PREDICTION OF THE FOOTBALL MATCH RESULTS WITH USING MACHINE LEARNING ALGORITHMS

EMRE ALTUĞ ÇİMEN

PREDICTION OF THE FOOTBALL MATCH RESULTS WITH USING MACHINE LEARNING ALGORITHMS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES OF
ÇANKAYA UNIVERSITY

BY
EMRE ALTUĞ ÇİMEN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
IN
COMPUTER ENGINEERING
DEPARTMENT

Title of the Thesis: **Prediction of the Football Match Results with Using Machine Learning Algorithms**

Submitted By **Emre Altuğ ÇİMEN**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya University.

Prof. Dr. Can ÇOĞUN
Director

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

Prof. Dr. Kemal İDER
Head of Department

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

Assist. Prof. Dr. Erol ÖZÇELİK
Supervisor

Examination Date:

Examining Committee Members

Assist. Prof. Dr. Gökhan ŞENGÜL	Atılım Univ.	
Dr. Erol ÖZÇELİK	Çankaya Univ.	
Assist. Prof. Dr. Murat SARAN	Çankaya Univ.	

STATEMENT OF NON-PLAGIARISM PAGE

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.

Name, Last Name: Emre Altuğ ÇİMEN
Signature:
Date: 09.10.2019

ABSTRACT

PREDICTION OF THE FOOTBALL MATCH RESULTS WITH USING MACHINE LEARNING ALGORITHMS

ÇİMEN, Emre Altuğ
M.Sc., Department of Computer Engineering
Supervisor: Dr. Erol ÖZÇELİK

SEPTEMBER 2019, 80 pages

In this thesis, prediction results of the Spanish La Liga football matches were obtained by using three machine learning algorithms. The dataset includes four season match statistics and the results of these matches. In addition, this thesis investigated which performance parameters of the football game statistics affected the game results. Feature selection techniques were used to reduce the number of attributes. Three different classifiers which are artificial neural network, support vector machine and k- nearest neighborhood were used for prediction. Support vector machine classifier reached better results than the other classifiers when applied for the chosen fifteen attributes in the dataset.

Keywords: Machine Learning, Artifcial Neural Networks, Support Vector Machine

ÖZ

MAKİNE ÖĞREMESİ ALGORİTMALARI KULLANILARAK FUTBOL MAÇ SONUÇLARI TAHMİNİ

ÇİMEN, Emre Altuğ
Yüksek Lisans, Bilgisayar Mühendisliği Anabilim Dalı
Tez Yöneticisi: Dr. Erol ÖZÇELİK
EYLÜL 2019, 80 sayfa

Bu tez çalışmasında, İspanya Premier Ligi futbol maçlarının tahmin sonuçları üç makine öğrenme algoritması kullanılarak elde edilmiştir. Veri seti, dört sezon maç istatistiklerini ve sonuçlarını içermektedir. Ayrıca bu tez, maç sonuçlarına doğrudan etki edebilecek istatistiksel bazda performans parametrelerini incelemiştir. Öznitelik seçimi teknikleri, veri kümesine ait olan ilgisiz öznitelikleri azaltmak için kullanılmıştır. Tahmin sonuçları hesaplanırken üç farklı sınıflandırma algoritması kullanılmıştır. Bunlar yapay sinir ağları, destekçi vektör makineleri ve k-en yakın komşu algoritmasıdır. Destek vektör makinesi sınıflandırıcısı, veri kümesinde seçilen on beş özellik için uygulandığında diğer sınıflandırıcılara göre yüksek sonuçlar üretmiştir.

Anahtar Kelimeler: Makine Öğrenmesi, Yapay Sinir Ağları, Destekçi Vektör Makineleri.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to Dr. Erol ÖZÇELİK for his supervision, special guidance, suggestions, and encouragement through the development of this thesis.

It is a pleasure to express my special thanks to my family for their valuable support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AC	Away Team Corners
AF	Away Team Fouls Committed
ANN	Artificial Neural Network
AR	Away Team Red Cards
AS	Away Team Shots
AST	Away Team Shots on Target
AY	Away Team Yellow Cards
CAE	Classifier Attribute Eval
CSE	CfsSubsetEval
Date	Match Date
FTAG	Full Time Away Team Goals
FTHG	Full Time Home Team Goals
FTR	Full Time Result
FutBa	Football Bayes Network
HC	Home Team Corners
HF	Home Team Fouls Committed
HR	Home Team Red Cards
HS	Home Team Shots
HST	Home Team Shots on Target
HTHG	Half Time Home Team Goals
HTR	Half Time Result
HY	Home Team Yellow Cards
KNN	K Nearest Neighbor
LMA	Last Match Away Team Result
LMH	Last Match Home Team Result
ML	Machine Learning
NBA	National Basketball Association
SVM	Support Vector Machine

CHAPTER 1

INTRODUCTION

1.1 Background of the study

Football is the game that contains two teams of eleven players, using any part of their bodies except their hands and arms, attempting to maneuver the ball into the opposing team's side of the field in order to score a goal. The goalkeeper is allowed to handle the ball solely inside the penalty space. The team that scores more goals wins and the team that scores fewer goals lose. If both teams score the same goals or cannot score at all, the result is considered to be a draw.

Millions of people around the world follow the football matches played in various leagues. With this situation, various sponsorship deals, active betting organizations worldwide, uniform and material sales, match ticket sales, and advertisements are increasing day by day. The development of the football industry has brought clubs together that want to have significant financial support, and this situation increased the competition between clubs considerably. It is difficult to measure human accuracy when assessing the results of a football match. Precision depends on predicted hypotheses. The assumptions for different leagues and tournaments give different accuracy and they predict the outcomes for more league and tournaments than people. This makes the network difficult when compared to human accuracy. Machine learning (ML) is one of the smart ways to show promising results in the field of classification and prediction. Club managers and owners are striving hard to find classification models so that they can understand and develop the necessary strategies to win the game.

These models are based on a variety of factors related to games such as past match results, team performance indicators, and object information. This study focuses on the application of support vector machine (SVM), artificial neural network (ANN) and k-nearest neighbors algorithm (KNN) for the prediction of the performance parameters that can be directly related to the results. The main point of the statement is about which performance measures in a match predict that match's result. In that process, we identified the learning method used, the source of the data, and the appropriate method for model evaluation.

1.2 Problem

Prediction of the football game results with machine learning is a quite interesting topic for researchers. Therefore, there are many searches that have been done about this topic by various researchers [1-10]. Some researches concentrate on building expert systems and frameworks that can service expertise for the end users in order to predict the outcomes of the football matches. Some of the researches focus on prediction results of the one specific team regarding the player's performance attributes of the entire season. Some of them collect their performance data from the games because it is hard to find the reliable real-time performance data of both players and teams. Prediction of the results can be difficult regarding the nature of the football game because in-game motivation, audience support, injuries, referee decisions, fitness status and stars of each team can easily change the situation and affect the results directly. Every season after the transfer period, some of the players and staff members can change. Even the management side personnel may resign and leave the operation. This situation may also affect the success and can change the achievement criteria and goals of the team. Due to the nature of the data structure of the football game statistics, generally non-game effects cannot be assessed. Developed expert systems and software can only analyze the previous in-game statistics of the teams and players and then, predict the outcomes. At this point, discovering the most effective in-game performance parameters gains importance in the way of increasing prediction accuracy and reliable performance results. Naturally, many people who are football experts, football fans, and supporters have their controversial ideas about these performance parameters. However, there is not enough academically research about finding the performance measures that affect the results directly. Defining the most important performance parameters, in-game statistics can help the teams when they are organizing their squad and future transfer policy. Therefore, finding accurate and reliable results positively affect the quality of the games, teams, and most of the leagues.

In this research, we are going to classify and predict the football game results with high accuracy and also find high important attributes and variables information that can be directly related to the game results. Our dataset includes a wide range of attributes raw data which are gathered from the field statistics, and we assume to infer useful information that can be used for public and football related organizations.

1.3 The goal of the study

In this study, we are going to predict the La Liga season football matches results with related performance measures in order to achieve the highest percentage success rate. With combining the four season data between 2014 and 2018, this data is used to determine the parameters that directly affect the game results. On the pre-processing data stage, we implement the exhaustive search algorithm, best first search algorithm and ranker search algorithm to select the most effective attributes on the dataset. After that, we implement the classifiers to classify the full-time result (FTR) attribute, which contains the results of the football matches in the season. Expected result summary contains correctly classified instances and percentage, kappa statistic, mean absolute error, root mean squared error, relative absolute error percentage, root relative squared error, detailed accuracy by class including three classified attribute groups true positive rate, false positive rate, precision, recall, f-measure, Matthews correlation coefficient, and confusion matrix. Our aim is defining and listing the performance measures, and the classification rate of the FTR class attribute with high accuracy.

1.4 Significance of the study

In modern world, many varieties of companies are aimed to gather input values of the football matches thanks to the information systems, application software, heat sensors, and related technical equipment. They provide this information to the football clubs and football industry-related organizations to increase the quality of the competition, analyzing and interpreting the teams deeply, developing something new in-game strategies like formations, attacking and defending techniques and styles which can also contribute to the evolution of the football in century, new training techniques especially in youth setups of the clubs, statistical information of the players that can orient football clubs into the transfer periods. Therefore, without analyzing and interpreting this gathered raw data from fields, there is no main positively affecting difference for the data collection of traditional scout teams in the football clubs. Even, the wide amount of raw data can cause misunderstandings and difficulties in order to find useful information that can be evaluated. With this study, we analyze the raw data, which includes previous fixture matches of the seasons and input values gathered from those matches. After analyzing, we expound the performance measures that can directly affect the results. Owing to this information, the most of the clubs on the worldwide can easily benefit from the statistical way of competing for their rivals. With the specification of the performance parameters, football teams can adapt their training techniques and developing various strategies both in the game and transfer period in order to build a successful team. Motivation is an important milestone for every team in the league, which is directly related to success rates. If the team generally defeats the other teams during the entire league, their confidence and the motivation increase and also, supporters of the team enjoy when they watch their team. This situation brings together a high amount of revenue in terms of ticket sales, views, commercials, stores' income, youth sports schools, etc.

CHAPTER 2

LITERATURE REVIEW

2.1 Definitions of Machine Learning

The term machine learning refers to the machine-controlled detection of purposeful patterns in information. Within the past few decades, it has become a standard tool in virtually any task that needs data extraction from large information sets. In modern world, machine learning is used in search engines, anti-spam software systems and computer codes, digital cameras, smart phones, accident interference systems, scientific applications including bioinformatics, medicine, physical science, computer engineering areas to make life easier for humans. For instance, search engines learn process of bringing the most straightforward results after each query. Anti-spam software systems learn to filter email messages and thus, credit card transactions can be secured by a software system. In addition, computer codes can learn a way to find frauds. Digital cameras learn to find and analyze faces; therefore, smartphones can achieve voice recognition commands and recognize faces. Modern cars are equipped with accident interference systems that can secure drivers and passengers for possible accidents. As it is understood, machine learning is used in many areas and disciplines. Thanks to this situation many definitions have been made by various researchers. It is defined as the logical thinking of calculations and factual models that computer frameworks utilize in to perform a particular task successfully without using unequivocal enlightening, depending on designs and induction instep. Machine learning is seen as a subset of artificial intelligence [11].

Alpaydın [12] defines machine learning as programming computers to optimize an execution model utilizing case information or past encounters. We have a model with a few parameters, and learning is an execution of a computer program that optimize these parameters. The model may be prescient to create expectations within the future, or graphic to collect information from information, or both. Machine learning uses the hypothesis of insights in building numerical models, since the main task is making deduction from a sample.

Murphy [13] defines machine learning as an evolving domain of informatics whose algorithms are outlined to imitate human insights by learning from the encompassing environment. Machine learning techniques have been implemented effectively in different areas including spacecraft engineering, pattern recognition, computer vision, financing, entertainment and medical applications.

Mitchell [14] defines machine learning as searching for the answers, which are related to construction of computer systems that consequently learn with experience, and defining fundamental processes of learning patterns. Also, Mitchell mentioned that statistics, human and animal learning methodologies in psychology and neuroscience are the fields that are closely related to machine learning. In future, it is can be anticipated the harmony between the studies of human learning and machine learning. These fields have a close relationship for the fundamental scientific questions. Speech recognition, computer vision, bio-surveillance, robot control and accelerating empirical sciences are the real world application fields of machine learning.

Bishop [15] defines machine learning and pattern recognition as concerning with automatic disclosure of regularities in the information through the utilization of computer algorithms and also, taking the actions with using these regularities. Bishop also mentioned that machine learning algorithms construct a model based on training data to perform the decision making and prediction tasks.

Tang et al. [16] defines machine learning as the study of utilizing computers for imitating capabilities of human learning. Besides, machine learning is the research area that reveals new knowledge and abilities, distinguish existing information and permanently improve performance. Rote learning, inductive reasoning, analogical learning, and deductive learning are the basic learning strategies of the machine learning. Rote learning represents the memory that stores and retrieves the information without reasoning and calculation. Inductive reasoning represents the common knowledge of specific instances, extracting the common regulations of data and reasoning from specific to general. Analogical learning is comparing similar actions and trying to find the relations and possible solutions of these actions. Deductive learning represents the result of certain explications and clarified process of examples from general to specific.

2.2 History of Machine Learning

When we examine the history of machine-learning field, in early 1940's Warreen McCulloch and Walter Pitts wrote a scientific paper [17] about artificial neurons and this research is the first study of neural networks in the literature. Proposed model is inspired by human brain data transmission mechanism and electrical circuits. In 1950 Alan Turing published a manuscript [18] on computing machinery and intelligence. In this paper, Alan Turing develops the Turing Test to specify a case that a computer has real intelligence. In order to pass the test, a computer must be able to trick a human into believing it is human. In 1952, Arthur Samuel developed a computer program that plays checkers, and the program improves itself after each play with itself. In 1957, Frank Rosenblatt published a paper [19] on perceptrons which is a probabilistic model for information storage and organization in the brain. He defines the perceptron structure consisting of neurons and aiming to recognize patterns. Moreover, this scientific study is crucial for the invention of developing feedforward neural networks model and back propagation technique, which are the fundamental methods of the modern artificial neural network algorithms in early 70's. In 1967, Pelillo published a a paper [20], which includes the nearest neighbor rule.

In 1980, Fukushima [21] proposed Neocognitron which models artificial neural networks in a different perspective, being accepted as the ancestor of the convolutional neural networks model. In 1982, Hopfield [22] invented recurrent neural networks. Recurrent structures are separated from feedforward structures because they use their outputs as inputs in the next process. Consequently, recurrent networks have memory. In 1986 Watkins [23] published his PhD thesis including the development of q learning that improves the usability of reinforcement learning. Reinforcement learning is the machine learning that consists of agents in a dynamic learning environment, which uses punishments and reward mechanisms. In 1992, Tesauro [24] developed a computer backgammon program, called TD-Gammon, which uses supervised learning and the multilayer neural network algorithm. The program was designed to play expert level backgammon. In 1995, Ho [25] published a researched paper that described random forest algorithm. Also, Vapnik [26] published a researched paper in the same year that was described support vector machine algorithm. Both researches are new scientific discoveries and they contributed to the literature.

2.3 Types of Machine Learning

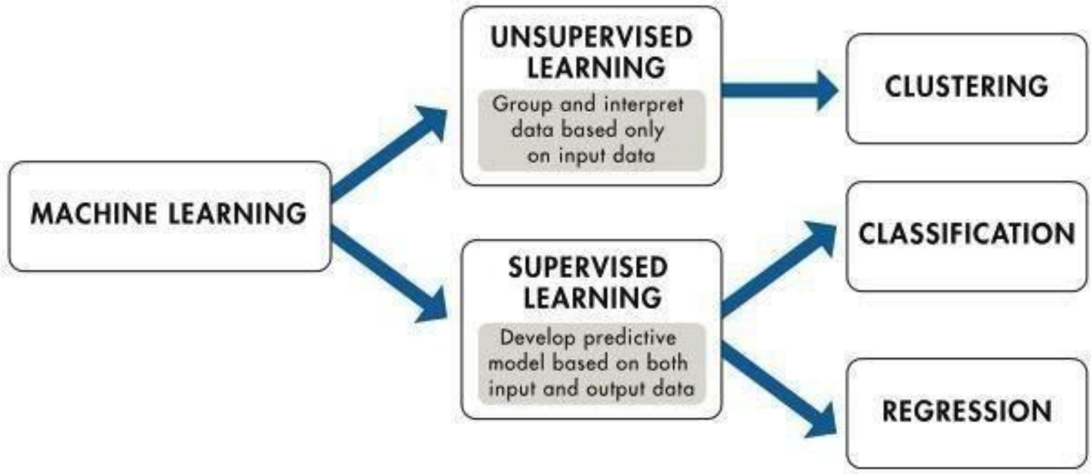

Figure 1: Machine Learning Structural Scheme

Figure 1 [27] shows the structural scheme of machine learning. As can be seen in the figure, two types of machine learning exist, which are called supervised and unsupervised learning. Classification and regression operations belong to the supervised learning, and the clustering operations are related to the unsupervised learning.

2.3.1 Supervised Learning

In supervised learning method, the input and output values are known. In this learning technique, the relation between inputs and expected outputs is found by generating a matching function. Bootkrajang [28] defines supervised learning as the determination process from labeled examples of the training set and forecasting the given labels and controlling the accuracy of the labels.

2.3.1.1 Classification

Classification is the strategy of categorizing the data into different classes. Classification forecasts the categorization of the data that it belongs to. Classification deals with text categorization, natural language processing, informatics, fraud detection, face recognition, marketing, optical character recognition. Artificial neural networks, support vector machine, nearest neighbor algorithm, naïve bayes and random forest algorithm are the popular classification algorithms. Classification can be applied to many application domains like computer vision, drug discovery and development, geostatics, speech recognition, handwritten recognition, biometric identification, biological classification, neural language processing, document classification, search engines, bank credit scoring and pattern recognition.

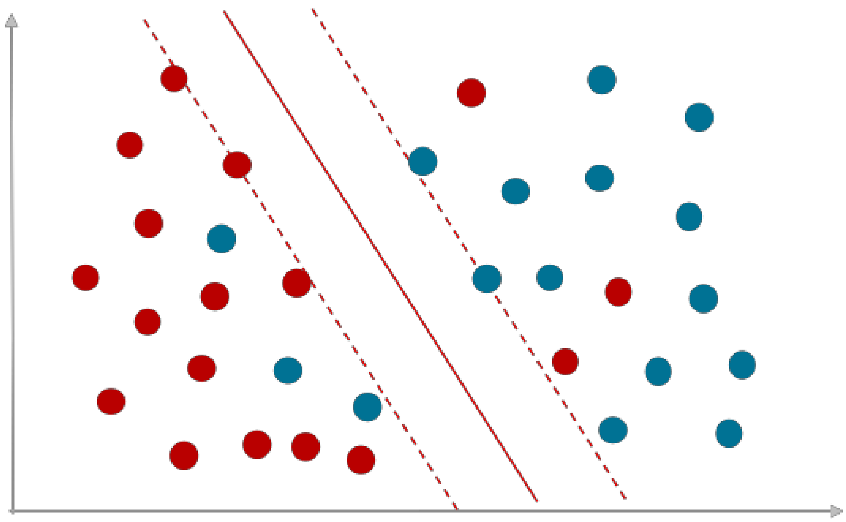

Figure 2: Classification Example

Figure 2 [58] represents the graphical notation of classification problem. As can illustrated in the figure 2, classification predicts the examples and classifies these examples in order to separate them from each other and to define the categorization of each class.

2.3.1.2 Regression

Regression is based on the linear relationship between two or more variables. Unlike the classification, predictions of the regression are based on numeric outputs in order to measure the relationship between two or more variables. If analysis is performed using a single variable, it is called univariate regression, and if more than one variable is used, it is called multivariate regression analysis. [29]

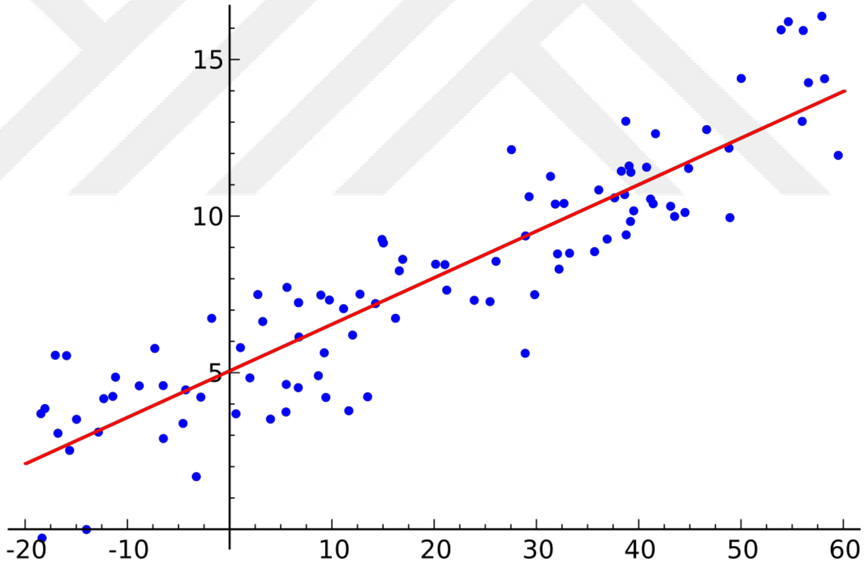

Figure 3: Linear regression example [46]

Equation 1: Yi = B0 + B1 X1 + E1

Equation 1 represents the linear regression model, where X is called as independent variable, Y is called as dependent variable, B0 represents population of Y intercept, B1 represents population slope coefficient and lastly E is called random error term. Population slope coefficient with independent variable and population y intercept part are called linear component and the random error term part represents random error component. [30]

Equation 2: $Y = B0 + B1\ X1 + B2\ X2 + \ldots + Bn\ Xn + E1$

Equation 2 represents the multiple linear regression model, where X's are called independent variables, Y is called dependent variable, E is called random error term. The aim of the regression analysis is to forecast the unknown parameters in the model and this process is also known as fitting the model to the data.

2.3.2 Unsupervised Learning

Unsupervised learning is a machine learning technique that uses a function to estimate an unknown structure of unmarked data. Unsupervised learning has the goal to learn the relationships and structures existing in the data without labeling the data as cause-effect or input-output. Nisson [31] defines unsupervised learning as utilizing the procedures that undertake to discover the relation between patterns. Bishop [15] defines the aims in such unsupervised learning problems that can detect the similar groups inside the data and this is called the clustering technique or it means settling the combination of the data within the input area known as density estimation. Alpaydın [12] defines the purpose of the unsupervised learning as finding consistencies in the input parameters. Structural input base space that certain patterns arise regularly than others needs to analyze what generally occurs more and what does not occur regularly. In statistics it is called density estimation. Clustering is the method regarding the density estimation which aims to define groups and clusters of the input base.

2.3.2.1 Clustering

Custering is determination of the similar subset of groups among the unlabeled data. Clustering is the unsupervised learning category based model, and thanks to the nature of unsupervised learning, input variables are unknown and not provided by a supervisor. Aldenderfer et al. [32] defines clustering analysis as the general name for the set of processes which can be utilized for creating categorization. Clustering is the process of grouping the variables with the experimental procedures in order to have similar variables togeather.

Alpaydın [12] defines clustering approach as discovering the mixture of the parameters from the given set of the data. Wikipedia's [33] definition is the operation of grouping objects in a way of separation with the same subset of objects in a same cluster. Clustering analysis is one of the primary objective of the data mining field, an ordinary technique for the statistical analysis and used in such areas including machine learning, pattern recognition, information retrieval, data compression, image analysis, computer graphics and bioinformatics. Cluster models are also important milestones of the clustering analysis. Connectivity models, distribution models, centroid models, density models, subspace models, group models, graph-based models, signed graph models and neural models are the typical types of the clustering models. Basically, connectivity models, which are also known as hierarchical clustering, mainly determine the more similar subsets and organize them into nearby and position these subsets away from unrelated subsets. Distribution model defines the same distribution of objects that belongs to with using Gaussian mixture models and distribution algorithms. Centroid models, also known as k means clustering, specifiy the clusters with using central vectors that can be defined as k value and it is not indispensably a value of the dataset. A density model considers the areas of the dataset and searches for the consistencies in defined sparse areas. Subspace clustering is an expansion of conventional clustering that looks for discovering clusters completely different subspaces inside a dataset. [34] Group models define the distribution of the objects as grouping information and grouping them together based on the provided information. Graph based models consider dividing the subset of nodes in a graph and defining relation forms of the cluster. A neural model is the self-organizing map that implements principal component analysis to the subspaces of the models.

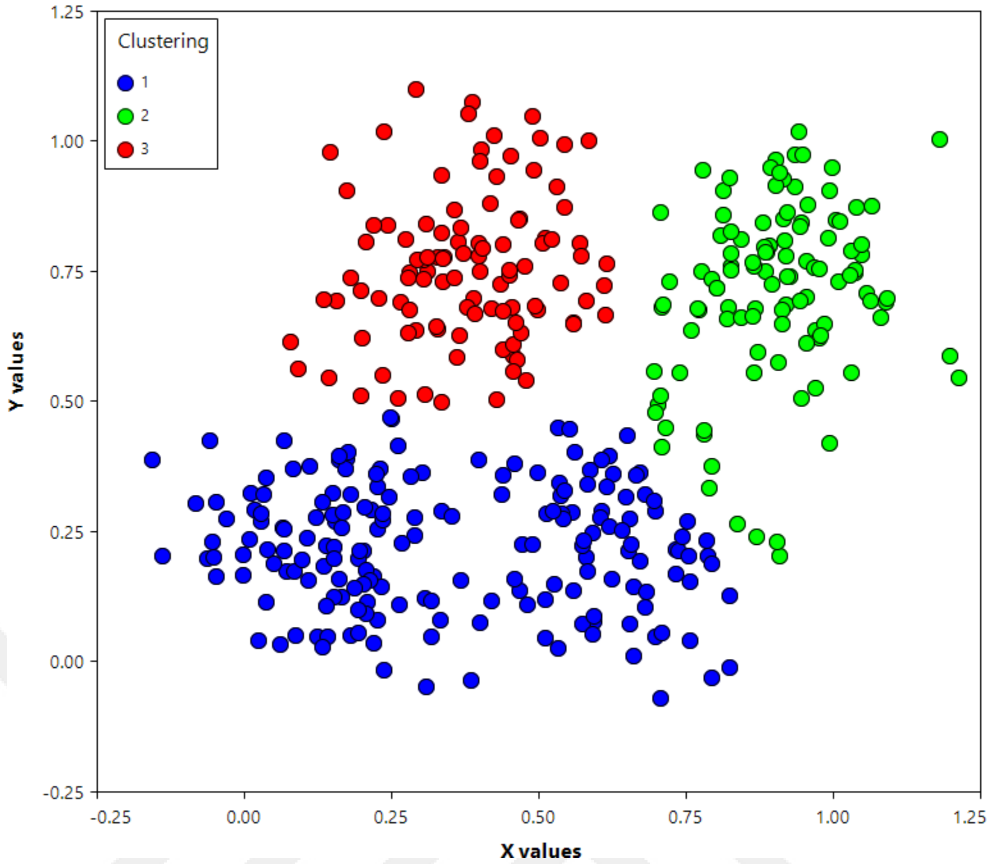

Figure 4: Clustering Example [57]

2.4 Support Vector Machine

Data Mining could be a pioneering and engaging analysis space thanks to its vast application areas and task primitives. Support Vector Machine (SVM) is enjoying a decisive role because it provides techniques that are particularly compatible to get ends up in an economical method and with a good level of quality**.** The use of support vector machine in various applications makes this tool inevitable for the development of products which have implications for the society. Support vector machines being computationally powerful tools for supervised learning, are widely employed in classification, clump, and regression problems**.** Support vector machines have a good performance when implementing various kind of problems like face recognition, text categorization, bioinformatics, computer-science related topics, civil engineering, and electric electronics, etc.

Support Vector Machine algorithm, firstly introduced by Vladimir Vapnik within the world of applied mathematics learning theory and structural risk reduction, have demonstrated to figure successfully on various prediction and classification issues [35]. Support vector machines can be used in several pattern recognition and regression issues about estimation and prediction[36]. Support vector machines can capture large feature space because of the generalization principle regarding the structural risk minimization theory [37].

In artificial neural network classifiers, input weights are automatically updated throughout the training phase then, summation of the errors between the network outputs and the predicted outputs are decreased. Unlike the support vector machine classifier, decision boundaries are specified from the training phase with separating margins of the boundaries regarding the dataset.

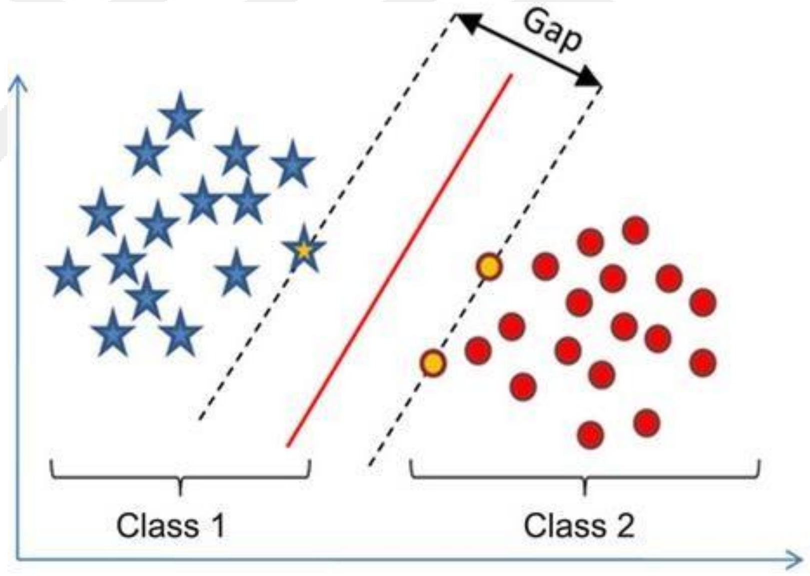

Figure 5: SVM Classification Hyperplane

Support vector machines can be categorized as supervised learning models which analyze the data and pattern recognition issues regarding the classification and regression analysis. Kernels, margin, duality, sparseness and convexity can be defined as the properties of the support vector machine algorithm [39]. Support vector machines can be used in text and hypertext categorization, classification of images, hand-written characters recognition, biological and other scientific areas.

Figure 5 [39] represents the linear decision surface which is called hyperplane, which aims to seperate class 1 and class 2 with the largest gap between the border lines of the both classes. These border lines can be also defined as support vectors and the vectors which are below from the decision limit called as negative hyperplane, the vectors that are above from the decision limit called as positive hyperplane.

2.5 K Nearest Neighbors Algorithm

Hart et al. [41] defines the k nearest neighborhood algorithm as a non-parametric decision method that can be utilized for classification and regression operations. In classification, k nearest neighbor algorithm categorizes and labels the test samples regarding the feature sets with the k value and finding the closest values to the k value in every class on the feature set. Euclidian distance, Manhattan distance and Minkowski distance are the popular distance vector functions when finding the closest values [42]. We can consider k nearest neighbor algorithm as a lazy learning approach that does not use any training value for generalization like in the eager learning approaches [43].

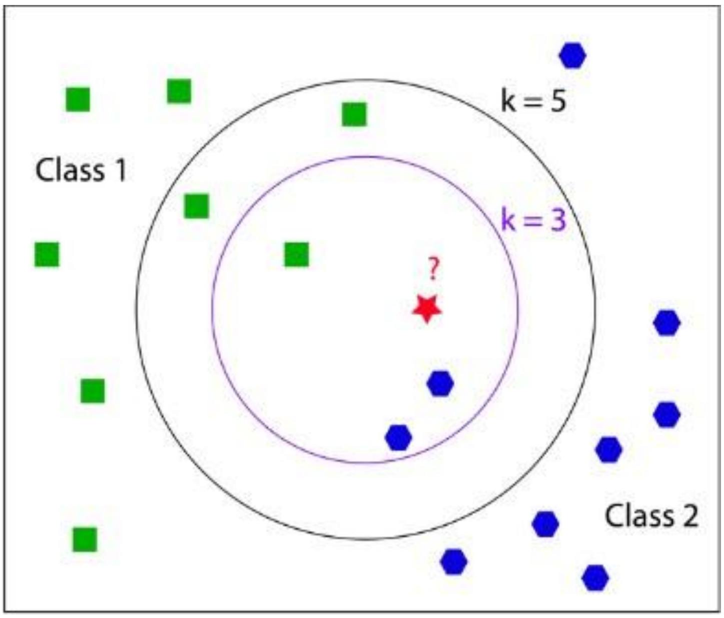

Figure 6: Nearest Neighbor Algorithm Usage Example

Figure 6 [44] represents the nearest neighbor algorithm usage with a classification example in terms of two classes, which are called class one and class two. As it is illustrated in the figure 6, green squares are represented with class one and blue hexagon representation belongs to the class two variable. Red star is represented for the value that we want to predict and it identifies that the value belongs to the which class in the possible k values. In this example, we can consider three variations of the k value, which are one, three and five. For k equals to the one case, algorithm calculates the distance of the variables on the feature space and as can be seen on the figure 6, blue hexagon value is more closer than the other variables to the red star and in this situation, the red star position will be classified as blue hexagon. For k equals to the three case, after the calculation of the distances, as can be seen on the figure 6, red star position will be classified as blue hexagon because it is closer to the two blue hexagon values instead of the one green square. Lastly, for k equals to the five case, as can be seen on the figure 6, red star position will be classified as green square because three green square values are greater than two blue hexagon values.

Distance functions

Euclidean

$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

Manhattan

$$\sum_{i=1}^{k}|x_i - y_i|$$

Minkowski

$$\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$$

Figure 7: Nearest Neighbor Distance Functions

Figure 7 [45] represents the Euclidian, Manhattan and Minkowski distance functions formulas. As can be seen on the figure 7, in both Euclidian, Manhattan and Minkowski formula, Xi and Yi represent the points on the coordinate axis and the main aim is to find the distance between Xi and Yi [47]. Han et al. [48] defines

Minkowski distance formula that can be considered as a generalization of the both Euclidian and Manhattan distances. In general usage, p value equals one or two. If p value equals to one, the formula is similar with the Manhattan distance formula. If p value equals to two, the formula is simiar with the Euclidian distance formula [49]. In this thesis, Euclidean formula is preferred as the distance function of the k nearest neighbor algorithm because of the highest performance results and the agile computational harmony between the function and the classifier.

2.6 Artificial Neural Networks

Artificial Neural Networks are computing systems mistily modeled by the biological nature of the human brain activity. Such systems learn to perform tasks by considering examples, while not being programmed with any task-specific rules typically. Artificial neural networks consist of the various interconnected artificial neurons like the neurons in the human brain. [50] The connection between the neurons represents the synapses and can transmit and receives signals one to another. In the general approach, the connection between nodes are weighted numbers, and the output of each neuron is calculated with the summation of the neurons in the ANN structure calculated by the activation functions. [51] The main purpose of implementing artificial neural network structure is resolving various types of problems.

2.6.1 Artificial Neuron

An Artificial neuron is the basic unit of artificial neural networks. Each node has a connection point which is capable of producing weighted inputs and transfer the output to other edges in the network. Same as biological neurons, when an input is received, the neuron determines to send the output passed through the next layer as an input. [52] The determination of sending output signal is representing the term 'bias'. Generally, bias is directly related to the activation function that the system uses. Figure 8 [52] represents the equation of the neuron with bias. Y represents the output of the model. W_j represents weight connections; X_j represents input variables.

W_0 is the intercept value to make a model more general; it is generally modeled as the weight coming from an extra bias unit, W_0, which is always +1.

$$y = \sum_{j=1}^{d} w_j x_j + w_0$$

Figure 8: Biological Neuron model with bias

2.6.2 Perceptron

The simplest associate degreed oldest model of artificial neural networks, the perception may be a linear classifier used for binary predictions. This implies that the information should be linearly dissociable. [53] Basic operation logic regarding perceptron structure is achieving total summation values regarding multiplication of both input weight variables and connection weight variables. [31] Next step is adding threshold values to the total summation values and applying an activation function to achieve output values.

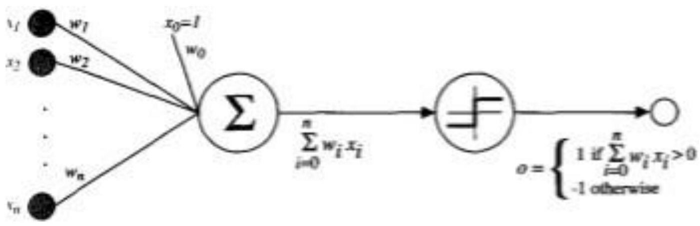

Figure 9: The perceptron structure [52]

2.6.3 Multilayer Artificial Neural Network

Multilayer artificial neural network or multilayer perceptron is an organized layer system which includes sequences of connected neurons. Structurally, artificial neural networks consist of three parts, which are input layer, hidden layer, and output layer. [53] Input layer brings the initial information into the system for any process by later layers of artificial neurons. The hidden layer is the operational layer regarding training and learning the values. The hidden layer is between the input layer and the output layer that transfer the weighted inputs from the input layer neurons to the output layer neurons. The output layer is the last layer of neurons that generate given outputs for the network. More refined than the perception, a Multi-layer ANN, like Convolutional Neural Network, is capable of resolving a lot of advanced classification and regression tasks because of its hidden layer(s) [50].

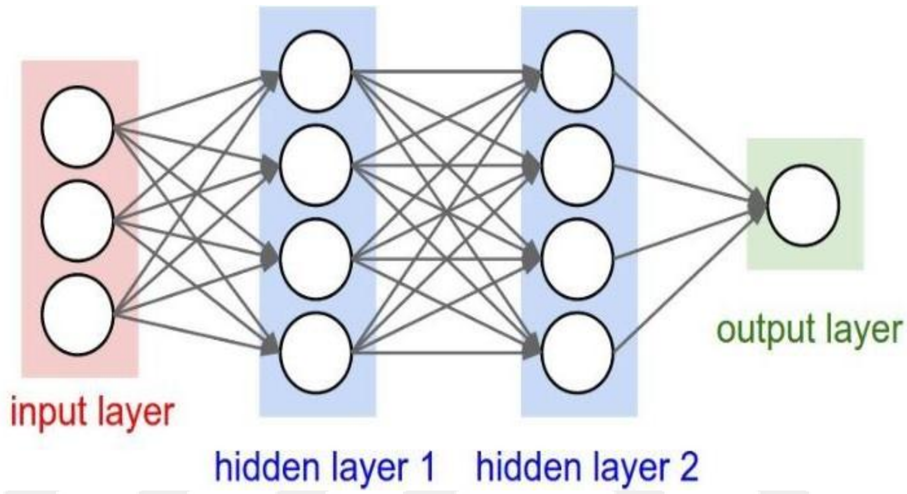

Figure 10: Multilayer ANN Architecture [55]

2.6.4 Activation Functions

Activation functions are the functions that produce output regarding the weighted summation of the inputs which belongs to the artificial neurons [53]. Basically, it computes the results of all multiplying input variables by the weighted connections with adding bias values; then, implementing non-linear function through that results and converts that values between 0 and 1 [52]. The main reason of using the non-linear function is to obtain differentiable results which can be able to derive each of them. This situation has an important role when performing backpropagation optimization in feedforward neural networks that can easily find non-linear error gradient to learn complex behavior of the problem [50]. Activation functions roughly model the way that neurons communicate in the brain with each other. Each node is activated and send signals if the node reaches a certain threshold value. In each layer feed forward with the same terminology, application function is applied and the output is passed through the next layer continuously until reaching the last layer in order to determine the prediction. In order to solve non-linear problems, there are several activation functions that can be used. Most popular activation functions are sigmoid, hyperbolic tangent function, and rectified linear unit activation function. In this study, we used a sigmoid activation function when we trained our model with multilayer perceptron algorithm in the Weka environment.

2.6.5 Sigmoid

A sigmoid function may be a function having a characteristic S-shaped curve or sigmoid curve [54]. Often, sigmoid perform refers to the special case of the provision function that generates a collection of likelihood outputs between zero and one once fed with a set of inputs. The sigmoid activation perform is widely utilized in binary classification.

Equation:

$$sigmoid(x) = \frac{e^x}{1 + e^x}$$

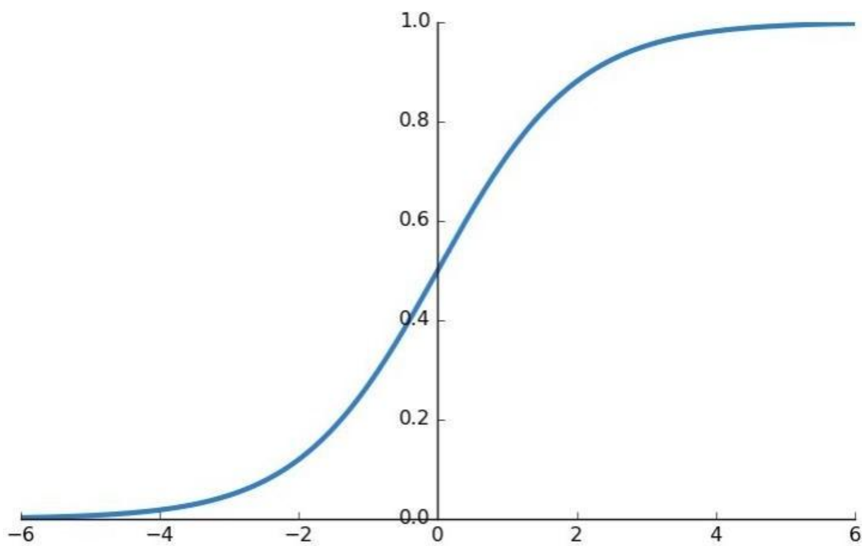

Figure 11: Normal Sigmoid graphical representation [56]

2.6.6 Backpropagation

Backpropagation is an optimization technique that is used to train nodes in artificial neural network architecture in order to reach desired outputs of the supervised learning. It helps the neural network to the correction of the prediction results and finds the absolute best weight values of each node. The purpose of the backpropagation is to find partial derivatives of the error function with respect to each individual weight in the network [53]. Partial derivatives are important for updating weights between nodes with gradient descent function. Figure 5 represents the scenario of reaching the global loss minimum value which is an optimal value for each weight in our network, starting some random weight value then step in the direction towards the minimum error which is the opposite of the gradient. With descending down the gradient eventually, the weight will find the minimum of the error, and that process is called gradient descent. Gradient descent function computes the error for every layer of the network and then, relating those error values to the quantity of real interest with partial derivative with respect to any weight in the network [52]. Backpropagation simply consists of repeatedly applying the chain rule in calculus through the all possible paths in the network. After finding the error values for the output layer, it computes error values of the neurons which belongs to hidden layers to the backwards [54]. Afterwards, the new weighted connection values are assigned between output and hidden layer connections then, it calculates the error values of the neurons which belong to the input layer and assign all new weighted connection values between the hidden layer and input layer.

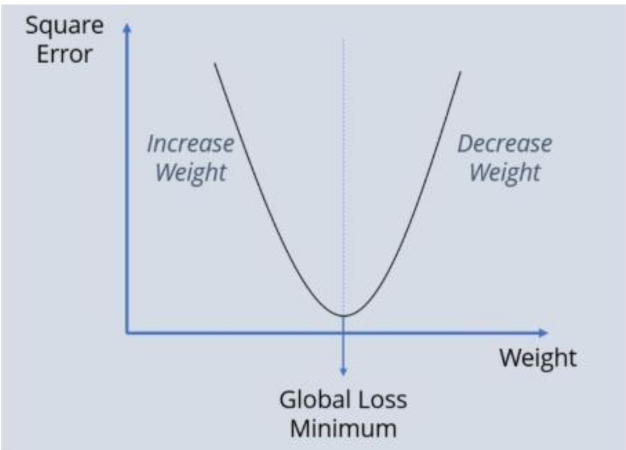

Figure 12: Gradient descent process [52]

2.7 Feature Selection

In machine learning, feature selection is the operation of defining and utilizing related features in order to use in model construction [59]. James [60] pointed out the four reasons that feature selection approach are applied for. These are the facilitation of the models, shorter training times, refrain the curse of dimensionality and improved generalization by reducing overfitting. One of the purposes of the feature selection is to avoid redundant data in the dataset and separate strongly correlated values. Guyon et al. [61] analyze the feature selection algorithms with three categorizations, which are known as filter methods, wrapper methods, and embedded methods.

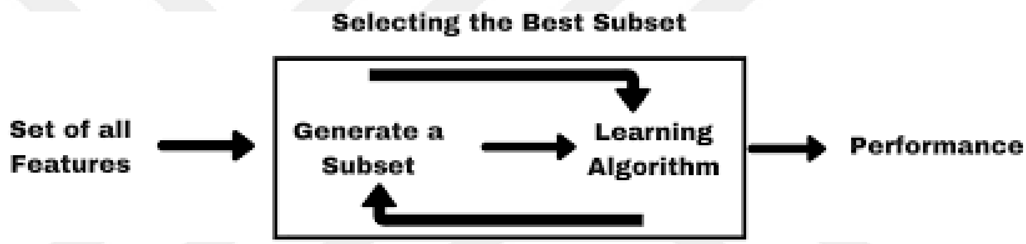

Figure 13: Wrapper Feature Selection Method Structure

Figure 13 [62] represents the structure of the wrapper feature selection method. Based on the representation, wrapper methods generate a subset and pass through this information to the learning algorithm. Learning algorithm trains with the selected subsets and supervisors inferences from the output and determines instead of adding or removing features from the subset. Forward selection, backward elimination, and recursive feature elimination are frequent examples of the wrapper methods [62].

Forward selection method basically defines starting with the zero feature and each iteration applying new feature and performing the system till the recently added variable cannot improve the performance of the system. Backward elimination method starts with the all attributes and discards the less significant attribute in each iteration, till the discrimination of the features cannot affect the performance of the system. Recursive feature elimination is a type of greedy optimization algorithm. Basically, model reiterates itself and defines the correlation status of the performing feature in each loop. At the end of the iteration, features are ranked thanks to the ordering of their discarding [62].

Figure 14: Filter Method of the Feature Selection Structure

Figure 14 [62] represents the structure of the filter feature selection method. Filtering methods are commonly utilized in the preprocessing step. According to Yang et al. [63] filter model deals with contemplates with the relation between the attributes and the class labels. Evaluation measures are crucial importance for the filtering model.

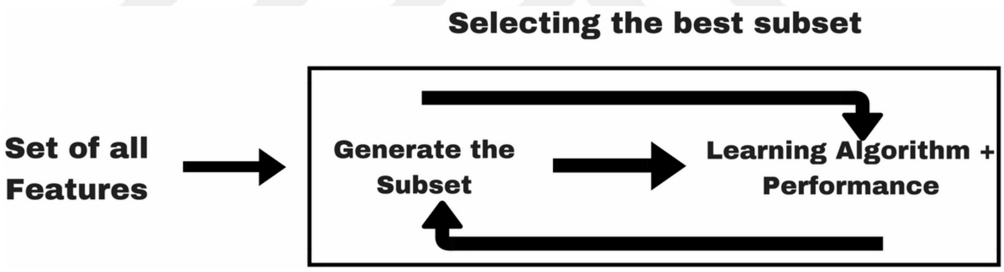

Figure 15: Embedded Method of the Feature Selection Structure

Figure 15 [65] represents the structure of an embedded feature selection method. The embedded model chooses attributes in the training process of the learning algorithm performance and the results of the feature selection are automatically computed during the process of training [63].

Subset selection can be defined as the technique that evaluates a subset of the attributes. Subset selection can be used with wrapper, filter, and embedded methods. Exhaustive, best first, genetic algorithm, greedy forward selection, greedy backward elimination, scatter search, target projection pursuit, and particle swarm optimization are the approaches regarding the investigation through the subsets [57].

Exhaustive search, also known as brute force search is a common approach, which enumerates the features and identifies the fulfillment of each attribute contribution. Exhaustive search algorithm aims to enumerate all the attributes from one to n and tests each combination of the features and observes the solutions of the combinations, whether it is successful or not. Implementing a brute force search is easy and should prefer using the size of the problem that is limited [64].

The best first search is the search algorithm that investigates the nodes enlarged by a graph according to an exact rule. Hall et al. [65] define best first search algorithm, which begins with the zero features and produces all possible expansions, electing the best of evaluation variable subsets and expanding with the attributes one by one. If the subset of the results cannot positively affect the performance, search algorithm rollbacks the previous best combination of the subset, and proceed with the process. After the best combination is found, search algorithm terminates the process.

2.8 Literature Study

Hucaljuk et al. [1] developed a software system, which predicts the outcome of champions league football matches. The authors reduced more than 30 attributes to 20 attributes in attribute selection part of the research. When they reduced attributes, features are selected by the author's knowledge based on the problem that they believe mostly affect the final result. Naïve Bayes, Bayesian networks, logit boost, k nearest neighbors, random forest and artificial neural network learning algorithms are used to achieve prediction results and determine which performance measures of the algorithms give the best results. Weka API is used for the implementation and dataset contains 96 matches that belong to the champions league football matches. K-nearest neighbors classifier reaches %62 accuracy in maximum with five nearest neighbors, and artificial neural networks reach %68 prediction accuracy in maximum in this research with a back-propagation technique and five hidden layers for training the neural network model.

Ulmer et al. [2] predicts the results of English premier league football matches. The dataset contains ten seasons information between 2002 to 2012 of English premier league. Also testing set includes two seasons, which are 2012-2013 and the 2013-2014 seasons. There are 3800 games used in the training set, and 760 games are used in the testing set. The dataset includes the home team, away team, the score, the winner, and the number of goals for each team. Python environment is used for the preprocessing and the implementation. Baseline, naïve Bayes, hidden Markov model, support vector machines, random forest, and one vs. all stochastic gradient descent models were applied on the given dataset. Results of the best performing models were support vector machine with radial and linear kernel, random forest model, and one vs. all stochastic gradient descent model, achieving error rate between 0.48 and 0.52. One vs. all-stochastic gradient descent model can predict three draws, 249 wins, and 389 losses. Support vector machine with linear kernel predicts 307 wins, 317 losses, and no draws. Random forest model can predict 12 draws, 302 wins, and 312 losses. Support vector machine with radial kernel predicted 42 draws, 275 wins, and 293 losses. At the end of the study, authors predicted the standings of 2012-2013 English premier league season and predicted Manchester United football club championship correctly.

Purucker [3] used an artificial neural network algorithm and tried to predict the National Football League (NFL) game winners with both supervised and unsupervised training strategies. Adaptive resonance theory, hamming, Kohonen self-organizing map, and backpropagation techniques are applied in the research. As a supervised learning algorithm, the multilayer perceptron architecture consists of five input neurons, two hidden neurons and lastly one output neuron. Learning rate was determined as 0.5 and momentum value was determined as 0.4. The hyperbolic tangent is the activation function used in the multilayer perceptron environment. As unsupervised learning: adaptive resonance theory, Kohonen self-organizing map networks, hamming network, and maxnet subnetwork are applied on the dataset. Unsupervised training results showed that hamming network achieves %50 success rate classification dividing into groups of 10 into 18 teams, the adaptive resonance theory achieves %50 success rate with week 15 prediction, and lastly, Kohonen self-organizing map network achieves %57.1 accuracy prediction rate. Supervised training results shows that the multilayer perceptron algorithm achieved %78.6 prediction accuracy when compared to the %64.3 success rate of football experts.

Baboota et al. [4] developed a generalized predictive model for the football game results of the English premier league. Feature engineering and feature selection techniques were applied in order to find the most relevant attributes. In feature selection, features were divided into two classes after that feature set was tested with Gaussian naïve Bayes model. Class B attributes were better to fit the predictive models. Gaussian naïve Bayes, support vector machine, random forest, and gradient boosting are the algorithms used for the prediction. Dataset was divided into nine seasons between 2005 to 2014 for the training data, two seasons from 2014 to 2016 were considered as test data. Python environment was used for implementation. Results of the study suggest that support vector machine with linear kernel predicts 254 wins, 93 losses, and no draws. Support vector machine with linear kernel predicts 238 wins, 96 losses, and 15 actual draws. Random forest algorithm predicts 225 wins, 101 losses, and 35 draws. Lastly, gradient boosting result predicts 222 wins, 99 losses, and 42 draws.

Joseph et al. [5] predicted the outcome of the results regarding Tottenham Hotspur football club. MC4 decision trees, naïve Bayesian learner, data-driven Bayesian learner, expert constructed Bayesian network, and k nearest neighbor algorithms were the machine learning techniques used in this study. The dataset contains two seasons of data, which are 1995, and 1996 seasons that contain 76 instances with 30 attributes. In conclusion, MC4 learner overall classification error is considered as %69.81 for disjoint training and testing data sets in general model and, %61.35 for the expert chosen data. In complete seasons, MC4 algorithm achieves %23.68 classification error rate for the general and expert models. Secondly, Naïve Bayes learner achieved % 64.26 classification error rate for a single season and cross seasons overall. K-nearest neighbor algorithm achieved %68.52 accuracy reached in a single season data, and %65.44 accuracy was reached in cross seasons. K nearest neighbor algorithm yielded the best performance than the other machine learning techniques of the study.

Ganesan et al. [6] proposed a model to predict the match outcomes of English premier league football matches. Dimensionally reduction was used in the pre-processing phase of the study. The authors reduced 65 attributes to most relevant 15 attributes in the dataset. Logistic regression, support vector machine, and xgboost classifiers are used in the proposed model. Full-time-result variable is obtained as a target variable of the model in prediction.

Igiri et al. [7] developed a prediction system for football match results. The authors developed a result predictive model by gathering nine in-game features that affect the football match outcomes directly. Rapid miner data mining tool is used as an environment for preprocessing and implementation. In the analysis and design part of the study, authors analyze two approaches, which are statistical, and machine learning approaches. The dataset includes the number of goals scored in home and away matches, the number of goals conceded home and away matches, attack, and defense strength, and players injuries. In learning model, artificial neural networks, logit boost, Bayesian network, support vector machine, and logistic regression are used to determine prediction performance. The result of the study showed that artificial neural network classifier reached %85 accuracy after the optimization. Logistic regression algorithm achieves %93 accuracy rate. However, logistic regression can only predict a win or loss result of the fixture and cannot predict draw matches. The authors emphasize that the artificial neural network technique is a better option if draw matches need to be predicted. Recommendations are organizing and collecting necessary football match fixture and feature information in excel files. Understanding the in-game features that directly affect the game results and further researches can guide young researchers to a proper sense of direction.

Huang et al. [8] developed a prediction model based on multilayer perceptron with backpropagation technique. The proposed model was adopted to predict the 2006 world cup football game. In feature selection, there are 64 matches reports with regarding 17 attributes in each match report. Authors selected eight attributes instead of 17 attributes which are goals for, shots, shots on goal, corner kicks, direct free kicks to goal, indirect free kicks to goal, ball possession and fouls suffered. Data normalization technique is applied in a dataset with selected attributes. Multilayer perceptron structure includes eight inputs, eleven hidden nodes, and one output node. Sigmoid transfer function is used in backpropagation technique. Momentum rate is considered 0.6 and learning rate is considered 0.9. Authors consider the testing part of the study in five stages. The result of the study showed that in stage 2, %85.7 prediction accuracy reached for seven games of the tournament. In stage 3, %66.7 prediction accuracy reached for the three games of the tournament. In stage 4, %50 prediction accuracy reached for two games and lastly in stage 5, %100 prediction accuracy reached for the final match of the tournament. Authors remarked that if tied games participate in, an average of thirteen matches prediction accuracy reaches %76.9.

Bunker et al. [9] developed a machine-learning framework for predicting sports results. Authors named this model by SRP-CRISP-DM type framework and define six steps based on traditional SRP-CRISP-DM model. These steps are business understanding, data understanding, data preparation, modeling, evaluation, and deployment. Basically domain understanding defines characteristics of the sport itself, data understanding implements the source and level of the data, data preparation includes feature extraction methods and preprocessing data by in-game variables, modeling defines models used for prediction and feature sets, model evaluation outputs the results of performance measures, deciding splitting techniques to implement and lastly, deploy model section retrain model and generate predictions for upcoming matches. In preprocessing original data or subset dividing into two parts, these are match related features and external features. After dividing, the average process the past matches information begins to operate. In training and testing section, held-out training and testing split is used. The paper defines another option which is to update dataset every match after being played, and this technique can be an example of leave one out cross-validation technique. In modeling part of the study, the authors generally focuse on artificial neural network classifier and also give a chance to the users to implement different machine learning techniques. In conclusion, the authors pointed out that generally, prediction of the mathematical and statistical models regarding match results verified by experts and machine learning strategies could be an appropriate model for sport prediction. Therefore, SRP-CRISP-DM framework is the proposed model to predict different sport field results.

Cao [10] built a model, which predicts the outcome of the national basketball association (NBA) league basketball matches using machine-learning algorithms. Dataset consists of five national basketball association (NBA) league and one season used for evaluating the dataset. Simple logistics, artificial neural network, support vector machine, and naïve Bayes classifiers are implemented to determine the outcomes of the basketball match results. Feature extraction technique is applied for the dataset. Statistics of the team, rival statistics, schedule, player statistics and recently played matches performance statistics were combined as features. The number of features reached thirty to sixty attributes. In the model evaluation part, the author used n fold cross-validation technique, n values are considered as five, ten, and twenty. After this process, performance metrics averaged the overall performance metrics. Weka environment is used for the implementation. Experiment evaluation formed including training and testing data, class assigner, randomizer, cross-validation fold maker, classifier, classifier performance evaluator, and model performance chart. Simple logistics function used five hundred maximum iterations, eight hundred five for heuristic and 0.02 for weight trimming of logit boost. Naïve Bayes classifier is used with default parameters of weka tool, support vector machine classifier gamma parameter is considered as 0.002 and coefficient determines 0.5. Artificial neural networks classifier is used with feedforward model with using sigmoid activation function. One hidden layer and two output nodes model implement with 0.1 learning rate, 0.05 momentum rate, 700 number of epochs, and 231 random seed initialization. Multilayer perceptron function in weka framework is used to implement artificial neural network classifier. In conclusion, simple logistics achieved %69.67 prediction accuracy, naïve Bayes achieved %66.25 prediction accuracy, support vector machine achieved %67.70 prediction accuracy, and artificial neural networks achieved %68.01 percent accuracy.

Tüfekci [66] predicts the results of the Turkish super league football matches using machine learning algorithms. Dataset is combination of the four seasons period statistical data between 2009 and 2013, which contains 70 attributes and 1222 instances. In feature selection, genetic search algorithm is selected for the evaluation parameter and a filter-based feature selection method, cfs subset evaluator parameter of the weka framework, three wrapper based feature selection methods and wrapper subset evaluators used to find the most relevant attributes. In learning model, support vector machine, bagging rep three and random forest algorithms are used to determine prediction performance. Weka framework is used for the implementation and evaluation of the results. Results showed that random forest achieve %70.61 prediction accuracy with using wrapper subset evaluation, which reduced the attributes 70 to 38. Support vector machine classifier achieves %67.94 prediction accuracy, and lastly, bagging rep three achieve %67.86 prediction accuracy.

Cengiz et al. [67] predicts the results of Turkish super league football matches with using machine learning algorithms. Dataset is the 2008-2009 season of the Turkish super league football game statistics. The authors used Bayesian approach with a poisson log linear approach and artificial neural network approach as a learning model. In order to make a comparison of prediction accuracy, poisson loglinear and two artificial neural network algorithms were considered. Feedforward neural network technique is applied for ann1 algorithm and radial basis function technique is applied for the ann2 classifier. Results of the study showed that two artificial neural network model give the closer performance results and the neural network models give better performance than Bayesian approach.

Karabıyık et al [68] proposed a model, which uses bayes networks in football prediction. Football Bayes Network (FutBa), which is the name of the proposed model and the aim of the model, is to predict Turkish super league football matches. According to the authors, FutBa uses the detailed statistical data collected from past matches with utilizing expert opinions, making a prediction of the outcomes. FutBa can predict the results with both teams, offensive and defensive forces, according to many variables and the prediction is based on factors such as team form and league status. In this model in which twenty two attributes specified with the experts, attributes are divided into general: defence and attacking. Evaluation result of the study showed that FutBa achieve %70 prediction accuracy. Also, when predicted using historical data, he predicted 61% of the matches to be accurate and 28% as close.

Karaoğlu [69] uses a formulization based on the teams scored and conceded goal averages since the beginning of the season and enquiring how machine learning algorithms can successfully predict the results. The dataset contains eleven european league and the twenty football league data, which are the statistics of the seasons between 1993 to 2015. Naïve bayes, bayes net, multilayer perceptron, logitboost, decision table, zeror and c4.5 algorithms are the learning models that are used to determine the prediction performance of the study. Weka framework is used for the implementation and evaluation of the results. Results showed that there was a maximum difference of 4% between the algorithms showing the best results and the worst results. Karaoğlu pointed out that in this case, where the highest result is 50-52%, an assessment can be made as existing classification algorithms are not very suitable for the football dataset.

Esme et al [70] proposed a machine learning model that can predict football matches. The dataset contains the statistical data of the Turkish super league seasons between 2010 and 2016. K nearest neighbor algorithm is the learning model in this study in order to determine prediction performance. According to the results, %55.56 prediction accuracy reached in a k value of fourteen.

Zaveri et al. [72] proposes a solution regarding the prediction of football match results with using machine learning algorithms. Dataset is the combination of the five seasons period statistical data between 2012 and 2017 Spanish La Liga football matches. Fifa 18 game database is used with 12 attributes and 29 teams which participate in the league between 2012 and 2017. Python sci-kit learn environment is used for the implementation and ten-fold cross validation technique is used for evaluation. In the learning model, logistic regression, random forest, artificial neural network, naive bayes and support vector machine are used to determine prediction performance. Results show that logistic regression achieves highest prediction accuracy with %71.63, random forest achieves %69.9, artificial neural networks achieve %69.2, support vector machine achieves %66.95 and lastly naive bayes achieves %63.57 prediction accuracy.

Pallingi et al. [73] predicts the outcomes of Spanish La Liga and Segunda Division football matches in order to the find best machine learning model. Dataset is the combination of statistical data between 2013-2017 seasons of the La Liga and Segunda Division which contains 3830 football matches. Meteorological data is also gathered from Agencia Estatal de Meteorologia that contains wheather data of the every football match. Python environment is used for the implementation, normalization technique is used in the preprocessing phase and random forest, k-nearest neighbor and support vector machine classifiers are used to determine the prediction performance. In evaluation part of the study, authors split the data into two experiments. In the first experiment, weather attributes are used to predict full time result attribute of the dataset, whereas in second experiment both weather and historical statistical data of the each team are used to forecast full time result attribute. Also 5-fold cross validation is applied for the evaluation. Results show that k-nearest neighbor reaches maximum %47.11 accuracy in experiment 1 and support vector machine classifier reaches %49.51 accuracy maximum in experiment 2. Also authors mention that weather conditions can be used as effective variables to forecast the outcomes of the football matches.

CHAPTER 3

METHOD

3.1 Dataset

In this study, we have a dataset [71] that contains 1520 instances and 23 attributes which consist of 4 La Liga seasons information. In La Liga, there are 20 teams which play 380 football match in one season for each year. The data include statistical information of 2014-2018 seasons. Attributes are called Date (Match Date), Home Team, Away Team, FTHG (Full Time Home Team Goals), FTAG (Full Time Away Team Goals), FTR (Full Time Result), HTHG (Half Time Home Team Goals), HTAG (Half Time Away Team Goals), HTR (Half Time Result), HS (Home Team Shots), AS (Away Team Shots), HST (Home Team Shots on Target), AST (Away Team Shots on Target), HF (Home Team Fouls Committed), AF (Away Team Fouls Committed), HC (Home Team Corners), AC (Away Team Corners), HY (Home Team Yellow Cards), AY (Away Team Yellow Cards), HR (Home Team Red Cards), AR (Away Team Red Cards), LMH (Last Match Home Team Result) and LMA (Last Match Away Team Result). Half-time result and full-time result attributes are coded as ‘H' if the home team wins, ‘D' for the draw and lastly, ‘A' if the away team wins. Last match home team results and last match away team results are coded as ‘W' for the win, ‘D' for the draw, ‘L' for lose and ‘N' for not played. Date, home team, away team, half time result, full-time result, last match home team results, and last match away team results attributes are nominal attributes. The rest of the attributes are numerical attributes, and all the values are positive. In the originaldataset, there is not any attribute of last match home team results or last match away team results. We calculate and add these attributes manually in order to enhance accuracy rates.

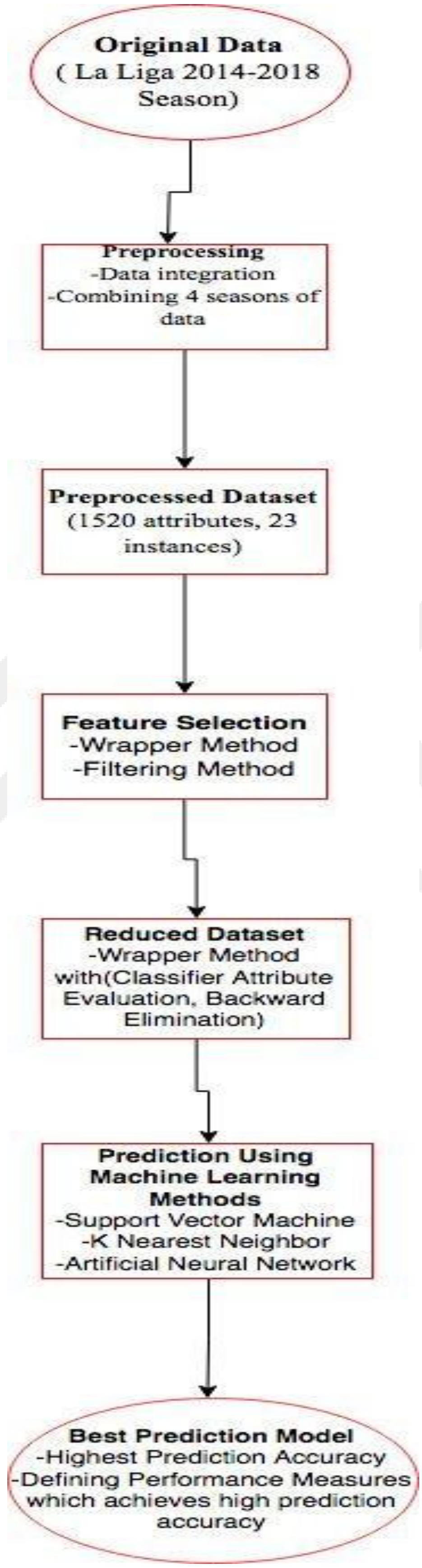

Figure 16: Flow Diagram of the Prediction Process

3.2 Attribute Selection

In the preprocessing phase, we used attribute selection feature in Weka Explorer in order to specify relevant and highly useful performance measures which are located under the filters, supervised and attribute folder. In this thesis, we performed more than one attribute selection option to achieve the best performance which are CfsSubsetEval (CSE) evaluation technique, Classifier Attribute Eval (CAE) evaluation technique and backward elimination technique.

CfsSubsetEval (CSE) evaluation option was used which evaluated the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them. In CSE, we performed the best first search and the exhaustive search algorithm to find the most relative attributes that can fit classifiers. After this process, five attributes were chosen in both best first and exhaustive search algorithm which are away team, full-time home team goals (FTHG), full time away team goals (FTAG), half time result (HTR) and full-time result (FTR).

Classifier Attribute Eval (CAE) evaluation option was used, which evaluated the worth of an attribute by using a user-specified classifier. In classifier attribute evaluation option, support vector machine, artificial neural network, and k-nearest neighbor classifiers were used with accuracy and root mean squared error evaluation measures. Random number seeds were applied with a 0.01 threshold value, which is the default value on classifier attribute evaluator selection method. Ranker search algorithm was the search method used in the classifier attribute evaluation method. After this process, 22 attributes were ranked which were full time away team goals (FTAG), full time home team goals (FTHG), half time result (HTR), half time home team goals (HTHG), half time away team goals (HTAG), home team shots on target (HST), away team shots on target (AST), away team, home team yellow cards (HY), away team shots (AS), home team red cards (HR), date, away team corners (AC), away team red cards (AR), last match home team result (LMH), last match home team result (LMA), home team, home team shots (HS), home team corners (HC), away team yellow cards (AY), away team fouls committed (AF) and home team fouls committed (HF).

Lastly, the backward elimination technique was used in order to achieve highest performance measures and to avoid overfitting issues. Attributes were reduced one by one in each experiment, and the results were tested with support vector machine, k-nearest neighbor, and artificial neural network classifiers. After this process, 15 attributes were chosen which were date, home team, away team, full time home team goals (FTHG), full time away team goals (FTAG), full time result (FTR), half time home team goals (HTHG), half time away team goals (HTAG), half time result (HTR), home team shots (HS), away team shots (AS), home team shots on target (HST), away team shots on target (AST), last match home team results (LMH) and last match away team results (LMA).

3.3 Support Vector Machine Classifier

Support Vector Machine (SVM) classifier is implemented for the dataset in the way of all possible feature selection results to determine full time result prediction accuracy of the La Liga season football matches. Library of support vector machine classifier is used with the type of C-support vector classification (C-SVC) regarding the batch size of 100 and the cache size of 40. Also, linear, polynomial, radial and sigmoid kernel types are used to find the best fit for the problem on the dataset. The tolerance of termination criterion epsilon is determined 0.001 with a 1.0 cost rate, which is the default parameters of the SVM function in weka explorer software. Also, 10-fold cross-validation technique is applied for the given selected attributes. When each classification output is determined, random number generator for input values is used ten times in order to calculate mean and variance value for the reliable outputs.

3.4 K-NN Classifier

K- Nearest Neighbors (KNN) algorithm is applied for the dataset in the way of all possible feature selection results to determine the classification accuracy of the class attribute full-time result. The attributes of the dataset are chosen with various techniques regarding attribute selection part, k- nearest neighbors (KNN) implemented for those selected attributes in 3 separate section. The linear brute force search algorithm is used with the Euclidean distance function for the nearest neighbor search. Classification regarding the batch size of 100 and k value starts with 3 and increment double after each classification task until it reaches 11. 10-fold cross-validation technique is applied for the given selected attributes. When each k value classification output is determined, random number generator for input values is used ten times in order to calculate mean and variance value for the reliable outputs.

3.5 Artificial Neural Network Classifier

Artificial neural network (ANN) classifier is implemented for the dataset in the way of all possible feature selection results to determine classification accuracy of the class attribute full-time result. Artificial neural network (ANN) classifier is implemented for those selected attributes with five input nodes in the input layer, five, ten and fifteen nodes are separately tested for each iteration in the hidden layer, and three nodes in output layer model is applied with a batch size of 100 and training time of 500. Momentum value and learning rate value is applied respectively 0.1 to 0.5 in each experiment to achieve highest classification results. Backpropagation technique is applied with sigmoid activation function, which is the default feature on multilayer perceptron classifier in weka framework. 10-fold cross-validation technique is applied for the given selected attributes. When each classification output is determined, random number generator for input values is used ten times in order to calculate mean and variance value for the reliable outputs.

3.6 Cross-validation

Ten-fold stratified cross-validation technique is applied for the given datasets. The output of the model includes class statistics, confusion matrix, and detailed accuracy by the class table. Class statistics results contain correctly classified instances, accuracy, kappa statistics, mean absolute error, root mean squared error, relative absolute error, and root relative squared error. Detailed Accuracy by class table output information contains true positive rate, false positive rate, precision, recall, f-measure, and weighted averages of the output variables. Table 1 shows related formulas regarding the results of performance metrics. In table1, tp represents true positive, fn represents false negative, fp represents false positive, and lastly, tn represents true negative.

Table 1: Performance Metrics Table

Performance Metrics	Formula
True Positive Rate	TP/(TP + FN)
False Positive Rate	FP/(FP+TN)
Precision	TP/(TP+FP)
Recall	TP/(TP + FN)
F-Measure	2*precision*recall/(precision +recall)
Accuracy	TP+TN/(TP+TN+FP+FN)

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Support Vector Machine Results

Firstly, support vector machine (SVM) classifier is applied for CfsSubsetEval (CSE) evaluation results and support vector machine classifier overfitted the kernels which is called linear and radial kernels. In polynomial kernel, %83.05 accuracy reached with a 0.006 variance rate, in sigmoid kernel, %90.95 accuracy reached with a 1.43 variance rate when performing ten times random shuffled seed inputs of the parameters in stratified cross-validation technique. Library of support vector machine function is used with the type of C-SVC classification regarding the batch size of 100 and the cache size of 40. The tolerance of termination criterion epsilon determined 0.001 with a 1.0 cost rate, which is the default parameters of the SVM function in weka explorer software.

Table 2: SVM Statistics of CfsSubsetEval (CSE) evaluation results with Polynomial Kernel

Kappa statistic	0.744
Mean absolute error	0.1132
Root mean squared error	0.3364
Relative absolute error	26.6433%
Root relative squared error	73.0012%
Total Number of Instances	1520

As table 2 shows, polynomial kernel statistics, kappa statistic error, mean absolute error, and root mean squared error were calculated as 0.74, 0.11, and 0.33, respectively..

Table 3: Detailed Accuracy by Class Table of CfsSubsetEval(CSE) evaluation results with SVM Polynomial Kernel

Average	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
	1.000	0.222	0.581	1.000	0.735	D
	0.807	0.000	1.000	0.807	0.893	H
	0.732	0.000	1.000	0.732	0.845	A
Weighted Avg.	0.830	0.052	0.901	0.830	0.842	

Also, regarding table 3, the weighted averages of true positive rate, false positive rate, precision, recall, and f-measure are calculated as 0.830, 0.052, 0.901, 0.830, and 0.842, respectively.

Table 4: Confusion Matrix table of CfsSubsetEval(CSE) evaluation results with SVM Polynomial Kernel

A	B	C	<-- classified as
276	38	44	a = D
45	669	0	b = H
55	1	392	c = A

Table 4 shows the confusion matrix table of the support vector machine classifier. As shown in the table a,b, and c parameters represent for the draw, home team win, and away team win, respectively. Our model has succeeded to predict draw matches with 276 instances, home team win situation with 669 instances and lastly, away team win situation with 392 instances.

Table 5: Sensitivity Table of CfsSubsetEval (CSE) evaluation results with SVM Polynomial Kernel

Class Parameter	Sensitivity Rate
D	1.000
H	0.807
A	0.732
Average	0.830

Table 5 illustrates the sensitivity information of the polynomial kernel performance parameters. In table 5, ‘d', ‘h', ‘a' and ‘Average' represent the draw, home team win, away team win, and the average of the performance parameters respectively. As table 5 shows, class draw, home team win, and away team win sensitivity rates are 1.000, 0.807, and 0.732. Average of these three classes are calculated as 0.830.

Table 6: Specificity Table of CfsSubsetEval(CSE) evaluation results with SVM Polynomial Kernel

Class Parameter	Specificity Rate
D	0.222
H	0.000
A	0.000
Average	0.052

Table 6 illustrates the specificity information of the polynomial kernel performance parameters. In table 6, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters respectively. As table 6 shows, class draw, home team win, away team win, and average sensitivity rates are calculated as 0.222, 0.000, 0.000, and 0.052.

Table 7: Statistics of CfsSubsetEval(CSE) evaluation results with SVM Sigmoid Kernel

Kappa statistic	0.8517
Mean absolute error	0.0632
Root mean squared error	0.2513
Relative absolute error	14.8707 %
Root relative squared error	54.5382 %
Total Number of Instances	1520

As table 7 illustrates, kappa statistic error, mean absolute error, and root mean squared error are calculated 0.85, 0.06, and 0.25, respectively.

Table 8: Detailed Accuracy by Class Table of CfsSubsetEval(CSE) evaluation results with SVM Sigmoid Kernel

Average	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
	0.844	0.071	0.784	0.844	0.813	D
	0.922	0.062	0.929	0.922	0.925	H
	0.929	0.010	0.974	0.929	0.951	A
Weighted Avg.	0.905	0.049	0.908	0.905	0.906	

Also, regarding table 8, the weighted averages of true positive rate, false positive rate, precision, recall, and f-measure are calculated as 0.905, 0.049, 0.908, 0.905, and 0.906.

Table 9: Confusion Matrix table of CfsSubsetEval(CSE) evaluation results with SVM Sigmoid Kernel

A	B	C	<-- classified as
302	48	8	a = D
53	658	3	b = H
30	2	416	c = A

Table 9 shows the confusion matrix table of the support vector machine classifier. As shown in table 9 a, b and c parameters represent for the draw, home team win, and away team win respectively. Our model has succeeded to predict draw matches with 302 instances, home team win situation with 658 instances and lastly, away team win situation with 416 instances.

Table 10: Sensitivity Table of CfsSubsetEval(CSE) evaluation results with SVM Sigmoid Kernel

Class Parameter	Sensitivity Rate
D	0.844
H	0.922
A	0.929
Average	0.905

Table 10 illustrates the sensitivity information of the polynomial kernel performance parameters. In table 11, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters respectively. As table 10 shows, class draw, home team win, and away team win sensitivity rates are 0.844, 0.922, and 0.929. Average of these three classes are calculated as 0.905.

Table 11: Specificity Table CfsSubsetEval(CSE) evaluation results with SVM Sigmoid Kernel

Class Parameter	Specificity Rate
D	0.071
H	0.062
A	0.010
Average	0.049

Table 11 illustrates the specificity information of the polynomial kernel performance parameters. In table 11, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters, respectively. As table 11 shows, class draw, home team win, away team win, and average sensitivity rates are calculated as 0.071, 0.062, 0.010, and 0.049.

Secondly, support vector machine (SVM) classifier is applied for Classifier Attribute Eval(CAE) evaluation results and, support vector machine classifier overfitted the problem on linear, polynomial and sigmoid kernel type and they cannot determine reliable solutions, in radial kernel classifier reach %81.09 percent accuracy reached with a 0.15 variance rate when performing ten times random shuffled seed inputs of the parameters in stratified cross-validation technique. The type of C-SVC classification is used regarding the batch size of 100 and the cache size of 40. The tolerance of termination criterion epsilon determined 0.001 with a 1.0 cost rate, which is the default parameters of the SVM function in weka explorer software.

Table 12: Statistics of Classifier Attribute Eval(CAE) evaluation results with SVM Radial Kernel

Kappa statistic	0.7011
Mean absolute error	0.1263
Root mean squared error	0.3554
Relative absolute error	29.7413 %
Root relative squared error	77.1287 %
Total Number of Instances	1520

As table 12 shows, polynomial kernel statistics, kappa statistic error, mean absolute error, and root mean squared error is calculated 0.70, 0.12, and 0.35 respectively.

Table 13: Detailed Accuracy by Class Table of Classifier Attribute Eval(CAE) evaluation results with SVM Radial Kernel

Average	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
	0.612	0.122	0.607	0.612	0.609	D
	0.913	0.112	0.879	0.913	0.896	H
	0.806	0.052	0.866	0.806	0.835	A
Weighted Avg.	0.811	0.097	0.811	0.811	0.810	

Also, regarding table 13, the weighted averages of true positive rate, false positive rate, precision, recall, and f-measure is calculated as 0.811, 0.097, 0.811, 0.811, and 0.810.

Table 14: Confusion Matrix table of Classifier Attribute Eval(CAE) evaluation results with SVM Radial Kernel

A	B	c	<-- classified as
219	84	55	a = D
61	652	1	b = H
81	6	361	c = A

Table 14 shows the confusion matrix table of the support vector machine classifier. As shown in the table a,b, and c parameters represent for the draw, home team win, and away team win respectively. Our model has succeeded to predict draw matches with 219 instances, home team win situation with 652 instances and lastly, away team win situation with 361 instances.

Table 15: Sensitivity Table of Classifier Attribute Eval(CAE) evaluation results with SVM Radial Kernel

Class Parameter	Sensitivity Rate
D	0.612
H	0.913
A	0.806
Average	0.811

Table 15 illustrates the sensitivity information of the polynomial kernel performance parameters. In table 15, ‘d', ‘h', ‘a' and ‘Average' represent draw, home team win, away team win, and the average of the performance parameters respectively. As table 15 shows, class draw, home team win, and away team win sensitivity rates are 0.612, 0.913, and 0.806. Average of these three classes are calculated as 0.811.

Table 16: Specificity Table Classifier Attribute Eval(CAE) evaluation results with SVM Radial Kernel

Class Parameter	Specificity Rate
D	0.122
H	0.112
A	0.052
Average	0.097

Table 16 illustrates the specificity information of the polynomial kernel performance parameters. In table 16, ‘d', ‘h', ‘a' and ‘Average' represent draw, home team win, away team win, and the average of the performance parameters respectively. As table 16 shows, class draw, home team win, away team win, and average sensitivity rates are calculated as 0.122, 0.112, 0.052, and 0.097.

Thirdly, support vector machine (SVM) classifier is applied with the backward elimination results and achieved %87.91 percent accuracy with a 0.43

variance rate when performing ten times random shuffled seed inputs of the parameters in the stratified cross-validation technique. Support vector machine classifier overfitted the problem on radial, polynomial and sigmoid kernel type and cannot determine reliable solutions; thus, linear kernel type is used for generalizing the problem on the dataset.

Table 17: Statistics of Backward Elimination Results with SVM Linear Kernel

Kappa statistic	0.8114
Mean absolute error	0.0803
Root mean squared error	0.2833
Relative absolute error	18.8981%
Root relative squared error	61.4816%
Total Number of Instances	1520

As table 17 shows us, kappa statistic error, mean absolute error, and root mean squared error are calculated as 0.81, 0.08, and 0.28 respectively when monitoring the output statistics.

Table 18: Detailed Accuracy by Class Table of Backward Elimination Results with SVM Linear Kernel

Average	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
	0.771	0.086	0.734	0.771	0.752	D
	0.937	0.048	0.945	0.937	0.941	H
	0.875	0.041	0.899	0.875	0.887	A
Weighted Avg.	0.880	0.055	0.882	0.880	0.881	

Also, regarding table 18, the weighted averages of true positive rate, false positive rate, precision, recall, and f-measure are calculated as 0.880, 0.055, 0.882, 0.880, and 0.881.

Table 19: Confusion Matrix table of Backward Elimination Results with SVM Linear Kernel

A	B	C	<-- classified as
276	38	44	a = D
45	669	0	b = H
55	1	392	c = A

Table 19 shows the confusion matrix table of the support vector machine classifier. As shown in thetable a,b, and c parameters represent for the draw, home team win, and away team win respectively. Our model has succeeded to predict draw matches with 276 instances, home team win situation with 669 instances and lastly, away team win situation with 392 instances.

Table 20: Backward Elimination Results with SVM Linear Kernel Sensitivity Table

Class Parameter	Sensitivity Rate
D	0.771
H	0.937
A	0.875
Average	0.880

Table 20 illustrates the sensitivity information of the performance parameters. In table 20, ‘d', ‘h', ‘a' and ‘Average' represent draw, home team win, away team win, and the average of the performance parameters respectively. As table 20 shows, class draw, home team win, and away team win sensitivity rates are 0.771, 0.937, and 0.875. Average of these three classes are calculated as 0.880.

Table 21: Backward Elimination Results with SVM Linear Kernel Specificity Table

Class Parameter	Specificity Rate
D	0.086
H	0.048
A	0.041
Average	0.055

Table 21 illustrates the specificity information of the performance parameters. In table 21, ‘d', ‘h', ‘a' and ‘Average' represent draw, home team win, away team win, and the average of the performance parameters respectively. As table 21 shows, class draw, home team win, away team win, and average sensitivity rates are calculated as 0.771, 0.937, 0.875, and 0.880 respectively.

4.2 K-NN Results

Firstly, K nearest neighbor classifier is applied with CfsSubsetEval(CSE) evaluation results and as shown in table 9, k nearest neighbor classifier reached %92.66 classification accuracy maximum with a 0.14 variance rate when performing ten times random shuffled seed inputs of the parameters in stratified cross-validation technique.

Table 22: K-NN Statistics of CfsSubsetEval(CSE) evaluation results

Kappa statistic	0.8855
Mean absolute error	0.0408
Root mean squared error	0.1713
Relative absolute error	0.1713 %
Root relative squared error	37.1838 %
Total Number of Instances	1520

Table 22 represents that the kappa statistic rate is calculated 0.88, mean absolute error rate is 0.04, and root mean squared error value is 0.17.

Table 23: Detailed Accuracy by Class Table of CfsSubsetEval(CSE) evaluation results with K-NN

Average	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
	0.872	0.052	0.839	0.872	0.855	D
	0.948	0.043	0.951	0.948	0.950	H
	0.938	0.015	0.963	0.938	0.950	A
Weighted Avg.	0.927	0.037	0.928	0.927	0.927	

Also, regarding table 23, maximum accuracy reached line parameters of true positive rate, false positive rate, precision, recall, and f-measure are calculated as 0.927, 0.037, 0.678, 0.928 and 0.927 respectively.

Table 24: Accuracy Table of CfsSubsetEval(CSE) evaluation results with K-NN

K	Accuracy	TP Rate	FP Rate	Precision	Recall	F-Measure
3	**92,66**	**0,927**	**0,037**	**0.928**	**0,927**	**0,927**
5	91,25	0,913	0,043	0,915	0,913	0,913
7	90,46	0,905	0,047	0,909	0,905	0,906
9	89,14	0,891	0,049	0,896	0,891	0,893
11	86,11	0,861	0,063	0,867	0,861	0,863

Table 24 represents K-NN accuracy table of experiment 1. In experiment 1, when k parameter is selected for 3 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 92.66, 0,927, 0.037, 0.928, 0.927 and 0.927. When k parameter is selected for 5 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 91.25, 0,913, 0.043, 0.915, 0.913 and 0.913. When k parameter is selected for 7 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 90.46, 0,905, 0.047, 0.909, 0.905 and 0.906. When k parameter is selected for 9 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 89.14, 0,891, 0.049, 0.896, 0.891 and 0.893. When k parameter is selected for 11 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 86.11, 0,861, 0.063, 0.867, 0.861 and 0.863.

Table 25: Confusion Matrix table of CfsSubsetEval(CSE) evaluation results with K-NN

A	B	c	<-- classified as
312	3	13	a = D
34	677	3	b = H
26	2	420	c = A

Table 25 shows the confusion matrix table when k parameter is selected for 3. As shown in the table 25, a,b and c parameters represent for draw, home team win and away team win respectively. Our model has succeed to predict draw matches with 312 instances, home team win situation with 677 instances and lastly, away team win situation with 420 instances.

Table 26: Sensitivity Table of CfsSubsetEval(CSE) evaluation results with K-NN

Class Parameter	Sensitivity Rate
D	0.872
H	0.948
A	0.938
Average	0.927

Table 26 illustrates the sensitivity information of the performance parameters. In table 26, ‘d', ‘h', ‘a' and ‘Average' represent draw, home team win, away team win, and the average of the performance parameters respectively. As table 26 shows, class draw, home team win, and away team win sensitivity rates are 0.872, 0.948, and 0.938. Average of these three classes are calculated as 0.927 respectively.

Table 27: Specificity Table of CfsSubsetEval(CSE) evaluation results with K-NN

Class Parameter	Specificity Rate
D	0.052
H	0.043
A	0.015
Average	0.037

Table 27 illustrates the specificity information of the performance parameters. In table 27, ‘d', ‘h', ‘a' and ‘Average' represent, draw, home team win, away team win, and the average of the performance parameters respectively. As table 27 shows, class draw, home team win, away team win, and average sensitivity rates are calculated as 0.052, 0.043, 0.015, and 0.037.

Table 28: Detailed Accuracy by Class Table of Classifier Attribute Eval(CAE) evaluation results with K-NN

K	Accuracy	TP Rate	FP Rate	Precision	Recall	F-Measure
3	63,35	0,634	0,188	0,641	0,634	0,636
5	64,53	0,645	0,194	0,643	0,645	0,643
7	64,93	0,649	0,197	0,629	0,649	0,637
9	66,71	0,667	0,185	0,651	0,667	0,657
11	**67,77**	**0,678**	**0,186**	**0,655**	**0,678**	**0,663**

Table 28 represents K-NN accuracy table of experiment 2. In experiment 2, when k parameter is selected for 3 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 63.35, 0,634, 0.188, 0.641, 0.634 and 0.636. When k parameter is selected for 5 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 64.53, 0,645, 0.194, 0.643, 0.645 and 0.643. When k parameter is selected for 7 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 64.93, 0,649, 0.197, 0.629, 0.649 and 0.637. When k parameter is selected for 9 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 66.71, 0,667, 0.185, 0.651, 0.667 and 0.657. When k parameter is selected for 11 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 66.77, 0,678, 0.186, 0.655, 0.678 and 0.663.

Secondly, K nearest neighbor classifier is applied with Classifier Attribute Eval(CAE) evaluation results and as shown in table 28, k nearest neighbor classifier reached %67.77 classification accuracy maximum when k value equals 11 with a 0.20 variance rate, performing ten times random shuffled seed inputs of the parameters in stratified cross-validation technique. Also, regarding table 29, maximum accuracy reached line parameters of true positive rate, false positive rate, precision, recall, and f-measure calculated as 0.678, 0.186, 0.655, 0.678 and 0.663.

Table 29: Statistics of Classifier Attribute Eval(CAE) evaluation results with K-NN

Kappa statistic	0.4814
Mean absolute error	0.283
Root mean squared error	0.3806
Relative absolute error	66.6251 %
Root relative squared error	82.5951 %
Total Number of Instances	1520

Table 29 represents that the kappa statistic rate is calculated 0.48, mean absolute error rate is 0.28, and root mean squared error value is 0.38.

Table 30: Confusion Matrix table of Classifier Attribute Eval(CAE) evaluation results with K-NN

A	B	c	<-- classified as
105	153	100	a = D
68	612	34	b = H
86	49	313	c = A

Table 30 shows the confusion matrix table when the k parameter is selected for 3. As shown in table 30, a,b and c parameters represent for the draw, home team win, and away team win respectively. Our model has succeeded to predict draw matches with 105 parameters, home team win situation with 612 parameters and lastly, away team win situation with 313 parameters.

Table 31: K-NN Sensitivity Table of Classifier Attribute Eval(CAE) evaluation results with K-NN

Class Parameter	Sensitivity Rate
D	0.293
H	0.857
A	0.699
Average	0.678

Table 31 illustrates the sensitivity information of the performance parameters. In table 31, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters respectively. As table 31 shows, class draw, home team win, and away team win sensitivity rates are 0.293, 0.857, and 0.699. Average of these three classes are calculated as 0.678.

Table 32: K-NN Specificity Table of Classifier Attribute Eval(CAE) evaluation results with K-NN

Class Parameter	Specificity Rate
D	0.133
H	0.251
A	0.125
Average	0.186

Table 32 illustrates the specificity information of the performance parameters. In table 32, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters respectively. As table 32 shows, class draw, home team win, away team win, and average sensitivity rates are calculated as 0.133, 0.251, 0.125, and 0.186.

Table 33: Accuracy Table of Backward Elimination Results with K-NN

K	Accuracy	TP Rate	FP Rate	Precision	Recall	F-Measure
3	65,65	0,657	0,179	0,658	0,399	0,394
5	66,44	0,664	0,181	0,665	0,664	0,664
7	67,89	0,679	0,176	0,668	0,679	0,672
9	68,02	0,681	0,174	0,668	0,681	0,673
11	**69,03**	**0,693**	**0,173**	**0,678**	**0,693**	**0,683**

Table 33 represents K-NN accuracy table of experiment 3. In experiment 3, when k parameter is selected for 3 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 65.65, 0,657, 0.179, 0.658, 0.399 and 0.394. When k parameter is selected for 5 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 66.44, 0,664, 0.181, 0.665, 0.664 and 0.664. When k parameter is selected for 7 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 67.89, 0,679, 0.176, 0.668, 0.679 and 0.672. When k parameter is selected for 9 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 68.02, 0,681, 0.174, 0.668, 0.681 and 0.673. When k parameter is selected for 11 accuracy rate, true positive rate, false positive rate, precision, recall and f-measure parameters are calculated as, 69.03, 0,693, 0.173, 0.678, 0.693 and 0.683.

Thirdly, k nearest neighbor classifier is applied with backward elimination results and as shown in table 33, k nearest neighbor classifier reached %69.03 classification accuracy maximum with a 0.22 variance rate when performing ten times random shuffled seed inputs of the parameters in stratified cross-validation technique when k parameter is selected for 11.

Table 34: Statistics of Backward Elimination Results with K-NN

Kappa statistic	0.5096
Mean absolute error	0.2722
Root mean squared error	0.3709
Relative absolute error	64.1002 %
Root relative squared error	80.4945 %
Total Number of Instances	1520

Table 34 represents that the kappa statistic rate is calculated 0.50, mean absolute error rate is 0.27, and root mean squared error value is 0.37.

Table 35: Detailed Accuracy by Class Table of Backward Elimination Results with K-NN

Average	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
	0.358	0.136	0.448	0.358	0.398	D
	0.853	0.228	0.768	0.853	0.808	H
	0.708	0.116	0.719	0.708	0.713	A
Weighted Avg.	0.693	0.173	0.678	0.693	0.683	

Also, regarding table 35, maximum accuracy reached line parameters of true positive rate, false positive rate, precision, recall, and f-measure calculated as 0.693, 0.173, 0.678, 0.693 and 0.683 respectively.

Table 36: Confusion Matrix table of Backward Elimination Results with K-NN

A	B	C	<-- classified as
128	139	91	a = D
72	609	33	b = H
86	45	317	c = A

Table 36 shows the confusion matrix table when the k parameter is selected for 3. As shown in table 36, a,b and c parameters represent for the draw, home team win, and away team win respectively. Our model has succeeded to predict draw matches with 128 parameters, home team win situation with 609 parameters and lastly, away team win situation with 317 parameters.

Table 37: Sensitivity Table of Backward Elimination Results with K-NN

Class Parameter	Sensitivity Rate
D	0.358
H	0.853
A	0.708
Average	0.693

Table 37 illustrates the sensitivity information of the performance parameters. In table 37, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters respectively. As table 37 shows, class draw, home team win, and away team win sensitivity rates are 0.358, 0.853, and 0.708. Average of these three classes are calculated as 0.693.

Table 38: Specificity Table of Backward Elimination Results with K-NN

Class Parameter	Specificity Rate
D	0.136
H	0.228
A	0.116
Average	0.173

Table 38 illustrates the specificity information of the performance parameters. In table 38, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters respectively. As table 38 shows, class draw, home team win, away team win, and average sensitivity rates are calculated as 0.136, 0.228, 0.116, and 0.173.

4.3 Artificial Neural Network Results

Artificial neural network classifier overfitted the problem when CfsSubsetEval (CSE) was applied. When performing ten times random shuffled seed inputs of the parameters in stratified cross-validation technique with five input nodes in the input layer, five, ten and fifteen nodes are separately tested for each iteration in the hidden layer, and three nodes in output layer model applied with a batch size of 100 and training time of 500. Momentum value and learning rate value are applied respectively 0.1 to 0.5 in each experiment, but the results show that classifier cannot generalize the problem.

Artificial neural network classifier is applied with Classifier Attribute Eval(CAE) evaluation results and reached %83.41 classification accuracy maximum with a 1.77 variance rate when performing ten times random shuffled seed inputs of the parameters in stratified cross-validation technique. Twenty-three input nodes in the input layer, five nodes in the hidden layer and three nodes in output layer model are applied with a batch size of 100 and training time of 500, momentum value and learning rate values are considered as 0.1.

Table 39: Classifier Attribute Eval(CAE) evaluation results with ANN Statistics

Kappa statistic	0.747
Mean absolute error	0.1222
Root mean squared error	0.2918
Relative absolute error	28.7803 %
Root relative squared error	63.3291 %
Total Number of Instances	1520

Table 39 represents the kappa statistic rate is calculated as 0.74, mean absolute error rate is 0.12, and root mean squared error value is 0.29.

Table 40: Classifier Attribute Eval(CAE) evaluation results with ANN Detailed Accuracy by Class Table

Average	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
	0.637	0.097	0.669	0.637	0.652	D
	0.912	0.099	0.891	0.912	0.901	H
	0.886	0.048	0.886	0.886	0.886	A
Weighted Avg.	0.839	0.084	0.837	0.839	0.838	

Also, regarding table 40, maximum accuracy reached line parameters of true positive rate, false positive rate, precision, recall, and f-measure calculated as 0.839, 0.084, 0.837, 0.839 and 0.838.

Table 41: Classifier Attribute Eval(CAE) evaluation results with ANN Confusion Matrix Table

A	B	c	<-- classified as
228	80	50	a = D
62	651	1	b = H
51	0	397	c = A

Table 41 shows the confusion matrix table of artificial neural network classifier. As shown in table 41, a,b and c parameters represent for the draw, home team win, and away team win. Our model has succeeded to predict draw matches with 228 parameters, home team win situation with 651 parameters and lastly, away team win situation with 397 parameters

Table 42: Classifier Attribute Eval(CAE) evaluation results with ANN Sensitivity Table

Class Parameter	Sensitivity Rate
D	0.637
H	0.912
A	0.886
Average	0.839

Table 42 illustrates the sensitivity information of the performance parameters. In table 42, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters respectively. As table 42 shows that class draw, home team win, and away team win sensitivity rates are 0.637, 0.912, and 0.886. Average of these three classes are calculated as 0.839.

Table 43: Classifier Attribute Eval(CAE) evaluation results with ANN Specificity Table

Class Parameter	Specificity Rate
D	0.097
H	0.099
A	0.048
Average	0.084

Table 43 illustrates the specificity information of the performance parameters. In table 42, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters respectively. As table 42 shows, class draw, home team win, away team win, and average sensitivity rates are calculated as 0.097, 0.99, 0.048, and 0.084.

Artificial neural network classifier is applied backward elimination results and reached %85.06 classification accuracy maximum with a 3.51 variance rate when performing ten times random shuffled seed inputs of the parameters in stratified cross-validation technique.

Table 44: Backward Elimination Results with ANN Statistics

Kappa statistic	0.7617
Mean absolute error	0.1166
Root mean squared error	0.2892
Relative absolute error	27.4545 %
Root relative squared error	62.7588 %
Total Number of Instances	1520

Table 44 represents the kappa statistic rate is calculated as 0.76, mean absolute error rate is 0.11, and root mean squared error value is 0.28.

Table 45: Backward Elimination Results with ANN Detailed Accuracy by Class Table

Average	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
	0.575	0.065	0.733	0.575	0.645	D
	0.971	0.104	0.892	0.971	0.930	H
	0.879	0.063	0.853	0.879	0.866	A
Weighted Avg.	0.851	0.083	0.843	0.851	0.844	

Also, regarding table 45, maximum accuracy reached line parameters of true positive rate, false positive rate, precision, recall, and f-measure calculated as 0.851, 0.083, 0.843, 0.851 and 0.844 respectively.

Table 46: Backward Elimination Results with ANN Confusion Matrix Table

A	B	c	<-- classified as
206	84	68	a = D
21	693	0	b = H
54	0	394	c = A

Table 46 shows the confusion matrix table of artificial neural network classifier. As shown in table 46, a, b and c parameters represent for the draw, home team win, and away team win respectively. Our model has succeeded to predict draw matches with 206 instances, home team win situation with 693 instances and lastly, away team win situation with 394 instances.

Table 47: Backward Elimination Results with ANN Sensitivity Table

Class Parameter	Sensitivity Rate
D	0.575
H	0.971
A	0.879
Average	0.851

Table 47 illustrates the sensitivity information of the performance parameters. In table 47, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters respectively. As table 47 shows, class draw, home team win, and away team win sensitivity rates are 0.575, 0.971, and 0.879. Average of these three classes are calculated as 0.851.

Table 48: Backward Elimination Results with ANN Specificity Table

Class Parameter	Specificity Rate
D	0.065
H	0.104
A	0.063
Average	0.083

Table 48 illustrates the specificity information of the performance parameters. In table 48, ‘d', ‘h', ‘a' and ‘Average' represents draw, home team win, away team win, and the average of the performance parameters respectively. As table 48 shows, class draw, home team win, away team win, and average sensitivity rates are calculated as 0.065, 0.104, 0.063, and 0.083 respectively.

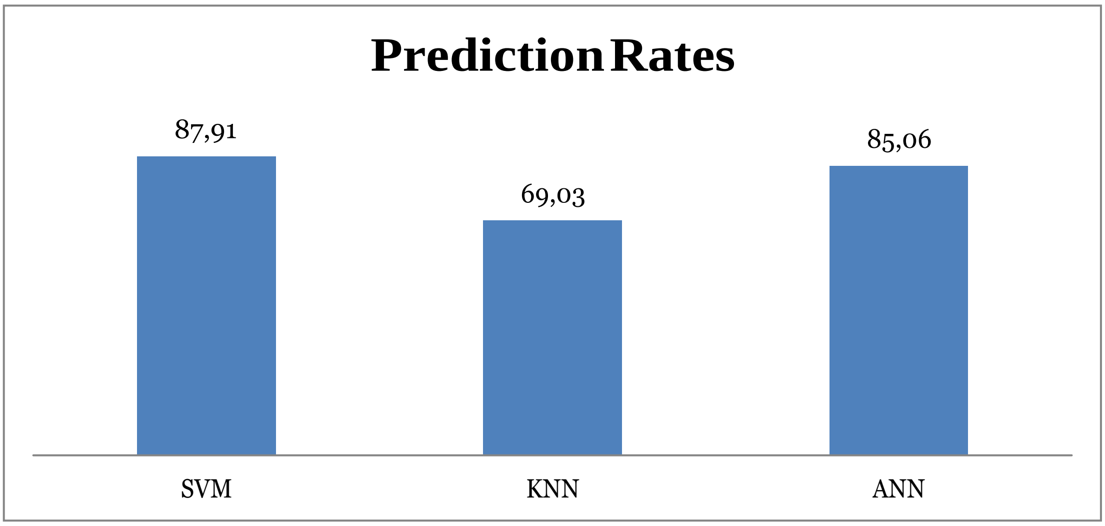

Figure 17: Classifiers Prediction Rates of Backward Elimination Results

4.4 Discussion

The aim of this thesis is to classify and predict the football game results with high accuracy and also, to find high important attributes and variables information that can be directly related to the Spanish La Liga game results. Machine-learning algorithms which are support vector machine (SVM), k nearest neighbors (k-NN) and artificial neural networks (ANN) were used to investigate performance measures. The results show us that features that are chosen with backward elimination technique has reached higher prediction rates than other evaluation results. In backward elimination results, as table 49 illustrates, support vector machine, k nearest neighbors, and artificial neural network classifiers achieve %87.91, %69,03 and %85.06 accuracy rate. According to the results of backward elimination, support vector machine classifier has better performance results than artificial neural network classifier and k nearest neighbors classifier. Due to the linearly structured problem, the linear kernel of the support vector machine classifier (SVM) may best fit and generalize the problem. Also, regarding this situation, artificial neural network classifier (ANN) achieved closer results with support vector machine classifier (SVM) and feedforward model with backpropagation technique may have shown a positive effect of generalizing the problem. K nearest neighbors classifier (k-NN) results are lower than SVM and ANN classifiers, and it may be the result of the problem-solving methodology of the k-NN algorithm that cannot fit and generalize the problem as well as SVM and ANN algorithms. However, when the k-NN algorithm was applied to the dataset, it was observed that the accuracy increased as the k value increased. Therefore, when considering the baseline accuracy of %46.97, which is calculated by ZeroR algorithm in a given dataset, k nearest neighbors (k-NN) classifier results can be considered as the average classifier performance. When analyzing the literature study of football prediction, Ulmer et al. [2], Baboota et al. [4] and Ganesan et al. [4] used English Premier League dataset in order to achieve highest prediction results, and both three scientific research used support vector machine classifier during the researches. Joseph et al. [5] predicted the outcome of the results regarding Tottenham Hotspur football club, which is an English Premier League team, used k-nearest neighbor classifier during the research. Hucaljuk et al. [1] developed a software system and Bunker et al. [9] proposed a machine learning framework for sport prediction results and both studies used artificial neural network and k nearest neighbor classifier to predict outcomes. Prucker et al. [3] predicted the outcomes of national football league games with using artificial neural network classifier. Zaveri et al.[72] used Spanish La Liga dataset with artificial neural network and support vector machine classifiers in order to predict outcomes. Pallingi et al. [73] proposed a machine learning model for football prediction and used Spanish La Liga dataset. Similar to our study, support vector machine and k-nearest neighbor classifiers are used in their study. Cao et al. [10] built a model which predicts the outcome of the national basketball association league basketball matches with using classifiers, support vector machine, and artificial neural network just as in our research. Also, Hucaljuk et al. [1], Baboota et al. [4], Ganesan et al. [6] and Huang et al. [8] used feature selection techniques and dimensionally reduction technique. Techniques and methodologies of our study show parallelism with the literature regarding the data preprocessing and learning models. Also, our results show affinity with the literature and even higher performance results regarding classifier performance parameters and accuracy. As Igiri et al. [7] discussed in the recommendation part of the study, our in-game performance implications from the statistical data may guide young researchers, scout organizations and football clubs to a proper sense of direction.

REFERENCES

[1] Hucaljuk, Josip & Rakipovic, Alen, “Predicting football scores using machine learning techniques,” pp.1623-1627,2011

[2] Ulmer, B., Fernandez, M., & Peterson, M, “Predicting Soccer Match Results in the Premier League,” Phd thesis, 2013

[3] “M. C. Purucker, “Neural network quarterbacking,” IEEE Potentials, vol. 15, no. 3, pp. 9–15, 1996.Available: http://www.oalib.com/references/13085413. [Accessed: 25-Apr-2019].

[4] R. Baboota and H. Kaur, "Predictive analysis and modeling football results using machine learning approach for English Premier League," *International Journal of Forecasting*, vol. 35, no. 2, pp. 741–755, 2019.

[5] A. Joseph, N. Fenton, and M. Neil, “Predicting football results using Bayesian nets and other machine learning techniques,” *Knowledge-Based Systems*, vol. 19, no. 7, pp. 544–553, 2006.

[6] Anad Ganesan and Harini M, “ENGLISH FOOTBALL PREDICTION USING MACHINE LEARNING CLASSIFIERS,” *International Journal of Pure and Applied Mathematics*, 2018

[7] C. P. Igiri, “An Improved Prediction System for Football a Match Result,” *IOSR Journal of Engineering*, vol. 04, no. 12, pp. 12–020, 2014.

[8] K. Huang and W. Chang, "A neural network method for prediction of 2006 World Cup Football Game," *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pp. 1-8, 2010.

[9] R. P. Bunker and F. Thabtah, “A machine learning framework for sport result prediction,” *Applied Computing and Informatics*, vol. 15, no. 1, pp. 27–33, 2019.

[10] Cao, C., “Sports data mining technology used in basketball outcome prediction,” Master’s Thesis, Dublin Institute of Technology, 2012.

[11] “Machine learning,” *Wikipedia*, 20-Jun-2019. [Online]. Available: https://en.wikipedia.org/wiki/Machine_learning. [Accessed: 22-Jun-2019]

[12] E. Alpaydin, “*Introduction to machine learning*,” Cambridge, MA: The MIT Press, 2014.

[13] El Naqa I., Murphy M.J, “What Is Machine Learning?,” Machine Learning in Radiation Oncology. Springer, Cham, 2015

[14] T. M. Mitchell, “*The discipline of machine learning*,” Pittsburgh, PA: Carnegie Mellon University, School of Computer Science, Machine Learning Dept., 2006.

[15] C. M. BISHOP, "*PATTERN RECOGNITION AND MACHINE LEARNING*," SPRINGER-VERLAG NEW YORK, 2016.

[16] H. Lv and H. Tang, "Machine Learning Methods and Their Application Research," *2011 2nd International Symposium on Intelligence Information Processing and Trusted Computing*, Hubei, 2011,pp.108-110. doi: 10.1109/IPTC.2011.34

[17] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervousactivity," *Springer Link*. [Online]. Available: https://link.springer .com/article/10.1007/BF02478259. [Accessed: 23-Jun-2019].

[18] A. M. Turing, "Computing Machinery and Intelligence," *Parsing the Turing Test*, pp. 23–65, 2009.

[19] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain.," *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958.

[20] M. Pelillo, "Alhazen and the nearest neighbor rule," *Pattern Recognition Letters*," vol. 38, pp. 34–37, 2014.

[21] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, no. 4, pp. 193–202, 1980.

[22] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *PNAS*, 01-Apr-1982. [Online]. Available: https://www.pnas.org/content/79/8/2554.abstract. [Accessed: 23-Jun-2019].

[23] C. J. C. H. Watkins, "Learning from delayed rewards," Master's thesis, 1989.

[24] G. Tesauro, "Temporal difference learning and TD-Gammon," *Communications of the ACM*, vol. 38, no. 3, pp. 58–68, 1995.

[25] T. K. Ho, "Random decision forests," *Proceedings of 3rd International Conference on Document Analysis and Recognition*, Montreal, Quebec, Canada, 1995, pp.278-282vol.1. doi: 10.1109/ICDAR.1995.598994

[26] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*. [Online]. Available: https://dl.acm.org/citation.cfm?id=218929. [Accessed: 23-Jun-2019].

[27] M.Alphonso, "Machine Learning for Risk Management in MATLAB," 2013. Available:https://www.mathworks.com/content/dam/mathworks/mathworks-dot-com/company/events/conferences/matlab-computational-finance-conference-nyc/2017/proceedings/machine-learning-for-risk-modeling-in-matlab.pdf [Accessed June 15, 2019].

[28] J. Bootkrajang, "Supervised learning with random labelling errors," Master's thesis, 2013.

[29] "Logistic regression," *Wikipedia*, 19-Jun-2019. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvTG9naXN0aWNfcmVncmVzc2lvbg. [Accessed: 24-Jun-2019].

[30] D. C. Montgomery, E. A. Peck, and G. G. Vining, "*Introduction to Linear Regression Analysis*," Chicester: Wiley, 2013.

[31] N. J. Nilsson "Introduction to Machine Learning," *Goodreads*. [Online]. Available:https://www.goodreads.com/book/show/16173001-introduction-to-machine-learning. [Accessed: 23-Jun-2019].

[32] M. S. Aldenderfer and R. K. Blashfield, "*Cluster analysis*," Newbury Park: Sage Publications, 2006.

[33] "Cluster analysis," *Wikipedia*, 16-Jun-2019. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvQ2x1c3Rlcl9hbmFseXNpcw. [Accessed: 24-Jun-2019].

[34] L. Parsons, E. Haque, and H. Liu, "Subspace clustering for high dimensional data," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 90–105, 2004.

[35] V. N. Vapnik, "*Statistical learning theory*," New York: Wiley, 1998.

[36] "Support Vector Machines for classification and locating faults on transmissionlines" [Online].Available:https://www.researchgate.net/publication/257635405_Support_Vector_Machines_for_classification_and_locating_faults_on_transmission_lines. [Accessed: 26-Apr-2019].

[37] Joachims. T, "Learning to Classify Text Using Support Vector Machines," Springer. [Online]. Available: https://www.springer.com/la/book/9780792376798. [Accessed: 26-Apr-2019].

[38] "S N Sivanandam S Sumathi S N Deepa - AbeBooks." [Online]. Available: https://www.abebooks.com/book-search/author/s-n-sivanandam-s-sumathi-s-n-deepa/. [Accessed: 26-Apr-2019].

[39] "*A simplest introduction to Support Vector Machine,"* 07-Aug-2014*"*.[Online]. Available:https://diggdata.in/post/94066544971/support-vector-machine-without-tears [Accessed: 11-May-2019].

[40] "Doğrusal Regresyona (Linear Regression) giriş," *Doğrusal Regresyona (Linear Regression) giriş*, 15-Sep-2018. [Online]. Available: https://www.datafloyd.com/tr/veri-biliminin-temelleri-dogrusal-regresyona-giris/. [Accessed: 14-Aug-2019].

[41] R. O. Duda, P. E. Hart, and D. G. Stork, "*Pattern classification*," Wiley-Interscience , 2000.

[42] "K-nearest neighbors algorithm," Wikipedia, 21-Jul-2019. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvSy1uZWFyZXN0X25laWdoYm9yc19hbGdvcml0aG0jY2l0ZV9ub3RlLTE. [Accessed: 01-Aug-2019].

[43] "Lazy learning," *Wikipedia*, 29-Jul-2019. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvTGF6eV9sZWFybmluZw. [Accessed: 01-Aug-2019].

[44] T. M. Cover and P. Hart, “Nearest neighbor pattern classification,” 2018. [Online]. Available: https://www.cs.bgu.ac.il/~adsmb182/wiki.files/borak-lecture notes.pdf. [Accessed: 01-Aug-2019].

[45] S. Sayad, “K Nearest Neighbors - Classification,” *KNN Classification*. [Online]. Available: https://www.saedsayad.com/k_nearest_neighbors.htm. [Accessed: 01-Aug-2019].

[46] K. Nar, “*Basit Doğrusal Regresyon,*23-Oct-2018*”*.[Online]. Available: https://kadirnar.com/post/basit-dogrusal-regresyon-simple-linear-regression/51 [Accessed: 15-Jun-2019].

[47] K. Chomboon, P. Chujai, P. Teerarassammee, K. Kerdprasop, and N. Kerdprasop, “An Empirical Study of Distance Metrics for k-Nearest Neighbor Algorithm,” *The Proceedings of the 2nd International Conference on Industrial Application Engineering 2015,* Jan. 2015.

[48] J. Han, M. Kamber, and J. Pei, “*Data mining: concepts and techniques*,” Amsterdam: Morgan Kaufmann, 2012.

[49] “Minkowski Distance,” *Wikipedia*, 28-May-2019. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvTWlua293c2tpX2Rpc3RhbmNl. [Accessed: 01-Aug-2019].

[50] “Artificial Neural Network,” *Wikipedia*, 10-Jun-2019. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvQXJ0aWZpY2lhbF9uZXVyYWxfbmV0d29yaw. [Accessed: 24-Jun-2019].

[51] M. Kubat, “*INTRODUCTION TO MACHINE LEARNING*,” S.l.: SPRINGER INTERNATIONAL PU, 2018.

[52] T. M. Mitchell, “*Machine learning*,” New York: McGraw Hill, 2017.

[53] D. Graupe, “*Principles of artificial neural networks*,” Singapore: World Scientific Publ., 2013.

[54] M. Kubat, “*INTRODUCTION TO MACHINE LEARNING*,” S.l.: SPRINGER INTERNATIONAL PU, 2018.

[55] N. Hiếu,“*Neural Networks,”* 13-Aug-2016*”*. [Online]. Available: https://dataminingduchieu.wordpress.com/2016/08/13/first-blog-post/ [Accessed: 07-May-2019].

[56] N. Hiếu, “*Neural Networks,”* 13-Aug-2016*”*. [Online]. Available: https://drive.google.com/file/d/0BxrAlmn48iHkX0dMY2liV0RTQ0k/view [Accessed: 07-May-2019].

[57] “*Cluster Analysis Two Examples,*10-Jun-2016*”*.[Online]. Available: http://ichrome.com/blogs/archives/221 [Accessed: 12-Jun-2019].

[58] P. Jeffcock, “*4 Machine Learning Techniques You Should Recognize,*16-Nov-2017*”*. [Online]. Available: https://blogs.oracle.com/bigdata/machine-learning-techniques. [Accessed: 10-Jun-2019].

[59] “Feature selection,” *Wikipedia*, 06-Jun-2019. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvRmVhdHVyZV9zZWxlY3Rpb24

Eub3JnL3dpa2kvRmVhdHVyZV9zZWxlY3Rpb24. [Accessed: 24-Jun-2019].

[60] G. James, “*An introduction to statistical learning: with applications in R*,” New York: Springer, 2017.

[61] I. Guyon and A. Elisseeff, “An Introduction to Feature Extraction,” *Feature Extraction Studies in Fuzziness and Soft Computing*, pp. 1–25, 2003.

[62] S. Kaushik and Saurav, “Feature Selection methods with example (Variable selection methods),” *Analytics Vidhya*, 24-Jun-2019. [Online]. Available: https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example-or-how-to-select-the-right-variables/. [Accessed: 24-Jun-2019].

[63] S. Yang, J. Wang, H. Luo, and C. Jie, “Feature selection in machine learning: A new perspective,” *Neurocomputing*, Jul. 2018.

[64] “Brute Force Seach,” *Wikipedia*, 01-Apr-2019. [Online]. Available: http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvQnJ1dGUtZm9yY2Vfc2VhcmNo. [Accessed: 24-Jun-2019].

[65] L. A. Smith and M. A. Hall, “Practical feature subset selection for machine learning,” *Research Commons*, 02-Dec-2008. [Online]. Available: https://researchcommons.waikato.ac.nz/handle/10289/1512. [Accessed: 25-Jun-2019].

[66] P. Tüfekci, “Prediction of Football Match Results in Turkish Super League Games,” *Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015 Advances in Intelligent Systems and Computing*, pp. 515–526, 2016.

[67] M. A. Cengiz, N. Murat, and H. Koç, “A Bayesian Computation for The Prediction Of Football Match Results Using Artificial Neural Network - Semantic Scholar,” *[PDF] A Bayesian Computation for The Prediction Of Football Match Results Using Artificial Neural Network - Semantic Scholar*, 01-Jan-1970. [Online]. Available: https://www.semanticscholar .org/paper/A-Bayesian-Computation-for-The-Prediction-Of-Match-Cengiz-Murat/7f1cd0c0957ac578d6311e69c385ef879abd2058. [Accessed: 24-Jun-2019].

[68] M. Karabıyık and B. Yet, “Football Analytics using Bayesian Networks: the FutBA Model,” *Pamukkale University Journal of Engineering Sciences*, vol. 25, no. 1, pp. 121–131, 2019.

[69] B. Karaoğlu, “Makine Öğrenmesi ile Spor Karşılaşmalarının Modellenmesi,” *EMO BİLİMSEL DERGİ*, 30-Jun-2015. [Online]. Available: https://dergipark.org.tr/emobd/issue/26508/278945. [Accessed: 24-Jun-2019].

[70] E. Esme and M. S. Kiran, “Prediction of Football Match Outcomes Based on Bookmaker.” [Online]. Available: http://www.ijmlc.org/vol8/658-A05.pdf. [Accessed: 24Jun-2019].

[71] Pollock, R. and Kariv, A. (2019). *Spanish La Liga (football)*. [online] DataHub. Available at: https://datahub.io/sports-data/spanish-la-liga [Accessed 18 Jun. 2019].

[72] N. Zaveri, U.Shah, S.Tiwari, P.Shinde, and L.T.Teli, "Prediction of Football Match Score and Decision Making Process", International Journal on Recent and Innovation Trends in Computing and Communication, Volume: 6 Issue: 2, pp. 162-165, 2018

[73] D. A. Pallingi, F.Ramos, J.Torres-Sospedra, S.T.Oliver and J.Huerta, "Using Weather Conditionand Advanced Machine Learning Methods to Predict Soccer Outcome", 2018