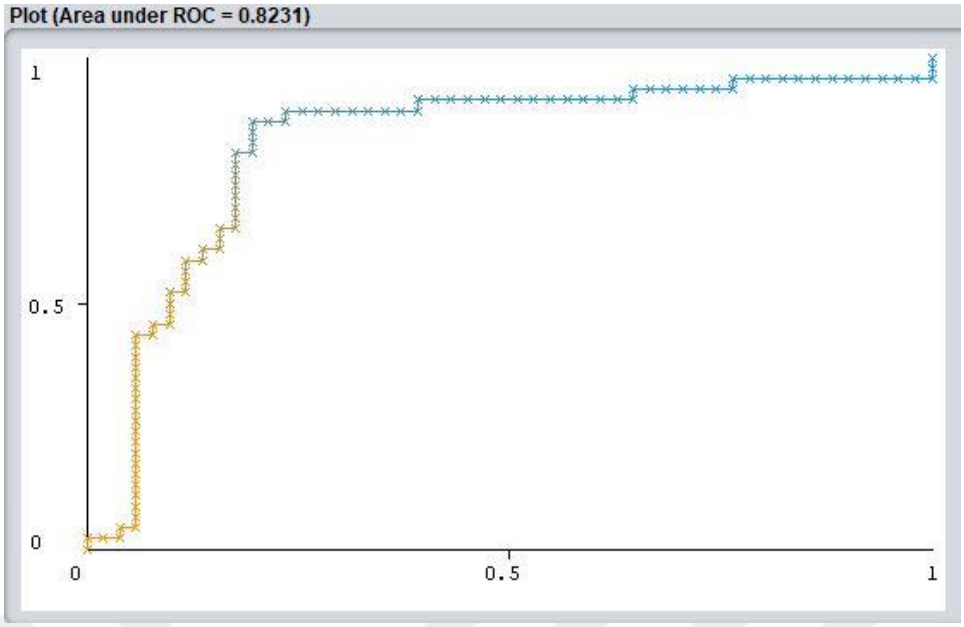

Figure 4.26. ROC curve and AUC value after classification by applying NB to breast cancer dataset with WrapperSubsetEval attribute selection (For 10-folds Cross Validation)

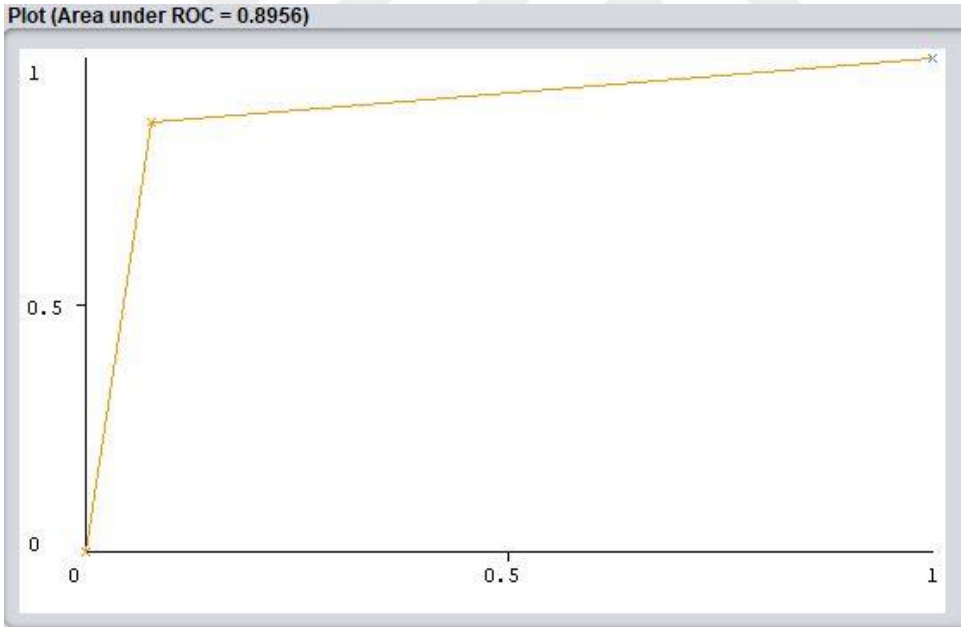

Figure 4.27. ROC curve and AUC value after classification by applying SMO to breast cancer dataset with WrapperSubsetEval attribute selection (For 10-folds Cross Validation)

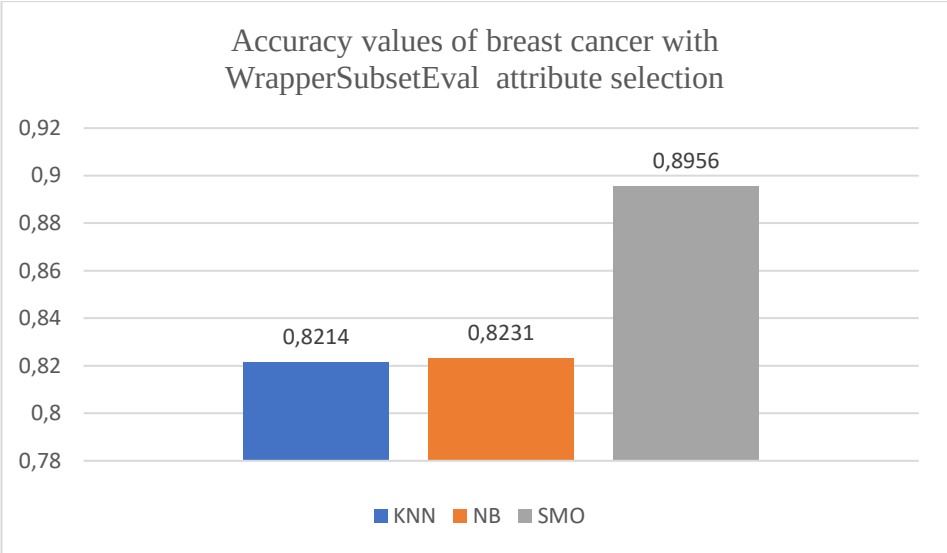

Figure 4.28. The graphical representation of the accuracy values of Breast Cancer with WrapperSubsetEval attribute selection (For 10-folds Cross Validation)

b) The CNS Cancer classification results of the data set being reduced after the attribute selection made using WrapperSubsetEval are in Table 4.9, Table 4.10. The AUC results of the classification algorithms applied are shown in Figure 4.13, Figure 4.14, Figure 4.15 and the graphical representation of the accuracy values is expressed as in Figure 4.16.

Table 4.26. CNS Cancer dataset classification success rate after attribute selection with WrapperSubsetEval

CNS Cancer		Cross Validation					Percentage Split	
		2	4	6	8	10	66%	70%
KNN	K1	78.333	73.333	80.000	76.667	81.667	75.000	72.222
	K3	78.333	83.333	80.000	78.333	75.000	70.000	66.667
	K5	73.333	75.000	71.667	80.000	80.000	65.000	61.111
	K7	73.333	75.000	75.000	73.333	73.333	60.000	55.556
NB		71.667	70.000	73.333	76.667	73.333	70.000	66.667
SMO		78.333	90.000	93.333	93.333	93.333	70.000	77.778

Table 4.27. Confusion matrix values after classification of CNS Cancer data set with WrapperSubsetEval

CNS Cancer Data Set Confusion Matrix								
		KNN(k=3)		Naive Bayes		SMO		Class
Cross Validation	2-Fold	9	12	8	13	8	13	relapse
		1	38	4	35	0	39	non-relapse
	4-Fold	13	8	7	14	15	6	relapse
		2	37	4	35	0	39	non-relapse
	6-Fold	13	8	9	12	17	4	relapse
		4	35	4	35	0	39	non-relapse
	8-Fold	11	10	10	11	17	4	relapse
		3	36	3	36	0	39	non-relapse
	10-Fold	10	11	8	13	17	4	relapse
		4	35	3	36	0	39	non-relapse
Percentage Split	%66	3	5	2	6	2	6	relapse
		1	11	0	12	0	12	non-relapse
	%70	3	5	2	6	4	4	relapse
		1	9	0	10	0	10	non-relapse

Table 4.28. MAE and RMSE values after classification of CNS Cancer dataset with WrapperSubsetEval

CNS Cancer Dataset MAE and RMSE							
		KNN(k=3)		Naive Bayes		SMO	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
Cross Validation	2-Fold	0.2772	0.4001	0.2876	0.5118	0.2167	0.4655
	4-Fold	0.2646	0.3724	0.2923	0.5257	0.1	0.3162
	6-Fold	0.2752	0.3869	0.2721	0.51	0.0667	0.2582
	8-Fold	0.2806	0.3846	0.2539	0.4876	0.0667	0.2582
	10-Fold	0.275	0.4007	0.2646	0.5013	0.0667	0.2582
Percentage Split	%66	0.4016	0.4707	0.3194	0.5528	0.3	0.5477
	%70	0.4089	0.4833	0.3589	0.5859	0.2222	0.4714

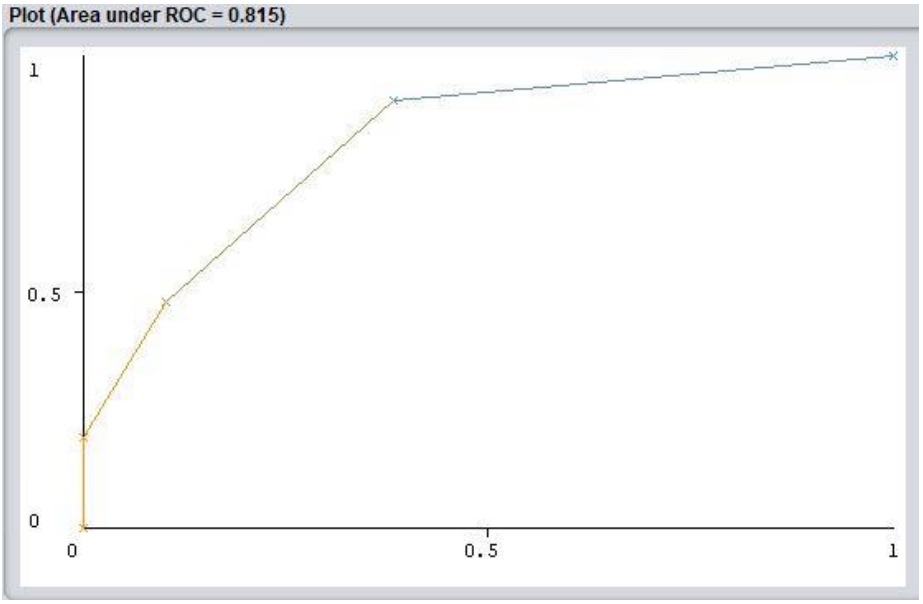

Figure 4.29. ROC curve and AUC value after classification by applying KNN to CNS Cancer dataset with WrapperSubsetEval attribute selection (For 10-folds Cross Validation)

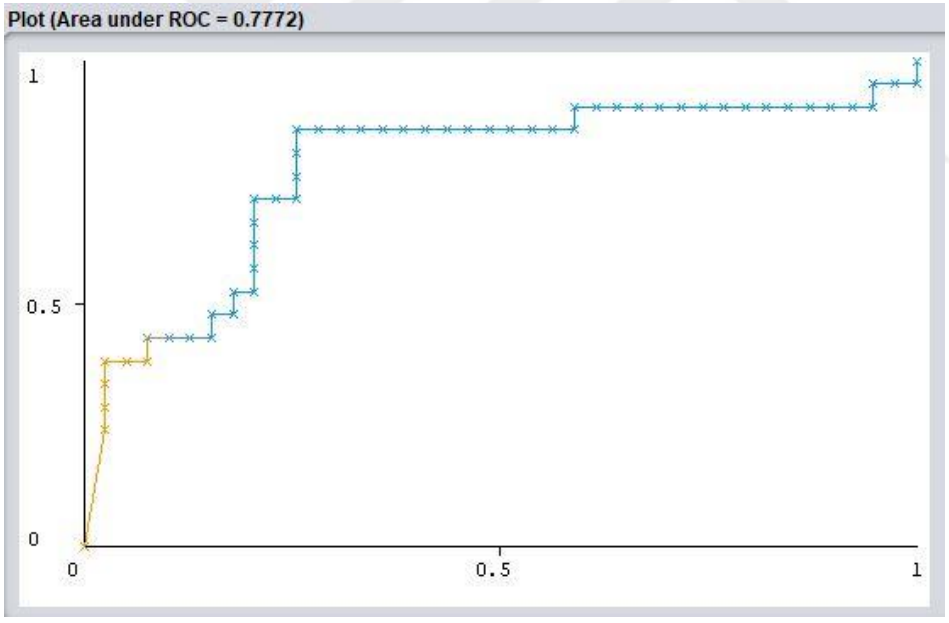

Figure 4.30. ROC curve and AUC value after classification by applying NB to CNS Cancer dataset with WrapperSubsetEval attribute selection (For 10-folds Cross Validation)

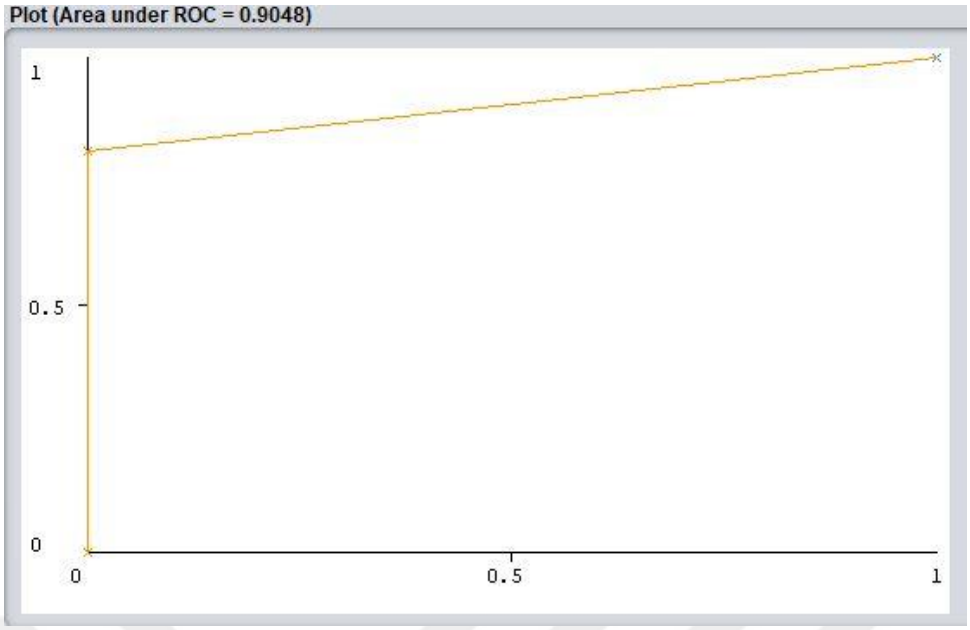

Figure 4.31. ROC curve and AUC value after classification by applying SMO to CNS Cancer dataset with WrapperSubsetEval attribute selection

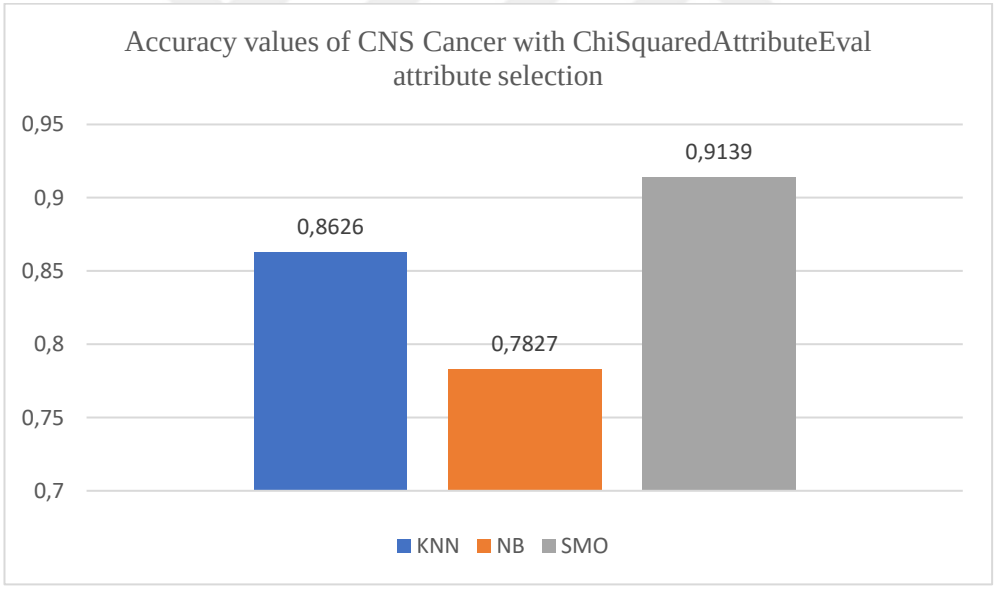

Figure 4.32. The graphical representation of the accuracy values of CNS Cancer with WrapperSubsetEval attribute selection (For 10-folds Cross Validation)

4.1.2.4. Classification of CNN Cancer Dataset using CfsSubsetEval Feature Selection Method with ANFIS Technique

After applying the CfsSubsetEval attribute selection method, the breast cancer attribute was reduced from 24482 to 138 and the CNS cancer was reduced from 7130 to 39.

Table 4.29. Number of attributes with/without CfsSubsetEval

Number of attributes	No Attribute Selection	With Attribute Selection by CfsSubsetEval
Breast Cancer	24482	138
CNS Cancer	7130	39

a) After applying the CfsSubsetEval attribute selection method to the Breast Cancer dataset, the results obtained from the application of artificial intelligence based ANFIS technique to the reduced dataset are as follows.

Table 4.30. Breast Cancer dataset classification success rate after attribute selection with CfsSubsetEval and ANFIS

Breast Cancer	Cross Validation					Percentage Split	
	2	4	6	8	10	66%	70%
ANFIS	94.834	95.876	95.463	95.434	95.536	95.312	95.588

Table 4.31. Train MAE, Test MAE and Train RMSE, Test RMSE values of Breast Cancer after attribute selection with CfsSubsetEval and ANFIS

Breast Cancer Dataset MAE and RMSE						
		Train MAE	Test MAE	Train RMSE	Test RMSE	ACC
Cross Validation	2-Fold	0.1895	0.8102	0.0723	0.6439	94.8342
	4-Fold	0.1940	1.0705	0.0756	0.7870	95.8762
	6-Fold	0.2066	1.1288	0.0856	0.8830	95.4630
	8-Fold	0.2054	1.1891	0.0847	0.8809	95.4342
	10-Fold	0.2064	0.8556	0.0853	0.6876	95.5364
Percentage Split	%66	0.2073	1.1411	0.0859	0.9156	95.3125
	%70	0.2011	0.8300	0.0809	0.7112	95.5882

b) After applying the CfsSubsetEval attribute selection method to the CNS Cancer dataset, the results obtained from the application of artificial intelligence based ANFIS technique to the reduced dataset are as follows.

Table 4.32. CNS Cancer dataset classification success rate after attribute selection with CfsSubsetEval and ANFIS

CNS Cancer	Cross Validation					Percentage Split	
	2	4	6	8	10	66%	70%
ANFIS	100	100	99.3333	99.5192	98.3333	100	100

Table 4.33. Train MAE, Test MAE and Train RMSE, Test RMSE values of CNS Cancer after attribute selection with CfsSubsetEval and ANFIS

CNS Cancer Dataset MAE and RMSE						
		Train MAE	Test MAE	Train RMSE	Test RMSE	ACC
Cross Validation	2-Fold	0.1895	0.8102	0.0723	0.6439	94.8342
	4-Fold	0.1006	1.7509	0.0743	1.3419	100
	6-Fold	0.1560	1.1853	0.1205	0.7966	99.3333
	8-Fold	0.1646	1.3019	0.1247	0.8892	99.5192
	10-Fold	0.1785	1.1685	0.1358	0.8314	98.3333
Percentage Split	%66	0.2073	1.1411	0.0859	0.9156	95.3125
	%70	0.0431	1.2127	0.0317	0.9223	100

CHAPTER 5

CONCLUSION AND EVALUATION

In this study, microarray gene expression of breast and central nervous system cancer cells was first examined. DVM, K-Nearest Neighbourhood, Naive Bayes classification methods were applied through WEKA program without any touching (i.e. no attribute selection) of these cancer data. The microarray data set of these two cancer types is shown in Table 5.1.

Table 5.1. Breast cancer and CNS cancer data set information

	Number of Sample	Number of Patients with Cancer	Number of Normal Patients	Number of Genes
Breast Cancer	97	46	51	24482
CNS Cancer	60	21	39	7130

While K-Nearest Neighbourhood classification method was applied to these two cancer datasets, 4 different k values were applied as 1,3,5,7. In addition, both cross validation and percentage split methods were used as training and test sets for these two types of cancer. For cross validation, 4 different values were applied, namely 2,4,6,8,10, and 2 different ratios, 66% and 70%, were used for percentage split. The success rates of the classification process applied to data sets without attribute selection are shown in Table 5.2 and Table 5.3.

Table 5.2. Success rate of classification of breast cancer data set without attribute selection

Breast Cancer		Cross Validation					Percentage Split	
		2	4	6	8	10	66%	70%
KNN	K1	57.732	55.670	59.794	58.763	60.825	57.576	48.276
	K3	59.794	57.732	58.763	58.763	58.763	63.636	65.517
	K5	71.134	64.949	59.794	61.856	62.887	60.606	65.517
	K7	69.072	65.979	61.856	67.010	67.010	60.606	62.069
NB		51.546	53.608	53.608	53.608	54.639	57.576	62.069
SMO		65.979	74.227	65.979	71.134	68.041	60.606	58.621

Table 5.3. Success rate of classification of CNS cancer data set without attribute selection

CNS Cancer		Cross Validation					Percentage Split	
		2	4	6	8	10	66%	70%
KNN	K1	51.667	51.667	60.000	60.000	56.667	40.000	38.889
	K3	61.667	61.667	61.667	61.667	56.667	40.000	33.333
	K5	63.333	58.333	65.000	60.000	65.000	40.000	27.778
	K7	63.333	58.333	63.333	56.667	63.333	55.000	38.889
NB		68.333	56.667	63.333	61.667	61.667	60.000	55.556
SMO		65.000	60.000	70.000	68.333	68.333	55.000	44.444

The success rates obtained from this procedure were found to be low. Afterwards, it was aimed to find the genes which are thought to be related to cancer by making feature selection process for these two types of cancer.

Three different methods were used for attribute selection. These feature selection algorithms; CfsSubsetEval, ChiSquaredAttributeEval, and WrapperSubsetEval.

For both Breast Cancer and CNS cancer; a significant increase was observed in the success rates obtained after the classification process was applied to the dataset obtained by selecting attribute with CfsSubsetEval compared to the success rate after classification without applying attribute selection. In the classification of Breast Cancer; the average success rate increased from 60,667 to 82.061 when the k value was selected as 3 in the classification made with KNN. The success rate in NB classification increased from 53,401 to 75,333. The success rate in SMO classification increased from 69,072 to 88,666. In the classification of CNS Cancer; the average

success rate increased from 60,667 to 74,333 when the k value was selected as 3 in the classification made with KNN. The success rate in NB classification increased from 62,333 to 75,333. The success rate in SMO classification increased from 66,333 to 88,666.

For both Breast Cancer and CNS cancer; a significant increase was observed in the success rates obtained after the classification process was applied to the dataset obtained by selecting attribute with ChiSquaredAttributeEval compared to the success rate after classification without applying attribute selection. In the classification of Breast Cancer; the average success rate increased from 60,667 to 78,556 when the k value was selected as 3 in the classification made with KNN. The success rate in NB classification increased from 53,401 to 60,618. The success rate in SMO classification increased from 69,072 to 82,680. In the classification of CNS Cancer; the average success rate increased from 60,667 to 73,999 when the k value was selected as 3 in the classification made with KNN. The success rate in NB classification increased from 62,333 to 73,666. The success rate in SMO classification increased from 66,333 to 88,333.

For both Breast Cancer and CNS cancer; a significant increase was observed in the success rates obtained after the classification process was applied to the dataset obtained by selecting attribute with WrapperSubsetEval compared to the success rate after classification without applying attribute selection. In the classification of Breast Cancer; the average success rate increased from 60,667 to 77,732 when the k value was selected as 3 in the classification made with KNN. The success rate in NB classification increased from 53,401 to 78,556. The success rate in SMO classification increased from 69,072 to 86,8042. In the classification of CNS Cancer; the average success rate increased from 60,667 to 78,999 when the k value was selected as 3 in the classification made with KNN. The success rate in NB classification increased from 62,333 to 73. The success rate in SMO classification increased from 66,333 to 89,666.

It is clearly seen that the success rates obtained by the classification process applied to the decreasing data set after attribute selection have been significantly increased for

both types of cancer in the success rates obtained as a result of the classification process applied without attribute selection.

In another study; ANFIS technique based on artificial intelligence was applied to both cancer datasets reduced by applying CfsSubsetEval attribute selection process. The success rates obtained as a result of this procedure are given in Table 5.4 and Table 5.5.

Table 5.4. Breast Cancer dataset classification success rate after attribute selection with CfsSubsetEval and ANFIS

Breast Cancer	Cross Validation					Percentage Split	
	2	4	6	8	10	66%	70%
ANFIS	94.834	95.876	95.463	95.434	95.536	95.312	95.588

Table 5.5 CNN Cancer dataset classification success rate after attribute selection with CfsSubsetEval and ANFIS

CNN Cancer	Cross Validation					Percentage Split	
	2	4	6	8	10	66%	70%
ANFIS	100	100	99.3333	99.5192	98.3333	100	100

The success rates achieved as a result of the classification process using the ANFIS technique were found to have an accuracy value of almost 100 per cent over 95 per cent. This shows that only statistical methods alone do not yield much higher success rates. The artificial intelligence-based ANFIS technique has been observed to yield much better results on these data than most statistical classification methods.

As a result, factors such as pre-processing of data sets, selection of classification algorithms and parameter selection and determination of the number of related attributes are the determining factors in the accuracy values of the classification. Making the right choices will yield better results in the field of bioinformatics and will shed light on clinical studies in the determination of disease for the future.

REFERENCES

Acı, M., & Avcı, M. (2011). K nearest neighbor reinforced expectation maximization method. *Expert Systems with Applications, 38*, 12585-12591.

Adaptive Neuro Fuzzy Inference System. (2019, 01 09). Retrieved 01 09, 2019, from https://en.wikipedia.org/wiki/Adaptive_neuro_fuzzy_inference_system

Alizadeh, A., Eisen, M., Davis, R., Ma, C., Lossos, I., Rosenwald, A., . . . Yu, X. (2000). Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature, 403*, 503-511.

Alon, U., Barkai, N., Notterman, D., Gish, K., Ybarra, S., Mack, D., & Levine, A. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences, 96*(12), 6745-6750.

Bal, S. H., & Budak, F. (Dergisi). Mikroarray Teknolojisi. *Uludağ Üniversitesi Tıp Fakültesi, 38*(3), 227-233.

Beni, G., & Wang, J. (1993). Swarm intelligence in cellular robotic systems. *In: Robots and Biological Systems: Towards a New Bionics? (Eds: G. Beni.,J. Wang) Springer Berlin Heidelberg*, 703-712.

Bier, F. F., von Nickisch-Rosenegk, M., Ehrentreich-Förster, E., ReiB, E., Henkel, J., Strehlow, R., & Andersen, D. (2008). *In:Biosensing for the 21st Century*, 433-453.

Binay, H. S. (2002). Yatırım Kararlarında Kaba Küme Yaklaşımı. Ankara Üniversitesi, Fen Bilimleri Entitüsü, Doktora Tezi, Ankara.

Bock, H. H. (2002). Data mining tasks and methods: Classification: the goal of classification. In *In Handbook of data mining and knowledge discovery* (pp. 254-258). Oxford University Press, Inc.

Bojarczuk, C. C., Lopes, H. S., Freitas, A. A., & Michalkiewicz, E. L. (2004). A Constrained-Syntax Genetic Programming System For Discovering Classification Rules: Application To Medical Data Sets. *Artificial Intelligence in Medicine, 30*(1), 27-48.

Bonabeau, E., & Theraulaz, G. (2000). In *Swarm smarts* (pp. 72-79). Scientific American.

Brieiman, L. (1996). Bagging predictors. *Machine learning, 24*(2), 123-140.

Brieman, L., Friedman, J. H., Olshen, R., & Stone, C. (1984). *Classification and regression trees.* UC San Diego, Chapman and Hall/CRC.

Brieman, L., Friedman, J. H., Olshen, R., & Stone, C. (1984). *Classification and regression trees (Belmont, CA: Wadsworth).* UC San Diego: Chapman and Hall/CRC.

Chen, H., Yang, B., Liu, J., & Liu, D. (2011). A support vector machine classifier with rough set-based feature selection for breast cancer diagnosis. *Expert Systems with Applications, 38*, 9014-9022.

De Falco, I., Della Cioppa, A., & Tarantino, E. (2002). Discovering interesting classification rules with genetic programming. *Applied Soft Computing,, 1*(4), 257-269.

De Falco, I., Della Ciopppa, A., & Tarantino, E. (2002). Discovering interesting classification rules with genetic programming. *Applied Soft Computing, 1*(4), 257-269.

De Jong, K. A., Spears, W. M., & Gordon, D. F. (1994). Using genetic algorithms for concept learning. *In: Genetic Algorithms for Machine Learning (Eds:K. A., De Jong, W. M., Spears, D. F., Gordon). Springer US*, 5-32.

Distilleries, D. (1999). Introduction to Data Mining-Discover Hidden Value in Your Databases. 7-9.

Efosa, C., & Akwukwuma, N. (2013). Knowledge based Fuzzy Inference System for Sepsis Diagnosis. *International Journal of Computational Science and Information Technology, 1*(3), 1-7.

Enas, G. G., & Choi, S. S. (1986). Choice of smoothing parameter and efficiency of k-nearest neighbor classification. *Computer & Mathematics with Applications, 12A*, 285-244.

Eren, Ö. (2008). *Alerjen Proteinlerin Otomatik Sınıflandırılması,.* Ankara: Başkent Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, .

Fayyad, U. (1997). Data mining and knowledge discovery in databases:implications for scientific databases. *Ninth International Conference on Scientific and Statistical Database Management*, (pp. 2,11). Olympia, WA,.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI magazine, 17*(3), 37.

Goldberg, D. E., Ohno, K., Esfarjani, K., & Kawazoe, Y. (1989). *Genetic algorithms in search, optimization, and machine learning, addison-wesley, reading, ma, 1989.* Computational Materi.

Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., . . . Caligiuri, M. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science, 286*(5439), 531.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter, 11*(1), 10-18.

Hamdan, H., & Garibaldi, M. (2010). Adaptive Neuro-Fuzzy Inference System (ANFIS) in Modelling Breast Cancer Survival. *IEEE World Congress on Computational Intelligence.*

Han , J., & Kamber, M. (2000). Data mining: concepts and techniques (the Morgan Kaufmann Series in data management systems).

Han, B., Li, L., Chen, Y., Zhu, L., & Dai, Q. (2011). A two step method to identify clinical outcome relevant genes with microarray data. *Journal of Biomedical Informatics, 44*, 229-238.

Han, J., & Kamber, M. (2001). Data mining: concepts and techniques (the Morgan Kaufmann Series in data management systems).

Han, J., & Kamber, M. (2018, 06 20). *Data mining: concepts and techniques*. Retrieved 06 20, 2018, from http://www.ict.griffith.edu.au/~vlad/teaching/kdd.d/6117cit.htm

Han, J., Cheng, H., Xin, D., & Yan, X. (2007). Frequent pattern mining: current status and future directions. *Data Mining and Knowledge Discovery*, 55,86.

Haykin, S., & Lippmann, R. (1994). Neural Networks, A Comprehensive Foundation. *International Journal of Neural Systems,, 5*(4), 363-364.

Herbrich, R., Graepel, T., & Obermayer, K. (1999). Support vector learning for ordinal regression. *9th International Conference on Artificial Neural Networks: ICANN '99, September 7-10 , 1999, IET Digital Library, 5 pp*, (pp. 97-102).

Ho, S. Y., Shu, L. S., & Chen, H. M. (1995). Intelligent genetic algorithm with a new intelligent crossover using orthogonal arrays. *Proceedings of the genetic and evolutionary computation conference,*, (pp. 289-296). Florida, USA.

Horne, B. G. (1993). Progress in supervised neural networks. *Signal Processing Magazine, IEEE, 10*(1), 8-39.

Huang, J., Fang, H., & Fan, X. (2010). Decision forest for classification of gene expression data",. *Computers in Biology and Medicine, 40*, 98-704.

İpekdal, K. (2011, 07"05). *Microarray Teknolojisi.* Retrieved 7 5, 2018, from http://yunus.hacettepe.edu.tr/~mergen/sunu/s_mikroarrayandecology.pdf

Işık, M. (2006). Bölünmeli Kümeleme Yöntemleri ile Veri Madenciliği Uygulamaları. Marmara Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, İstanbul, 73 s.

Jang, R. (1993). ANFIS :Adaptive-Network-Based Fuzzy Inference System. *IEEE Transactions on Systems, MAN, and Cybernetics, 23*(3).

Jang, R. (1995). Neuro-Fuzzy Modeling proceedings of the IEEE. *83*(3).

Kamrani, A., Rong, W., & Gonzalez, R. (2001). A genetic algorithm methodology for data mining and intelligent knowledge acquisition. *Computers & Industrial Engineering, 40*(4), 361-377.

Kantardzic, M. (2003). Chapter 9: Artificial Neural Networks Chapter 1-1.4.DataMining Concepts, Models, Methods and Algorithms, (Eds: J. Wiley ,Sons).

Kaur, H., Wasan, S. K., Al-Hegami, A. S., & Bhatnagar, V. (2006). A unified approach for discovery of interesting association rules in medical databases. *In Advances in Data Mining*, 53-63.

Korkem, E. (2013). Mikroarray Gen Ekspresyon Veri Setlerinde Random Forest ve Naive Bayes Sınıflama Yöntemleri Yaklaşımı. In *Hacettepe Üniversitesi, Sağlık* (p. 65). Ankara: Hacettepe Üniversitesi, Sağlık Bilimleri Enstitüsü, Biyoistatistik Programı, Yüksek Lisans Tezi.

Krishnan, M., Banerjee, S., Chakraborty, C., & Ajoy, K. (2010). Statistical analysis of mammographic features and its classification using support vector machine. *Expert Systems with Applications, 37*, 470-478.

Kuncheva, L. I. (1995). Editing for the k-nearest neighbors rule by a genetic algorithm. *Pattern Recognition Letters, 16*, 289-296.

Larose, D. T. (2005). *An introduction to data mining. Traduction et adaptation de Thierry Vallaud.*

Li, D., & Liu, C. (2010). A class possibility based kernel to increase classification accuracy for small data sets using support vector machines. *Expert Systems with Applications, 37*, 3104-3110.

Liu, H., Bebu, I., & Li, X. (2010). Microarray probes and probe sets. *Frontiers in bioscience (Elite edition), 2*, 325.

Lobenhofer, E., Auman, J., Blackshear, P., Boorman, G., Bushel, P., Cunningham, M., . . . Irwin, R. (2008). Gene expression response in target organ and whole blood varies as a function of target organ injury phenotype. *Genome Biology, 9*(6), 100.

Lu, J., Getz, G., Miska, A. E., Alvarez-Saavedra, J., Lamb, J., Peck, D., . . . Golub, R. T. (2005). MicroRNA expression profiles classify human cancers. *Nature, 435*, 834-838.

Lüleyap, H. Ü. (2008). *Moleküler Genetiğin Esasları.* Nobel Kitabevi.

MAE and RMSE. (2019, 06 02). Retrieved 06 02, 2019, from https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d

Michalewicz, Z. (2013). *Genetic algorithms+ data structures= evolution programs.* Springer Science & Business Media,.

Michalewicz, Z. (2013). *Genetic algorithms+ data structures= evolution programs.* Springer Science & Business Media,.

Mitra, S., & Acharya, T. (2005). In *Data mining: multimedia, soft computing, and bioinformatics* (p. 399). John Wiley & Sons,.

Murthy, S. K. (1998). Automatic construction of decision trees from data: A multi multi-disciplinary. *Data mining and knowledge discovery, 2*(4), 345-389.

Naive Bayes. (2018, 10 11). Retrieved 9 10, 2018, from https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

Nel, G. M. (2004). A Memetic Genetic Program For Knowledge Discovery.University of Pretoria, Faculty of Engineering, Master of Science, Pretoria,. 179.

Novakovic, J., Minic, M., & Veljovic, A. (2010). Genetic search for feature selection in rule induction algorithms. *In 18th Telecommunications forum TELFOR, November 23-25, 2010, Serbia, Belgrade, 4 pp.*, 1109-1112.

Özbakır, L., Baykasoğlu, A., & Kulluk , S. (2008). Rule extraction from neural networks via ant colony algorithm for data mining applications. *In Learning and Intelligent Optimization, Springer Berlin Heidelberg*, 177-191.

Özcan, G. S. (2014). *Bütünleştirici Modül Ağlarıyla Gen Düzenleme Analizi.* Ankara: Başkent Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi.

Özkan, Y. (2008). *Veri Madenciliği Yöntemleri.* İstanbul: Papatya Yayıncılık.

Parpinelli, R. S., Lopes, H. S., & Freitas, A. A. (2002). An ant colony algorithm for classification rule discovery. 191-208.

Patel, D., & Parekh, F. (2014). Flood Forecasting using Adaptive Neuro-Fuzzy Inference System. *International Journal of Engineering Trends and Technology, 12*(10), 510-514.

Patterson, D., Liu, F., Turner, D., Concepcion, A., & Lynch, R. (2008). Performance comparision of the data reduction system. *In Proceedings of the SPIE Symposium on Defense and Security*, (pp. 1-8). Orlando, FL.

Pawlak, Z. (1991). *Rough sets-theoretical aspect of reasoning about data.* Springer Netherlands.

Pawlak, Z., & Skowron, A. (1999). *Rough set rudiments.* Springer Berlin Heidelberg: Bulletin of the International Rough Set Society.

Perceptron. (2018, 12 12). Retrieved 12 12, 2018, from http://www.cogconfluence.com/wp-content/uploads/2013/12/notes.pdf

Perceptrons and Multi-layer Perceptrons. (2018, 12 20). Retrieved 12 20, 2018, from https://missinglink.ai/guides/neural-network-concepts/perceptrons-and-multi-layer-perceptrons-the-artificial-neuron-at-the-core-of-deep-learning/

Polat, M., & Karahan, A. (2009). *Multidisipliner yeni bir bilim dalı:biyoinformatik ve tıpta uygulamaları.* S.D.Ü Tıp Fak. Derg.,16(3): 41-50.

Power, E. (n.d.). Adaptive Neuro Fuzzy Inference System (ANFIS) For Fault Classification in the Transmission Lines. *Online Journal on Electronics and Electrical Engineering*, 164-169.

Pratama, M., Rajab, S., & Joo, E. (2011). Extended Approach of ANFIS in Cascade Control. *International Journal of Computer and Electrical Engineering, 3*(4), 1-5.

Quinlan, R. J. (1993). 4.5: Programs for machine learning. San Francisco, USA,: Morgan Kaufmann Publishers Inc.

Rastogi, R., & Shim, K. (2000). PUBLIC: A decision tree classifier that integrates building and pruning. *Data Mining and Knowledge Discovery,, 4*(4), 315-344.

Rezaei, K., Hosseini, R., & Mazinani, M. (2014). *A Fuzzy Inference System for Assessment of the Severity of the peptic ulcers.* Computer Science & Information Technology.

Romao, W., Freitas, A. A., & Gimenes, I. D. (2004). Discovering interesting knowledge from a science and technology database with a genetic algorithm. *Applied soft computing, 4*(2), 121-137.

Rouwhorst, S. E., & Engelbrecht, A. P. (2000). Searching The Forest: A Building Block Approach to Genetic Programming for Classification Problems in Data Mining Vrije Universiteit Amsterdam, Department of Mathematics and Informatics, Master of Thesis, Amsterdam,. 156.

Rouwhorst, S. E., & Engelbrecht, A. P. (2000). Searching The Forest: A Building Block Approach to Genetic Programming for Classification Problems in Data Mining, Vrije Universiteit Amsterdam, Department of Mathematics and Informatics, Master of Thesis, Amsterdam. 156.

Roy, S. (2005). Design of adaptive neuro-fuzzy inference system for predicting surface roughness in turning operation. *Journal of Scientific & Industrial Research, 64*, 653-659.

Schank, R. C. (1982). *Dynamic Memory: a theory of learning in computer and people.* New York: Cambridge University, Press.

Setiono, R., Leow, W. K., & Thong, J. Y. (2000). Opening the neural network blackbox: An algorithm for extracting rules from function approximating neural networks. *In 21st International Conference on Information Systems(ICIS).* Brisbane, Australia.

Shakya, K., Ruskin, H. J., Kerr, G., Crane, M., & Becker, J. (2010). Comparison of microarray preprocessing methods. *In Advances in Computational Biology*, 139-147.

Shenzhen University. (2018, 07 20). Retrieved 07 20, 2018, from http://csse.szu.edu.cn/cn/index.html

Silahtaroğlu, G. (2008). *Kavram ve Algoritmalarıyla Temel Veri Madenciliği.* İstanbul: Papatya Yayıncılık Eğitim İstanbul.

Svetlana, S. (n.d.). *Machine Learning with WEKA WEKA Explorer Tutorial for WEKA Version 3.4.3.* California State University, Sacramento.

Sydow, A. (1977). Pattern Recognition Principles. *ZAMM Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik, 57*(6), 353-354.

Şimşek, D. Ö. (2013). Mikroarray teknolojisi ve diş hekimliği'nde kullanımı. *atatürk Üniversitesi Diş Hekimliği Fakültesi Dergisi, 2013*(7).

Tan , K. C., Yu, Q., & Ang, J. H. (2006). A dual-objective evolutionary algorithm for rules extraction in data mining. *Computational optimization and applications, 34*(2), 273-294.

Tan, K. C., Yu, Q., & Ang, J. H. (2006). A coevolutionary algorithm for rules discovery in data mining. *International Journal of Systems Science, 37*(12), 835-864.

Tan, K. C., Yu, Q., Heng, C. M., & Lee, T. H. (2003). Evolutionary computing for knowledge discovery in medical diagnosis. *Artificial Intelligence in Medicine, 27*(2), 129-154.

Tan, K. C., Yu, Q., Heng, C. M., & Lee, T. H. (2003). Evolutionary computing for knowledge discovery in medical diagnosis. *Artificial Intelligence in Medicine, 27*(2), 129-154.

Tan, P. N., Steinbach, M., & Kumar, V. (2006). *Introduction to Data Mining.* Retrieved 06 23, 2018, from http://www-users.cs.umn.edu/~kumar/dmbook/index.php

Telcioğlu, M. B. (2007). Veri Madenciliğinde Genetik Programlama Temelli Yeni Bir Sınıflandırma Yaklaşımı ve Uygulaması. *Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Kayseri*, 145.

Tran, D. H., Ho, T. B., Pham, T. H., & Satou, K. (2011). MicroRNA Expression Profiles forClassification and Analysis of Tumor Samples. *IEICE Trans. Inf&Syst., 94*(3).

Uc, T., Karahoca, A., & Karahoca, D. (2013). Tuberculosis disease diagnosis by using adaptive neuro-fuzzy inference system and rough sets. *Neural Comput & Applications*, 471-483.

Uci Machine Learning Reporsitory. (2018, 07 20). Retrieved 7 20, 2018, from http://archive.ics.uci.edu/ml/datasets/

Understaing Support Vector Machine. (2018, 10 10). Retrieved 10 10, 2018, from https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/

Understanding AUC, ROC Curve. (2019, 06 20). Retrieved 06 20, 2019, from https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

Uran, G. (2005). *A Hybrid Heuristic Model for Classification Rule Discovery.* New York, NY, USA: Pace University.

Vahaplar, A., & İnceoğlu, M. M. (2001). Veri madenciliği ve elektronik ticaret. *Türkiye'de İnternet Konferansları.* Harbiye İstanbul.

Wang, X., & Gotoh, O. (2010). A robust Gene Selection Method for Microarray-based Cancer Classification. *Cancer Informatics, 9*, 15-30.

Weka. (2018, 08 10). Retrieved 08 10, 2018, from https://www.cs.waikato.ac.nz/~ml/weka/arff.html

What is a Decision Tree. (2018, 11 01). Retrieved 11 01, 2018, from https://www.techopedia.com/definition/28634/decision-tree

Witten, I. H., & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools And Techniques.* Morgan Kaufmann.

Wong, M. L., & Leung, K. S. (2006). *Data mining using grammar based genetic programming and applications (Vol. 3).* Springer Science & Business Media.

Yao, Y., & Zhou, B. (2008). Micro and macro evaluation of classification rules.In Cognitive Informatics. *7th IEEE International Conference on IEEE*, (pp. 441-448).

Zadeh, L. A. (195). Fuzzy sets. *İnformation and control*, *8*(3), 338-353.

Zhou, Z. H. (2013). Three perspectives of data mining,. *In: Artificial Intelligence Elsevier Science Publishers*, 139-146.

Zhu, Z., Ong, Y., & Dash, M. (2007). Markov Blanket-Embedded Genetic Algorithm for Gene Selection. *Pattern Recognition, 49*(11), 3236-3248. Retrieved June 20, 2018, from http://csse.szu.edu.cn/staff/zhuzx/Datasets.html

5. CURRICULUM VITAE

PERSONAL INFORMATION

NAME AND SURNAME	**MUMBUÇOĞLU, MEHMET ŞÜKRÜ**
NATIONALITY	Republic of Turkey
BIRTH PLACE AND DATE	16.09.1989, Şahinbey
MARRIAGE STATUS	Single
PHONE	0 538 453 42 73
EMAIL	msmumbucoglu@gmail.com

EDUCATION INFORMATION

	GRADUATED SCHOOL	GRADUATION YEAR
MASTER'S DEGREE	Hasan Kalyoncu University Institute of Science/Computer Engineering	-
BACHELOR DEGREE	İzmir Institute of Technology Institute of Science/Computer Engineering	2015
HIGH SCHOOL	Gaziantep High School	2006

JOB EXPERIENCE

	COMPANY	TASK
2018-	Sikke-Parifix	Senior Software Engineer
2015-2018	Belsoft Information Technology	Senior Software Engineer
2013-2015	Cybersoft Enformation Technology (ŞEKERBANK IT)	Software Engineer

FOREIGN LANGUAGE

English

ACTIVITIES

Swimming, Paragliding, Scuba diving