Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2021) 29: 1672 – 1691

doi:10.3906/elk-2005-27

A case study on player selection and team formation in football with machine learning

Didem ABİDİN*
Department of Computer Engineering, Faculty of Engineering, Manisa Celal Bayar University, Manisa, Turkey

Received: 03.05.2020 • **Accepted/Published Online:** 08.12.2020 • **Final Version:** 31.05.2021

Abstract: Machine learning has been widely used in different domains to extract information from raw data. Sports is one of the popular domains for researchers to work on recently. Although score prediction for matches is the most preferred application area for artificial intelligence, player selection, and team formation is also an application area worth working on. There are some studies in the literature about player selection and team formation which are examined in this study. The study has two important contributions: First one is to apply seven different machine learning algorithms on our dataset to find the best player combination for the U13 team of Altınordu Football Academy and comparing the results with that of the coach's lineup and lineups of 20 matches played in 2019–2020 season. Second is combining the data obtained from the trainings of the players and coach evaluations of the players and feeding the machine to make more accurate predictions. The data from the trainings is gathered with Hit/it Assistant and the coach evaluations of the players are stated by the golden standard according to eighteen criteria stated in the literature. Synthetically generated data is also used in the final dataset to obtain more accurate classification results. Another remarkable aspect of the study is that no match data is used to form the team to be proposed for the next match, instead real match data is only used for evaluation. The results show that machine learning algorithms can be used for player selection and team formation process because random forest algorithm, which is executed on WEKA environment, can make player selections with 93.93% reliability and the lineup suggestions of these algorithms are 97.16% similar to coach's ideal team and also the best performing algorithm has an average performance of 89.36% for team formation when compared with the match lineups of 2019–2020 football season.

Key words: Machine learning, data mining, player selection, sports training

1. Introduction

Football (the name "soccer" is also used in some countries) is one of the most popular sports in the world. It is played with eleven players on the field but the team should contain more footballers for the coaches to choose the correct players for a match. The process of player selection and team formation is a complex problem in which many criteria must be taken into consideration. Coaches need to analyze the players from many different aspects, from physical to mental conditions. There are many parameters for a player which affects the probability of being chosen for the team. These parameters include some qualitative and quantitative evaluations. They may also include the player's skills and performance statistics, a combination of players' physical fitness, psychological factors, and injuries [1]. Some coaches use certain weight values to be sure about the correct analysis of the quantitative data of the players. For the analysis of quantitative data, the weight

*Correspondence: doktem@hotmail.com

coefficients are useful to coaches because they show attributes affecting the player selection process in numerical values. This study also benefits from some weight coefficients given in the literature.

Since football is such a popular team sport in almost all countries of the world, the player selection process for a team is a very important task to be accomplished to win a match. A wrong selection of a player in a very important match may cost huge amounts of money if the chosen combination of players fails. Traditionally, professional soccer teams use many evaluation assessments to decide about team formation. These assessments provide great benefit and the ability to select suitable players for an effective team formation is mandatory to be successful in team sports [2]. Coaches need to apply many tests to the players to find out the featured skills of the players. The skills and physical abilities of a player both affect the probability to be chosen to the team and the position of that player in the team and during the game.

The player selection problem can be widened with the addition of a position decision problem for every player in the team. The coach has to find the correct player for each position in the team, which is also called team formation. Although the positions of the players do not change so much, the coach may need to change the player of that position for a certain match. To make the player selection process more accurate, data mining techniques may help coaches to make the correct decisions. These techniques require data, which must be clarified, organized, and processed in the first place. In this study, machine learning (ML) algorithms are applied to player selection and team formation to make player selections and position recommendations for the players of a football team. The data belongs to the U13 infrastructure team of the ALtınordu Football Academy (ALFA), Turkey. This data has two different subsets to form the whole. The first part of the data is from Hit/it Assistant, which is a device used to improve young player performances in any kind of sports played with a ball [1]. ALFA uses Hit/it effectively in all training programs of all age groups in its infrastructure. This data is as valuable as match data for player selection decisions. The second part consists of the data obtained from coach evaluations of the players. These evaluations are generated after the players were observed for a long training period of 12 months. The study has two important contributions: First one is to apply seven different machine learning algorithms on our dataset to find the best player combination for the U13 team of ALFA and comparing the results with that of the coach's lineup and lineups of 20 matches played in 2019–2020 season. Second contribution is generating a new dataset with the training data obtained from Hit/it with coach evaluations and making player position predictions for the following matches "without using any match data". The results are compared with the unseen test data, which is the coach's lineup and lineups of 20 consecutive matches played in 2019–2020 football season to emphasize that the output of the ML algorithms used are also reliable to use. The paper is organized as follows: Section 2 covers the related work. Section 3 explains the data acquisition and properties of data with preprocessing steps. Section 4 explains the machine learning algorithms used in the study and Section 5 presents the experiments and their results with the evaluation phase of the study. Section 6 covers the conclusion and future work. The ML algorithms are executed on the WEKA environment, which is a tool containing many ML algorithms for data mining [3].

2. Related work

In sports, a huge amount of data is generated by some devices which monitor players during training and matches. This data can easily be used for analyzing the players, where it can also be used for making predictions about the game results. To use this data properly, the coaches should have some knowledge from different areas

[1] Performa.nz (2021). Performance Measurement in Sports and Military [online]. Website http://performa.nz/ [accessed 20 May 2018].

like anatomy, physiology, biomechanics, and psychology. Combining the coach's experience with the analysis results, he/she can assign a training program to each player. The aim of training the players is to achieve maximum performance. The efficiency of a player can be increased by repeating targeted performances. Some physical conditions and external factors affect the performance of the player. The external factors are the ones like temperature, humidity, altitude, field condition, and nutrition. These affect the physical conditions of the player in endurance, speed, force, coordination, and flexibility [4].

The sports mining domain has emerged with different application areas, in which ML techniques are applied. These application areas that use ML on sports training can be listed as analyzing the performances in sports [5], rapid feedback systems [6], automatic evaluation of exercises [7], performance evaluation [8], exercise repetition detection [9], intelligent systems for personalized sport training [10] and planning the sports training sessions [11], which are not within the scope of our work. The world of sports now also embraced the information technologies in many phases like live analysis, statistics, or player performance analysis [12]. In most of the studies done in the sports domain, result prediction takes an important place since there are lots of people interested in predicting the results of the games. There are also some studies in computer science literature that deal with game result prediction in baseball, basketball, or soccer.

In another study [2], researchers use Bayesian logic (BLOG) and Markov logic networks (MLNs) for the prediction of NBA games. These methods have a success of 63% and 64%, respectively. Another study on NBA games in the 2015–2016 regular season deals with the prediction of the results with the design of experiment (DOE) method and obtains an improvement from 67.94% to 79.90% in prediction performances [13]. A similar study is done for predicting the results for the college football games by using data mining techniques on American football data from National Collegiate Athletic Association (NCAA) and the study gets remarkable improvement on prediction with 91% to 97% success in two adjacent football seasons [14]. On the other hand, [15] focuses on artificial neural networks (ANN) to make result predictions and proposes a framework in which ML is used as a learning strategy.

Player evaluation is one of the popular subjects in the sports mining domain. In [16], sports player evaluation is done with deep learning techniques which rely on text data, making the study an example of text mining in the sports domain. By doing so, statistics and analysis of news articles are used in a study for constructing a player evaluation model with computational intelligence (CI) techniques. Another study uses a parallel random tree algorithm for athlete evaluation, and results are compared with the results obtained with genetic algorithms (GA) [17]. Among all CI methods, the fuzzy logic approach is also a popular technique to apply to the sports training domain for the evaluation of strength in training exercises [18].

Another popular subject in the domain is player selection, which has as many studies as result prediction. For example, in [19], ML is used in athlete selection problems for cycling omnium by making performance predictions for the medalists. In [20], the fuzzy logic approach is used for both player selection and team formation in multiplayer sports. The fuzzy system applied chooses the best players and forms the best combination of players for a soccer team. This study can be considered as a guide to our study because a similar player selection is done by using similar evaluation methods but our study differs from this one with data and classification techniques used. Similarly, [21] used fuzzy techniques to select footballers for the Turkish National Football Team. Antecedents to these studies have also dealt with different team-building problems

[2]Jean-Baptise G, Liu X, Santiago D (2014). NBA game prediction based on historical data and injuries [online]. Website: http://dionny.gethub.io/MBAPredictions/website/ [accessed 07.03.2021]

by using task partitioning techniques [22], axiomatic design principles [23] and a linear optimization model for selecting players for soccer and volleyball [2]. Taking these studies as pioneers, the term sports data mining has emerged recently [24] [25] which puts forth its own sports data mining approach. For the player selection process, the procedure for player selection in n-player sports such as soccer can be considered as a complex multifactor problem with multiobjectives. In [26], a specific algorithm was developed to select football players age 15-17 by using the vertical jump, yoyo, 10 meters shuttle run and Hoff tests. A relatively new study handles the player selection problem for team formation by using population search techniques such as particle swarm optimization (PSO) [27], differential evolution (DE) [28] and artificial bee colony (ABC) [29]. The results obtained from the study are interesting because they got improved performances for some of the known benchmark data [30] [31].

This study is based on data generated by a performance improvement device explained in the following sections. The data generated with this device is used to support the player selection and prediction process in a football team. One of the contributions of this study to the literature is combining the data obtained from the trainings of the players and coach evaluations of the players and feeding the machine with combined and preprocessed data to make more accurate predictions without using any match data as input. The coach evaluations used are converted to quantitative values as done in other studies but in this study, they are supported with the training data. In training data, the improvement of players during the training is taken into consideration for player position classification and team formation. The methods that form the structure of this study are explained in the following section in detail.

3. Dataset

Dataset of this study consists of two sources as explained in the next two subsections. The first source is the data obtained from Hit/it device. This data consists of real and synthetic data of overall performances of players for 100 sequences of trainings with Hit/it. The second source depends on the evaluations of a human expert (coach of the team in our case), which is accepted as the golden standard. This dataset was created with the scores given by the coach and provides preliminary information about the possible positions of the players.

After preprocessing and standardization of two data sources, synthetic data generation is done, which is explained in subsection 3.4 to make the classification process more accurate. Then the overall dataset is used as input to the ML algorithms for the classification process to find out which player belongs to which class. As the output of our proposed methodology, the system produces player selection and position estimations. It also generates lineups for each ML algorithm and then the output is compared with the coach's ideal lineup and real match lineups in the evaluation phase.

3.1. Hit/it data

Hit/it Assistant is an electronic sport system that was designed and manufactured by Performa.nz Company [3]. The system can measure the control, technique, reaction time, speed, agility, coordination and surround control skills of a player. It supports the football players of every age and positively affects the motivations of them by making them train with workouts repeatedly. These repeated workouts are converted to fun for players with the games programmed in the system. With the help of the scoring system of Hit/it, a scalable competitive environment is obtained. Thus, it helps the players to reach their highest performances. The coaches do not teach the players how to use Hit/it because the aim is to make the players learn it themselves. As they use the

[3]Performa.nz (2021).Performance Measurement in Sports and Military [online]. Website http://performa.nz/ [accessed 20 May 2018].

device, they both learn what to do and how to react to the workouts to improve their skills. Figure 1 shows Hit/it Assistant.

The system is constructed with hit-sensitive panels. These panels have full-color LEDs to give feedback to the player. The panels can be fixed to each other as an arc or a circle. By default, the system is configured as a circle. The diameters of the circles may vary from 6.6 m to 12 m. For example, for older players, constructing a circle with a bigger diameter helps them simulate a real-sized football field. The system is controlled by a computer and it records hit count per workout, according to the workout scenario, reaction time in millisecond (ms), ball speed (estimation), and success rate for a single player.

Figure 1. Hit/it Assistant.

Hit/It workouts are prepared by the guidance of the coaches to improve the skills of the players in many ways. The coaches can make a player focus on certain training programs by simply choosing the appropriate workouts. The workout used in this study is "Sequence", which is played by simply hitting the panels in one direction. It aims to improve the fundamental skills and the muscle memory of the player. As the players rehearse, they begin hitting the next panel in a shorter time, which is called the reaction time measured in millisecond.

Hit/it collects data from the workouts of ALFA infrastructure players with Hit/it Assistant Controller Software [4] and stores in the SQLite database with 4 tables. The "Groups" table has information about the 9 age groups of players from U11 to U19 with columns (ID, Name). The "Players" table stores personal information, group, and the position of the player in columns [ID, FaceID, Name, Surname, Number, Groups, AccessLevel, Password, Mevki (position)]. The "Games" table stores the names of the games and player and date information in columns (ID, GameID, Name, StartTS, PlayerID). The "Results" table stores the results for all players in all workouts in columns (ID, GameRef, DataType, TS, ValueInt, ValueFloat, IsValid, Elapsed). This study uses the elapsed column of the "Results" table for the performances of the mentioned players from the workout they play.

Data used in this study is the average reaction times of U13 team for the sequence workout. With a simple SQL query on Hit/It database, the reaction times in millisecond from the elapsed column in the Results table are extracted. The average reaction times for the workout sequence for every player are shown in Table 1.

[4]Performa.nz (2021). Hit/it Assistant [online]. Website http://performa.nz/u1-hitit.html [accessed 30 May 2018].

Table 1. Average reaction times (ms) for sequence workout in Hit/it.

P#	Avg.1	Avg.2	Avg.3	Avg.4	Avg.5	Avg.6
1	1531.88	1296.21	1256.9	1411	1222.55	1116.63
2	1378.74	1341.97	1154.92	1714.41	1376.3	1243.89
3	1179.09	1332.65	1088.29	1272.25	1155.07	
4	1215.59	1281.58	1152.74	1049.19	1335.21	1164.3
5	1149.26	1181.36	1403.79	1137.03	1455.27	1163.28
6	1455.93	1235.3	1237.86	1235.44	1573.81	1171.38
7	1449.48	1573.34	1120.78	1204.85	1183.42	1129.63
8	1257.55	1110	947.714	1108.58	1211.09	
9		1048.73	1149.1	1393.6	1307.93	
10	1166.73	962.452	1052.82	1080.95	986.184	989.049
11	1170.53	1139.05	1262.05	1256.94	1130.82	1229.28
12	1430.09	1528.31	1342.28	1311.5	1078.92	992.071
13	1389.24	1146.81	1022.95	1180.83	1108.88	1271.53
14	1400.97	1128.25	1088.18	1067.33	1271.68	1343.65
15				1524.03		1480.77
16	1024.85	1039.28	1161.76	1256.4	1148.7	1244.24
17	1377	1176.94	1101.47	1126.14	1407.44	1113.53
18	1095.55	1462.46		1111.16	1101.98	
19	1248.65	1101.68	1061.98	1186.36	1200.52	
20	1007.66	1006.33	1072.76	1107.08	1191.77	1334.69
21	1069.98	1039.76	1062.61	1102.46	1068.23	1207.03

3.2. Coach evaluation data

The coaches observe their players during the training programs throughout the football season. They make an evaluation according to 18 criteria that are given in Table 2 after a long period of training. These evaluation criteria have linguistic variables “P” (poor), “F” (fair), “G” (good) and “VG” (very good) stated in [32] and their numerical values are given as 0.3, 0.5, 0.8, and 1.0, respectively. Numerical values used in this table are also used in [20] and [33] for other strategies.

Unlike that work, coaches in ALFA evaluated the players according to these criteria for this study. Every criterion is given an evaluation value for every player by the coach as shown in Table 3. These are characteristic and quantitative data about general features of a footballer.

The goalkeepers are excluded from the evaluation process because it is not possible to suggest a goalkeeper for any other positions in a match lineup. For this reason, player selection and team formation are done only for the other three positions: defenders (D), midfielders (M), forwards (F). The values in Table 3 are used as independent input data because they do not give any idea about the position of the player. These input values play an important role in calculating D-M-F position scores for each player, which is also used as training data for our ML algorithms in classification. By doing this, we do not add any match data as input in classification of the player positions.

Table 2. Evaluation criteria used to assess the skills of the footballers.

Criterion name	Explanation	Criterion name	Explanation
C1	Heading, jumping	C10	Create a goal scoring position
C2	Shoot	C11	Tackling
C3	Short passing	C12	Both feet
C4	Crossing	C13	Great stamina
C5	Ball control	C14	Height
C6	Dribbling	C15	Providing through (long) pass
C7	Finishing (composure)	C16	Technical ability
C8	Speed	C17	Create attacking opportunities
C9	Creativity	C18	Read the game

Table 3. Evaluation of U13 players of ALFA with given 18 criteria.

P#	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18
1	0.5	0.5	0.5	0.5	0.8	0.5	0.5	0.8	0.3	0.3	0.8	0.3	0.5	0.5	0.5	0.5	0.5	0.5
2	0.5	0.5	0.8	0.8	0.8	0.5	0.5	0.8	0.5	0.5	0.8	0.3	0.5	0.5	0.5	0.8	0.8	0.8
3	0.8	0.8	0.8	0.8	0.8	0.5	0.5	0.8	0.5	0.5	0.8	0.3	0.5	0.5	0.5	0.8	0.8	0.8
4	0.5	0.8	0.5	0.5	0.8	0.5	0.5	0.8	0.5	0.3	0.8	0.3	0.5	0.5	0.8	0.5	0.8	0.8
5	1	0.8	1	0.8	0.8	0.3	0.5	0.8	0.3	0.3	1	0.5	1	0.8	1	0.5	0.8	1
6	1	0.8	1	0.8	0.8	0.3	0.5	0.8	0.5	0.3	1	0.8	1	1	1	0.8	0.8	1
7	0.8	0.5	0.8	0.8	0.8	0.3	0.5	0.5	0.5	0.3	0.8	0.5	0.5	0.8	0.8	0.8	0.8	0.8
8	0.8	1	1	1	1	0.8	0.8	0.8	1	1	0.8	0.8	1	1	1	1	1	1
9	0.5	0.5	1	0.5	0.8	0.5	0.5	0.5	0.8	0.5	0.5	0.8	0.8	0.5	0.8	0.8	0.5	0.8
10	1	1	1	0.8	1	1	0.8	1	0.8	0.8	1	0.8	1	1	1	0.8	1	1
11	0.5	0.3	1	0.5	0.8	0.3	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.3	0.5	1	0.5	0.5
12	0.8	0.8	1	0.8	0.8	0.5	0.8	0.5	0.8	0.8	0.8	0.8	0.8	0.8	0.8	1	0.8	0.8
13	0.5	0.5	1	0.5	1	0.5	0.5	0.5	1	1	0.8	1	0.8	0.5	0.8	1	1	1
14	0.8	0.5	1	0.8	0.8	0.5	0.5	0.5	1	0.8	0.8	0.8	0.8	1	1	0.8	0.5	0.8
15	0.5	0.8	0.8	0.8	0.8	0.5	0.8	0.8	0.5	0.5	0.5	0.3	0.5	0.8	0.5	0.8	0.8	0.8
16	0.5	0.8	0.8	0.5	0.5	0.5	0.5	0.5	0.3	0.3	0.5	0.3	0.5	0.8	0.5	0.5	0.5	0.5
17	0.8	0.8	0.8	0.8	0.5	0.5	0.8	0.5	0.5	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8
18	0.5	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.5	0.8	0.5	0.8	0.8	1	0.8	0.5	1	0.5
19	0.8	1	0.8	0.8	0.5	0.8	0.8	1	0.5	0.8	0.8	0.3	0.8	1	0.8	0.8	0.8	0.8
20	0.5	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.5	0.8	0.5	0.5	0.8	0.8	1	0.8
21	0.8	1	0.8	0.8	0.8	1	0.8	1	0.5	1	0.5	0.5	0.8	1	0.8	0.8	1	0.5

3.3. Preprocessing

Neither the data in Hit/it database nor coach evaluation scores cannot be used as they are gathered. Hit/it data and coach evaluations must also be preprocessed to be ready for synthetic data generation. Then it can be used as input to ML algorithms for classification.

3.3.1. Preprocessing Hit/it data

To use Hit/it data for classification, the raw data extracted from the database is preprocessed. The reaction times generated in Hit/It workouts are monitored to observe the differences between the first and the last workout performances. If a player has a greater score at the beginning and a smaller one at the end, then it can be said that the reflexes of the player in improving and his/her reaction time is sharpening. To understand the improvement of the players, the first workout's average value is subtracted from the last workout's average value for each player. For most of the players, results are negative values, which is already a sign of improvement. To use these values in the data set, all differences are multiplied by –1. The obtained values are normalized from 0 to 1 to be compatible with coach evaluations. The normalization formula is given in Equation 1:

$$Normalized \;\; P_i = (v_i - Min(P_1 : P_{21}))/(Max(P_1 : P_{21}) - Min(P_1 : P_{21})), \tag{1}$$

where vi indicates the negated difference value for the ith player.

In Table 4, the steps of preprocessing Hit/it data are given. Six reaction times are used in dataset as 6 attributes for each instance and the normalized data is used as a seventh attribute in the final dataset as input. In this way, Hit/it is represented with 7 attributes in the dataset.

Table 4. Three preprocessing steps of Hit/it data.

P#	Difference	*(–1)	Normalized
1	–415.25	415.25	0.970234
2	–134.8496	134.8496	0.603721
3	–24.01753	24.01753	0.458852
4	–51.2973	51.2973	0.49451
5	14.013072	-14.0131	0.409142
6	–284.551	284.551	0.799397
7	–319.8563	319.8563	0.845545
8	–46.46753	46.46753	0.488197
9	259.20606	–259.206	0.088649
10	–177.6809	177.6809	0.659706
11	58.75	–58.75	0.350666
12	–438.0223	438.0223	1
13	–117.713	117.713	0.581322
14	–57.32353	57.32353	0.502387
15	–43.25611	43.25611	0.483999
16	219.38914	–219.389	0.140694
17	–263.4737	263.4737	0.771847
18	6.4273684	–6.42737	0.419058
19	–48.12484	48.12484	0.490363
20	327.02718	–327.027	0
21	137.05132	–137.051	0.248318

3.3.2. Preprocessing coach evaluation data

In football, the players must have different duties within the match. They try to do their best while interacting with their teammates. All football teams need a certain structure for forming their teams to apply certain strategies for each of their matches. There are four main positions for the football players in a team. These are goalkeeper (GK), defenders (D), midfielders (M), and forwards (F). Each team approximately has 20 players and 11 are chosen by the coaches as the starting lineup. The number of players in each of the D-M-F positions is based on the team formation selected for the game where only one GK is chosen. Coach evaluations given by the coaches are used to compute some quantitative values for the players to give some idea about their positions in the team. As an addition to the player evaluations, the criteria weights for D-M-F positions are also determined [20]. The linguistic values of importance weights are listed as "NI" (not important), "NS" (not so important), "N" (normal), "I" (important) and "VI" (very important) with the numerical values 1.0, 1.5, 2.0, 2.5, and 3.0, respectively. Table 5 shows the numerical and linguistic values for the importance weights of the evaluation criteria.

Table 5. The weight values of each criteria for the positions.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18
D	3	1.5	3	1.5	2.5	1.5	1.5	2	1.5	1.5	3	1.5	2	2.5	1.5	1.5	2.5	2.5
M	2	2.5	3	2.5	3	3	2	2.5	3	2.5	2.5	1.5	3	1.5	3	2.5	3	3
F	3	3	2.5	2.5	3	2.5	3	3	2.5	3	1.5	2.5	2.5	3	2.5	3	2.5	2.5

To obtain quantitative values about players for a particular position, evaluation values of the player are multiplied with the weight values of these 18 criteria one by one, and a sum is obtained for each player. For example, for Player 1, criteria values from C1 to C18 are multiplied by weight values of the criteria for defender position (First row of Table 3):

$$0.5*3 + 0.5*1.5 + 0.5*3 + \ldots + 0.5*2.5 + 0.5*2.5 = 19.6$$

The same is done for all players for all D-M-F positions. As a result, the values given in Table 6 are obtained which can be considered as derived variables of the dataset avoiding the usage of proxy variables for objectiveness. These derived variables do not give direct information about the position of the player. For example, it is not meaningful to sort the players according to their D, M or F values. These values are meaningful in the dataset with Hit/it performances to be used as input for ML algorithms.

3.4. Machine learning input data

U13 football team in ALFA has only 21 players to classify into three positions (D-M-F). Since the sample size is not enough to apply ML algorithms for classification, the need to generate synthetic data has arisen. Thus, after the generation of these synthetic data instances, our dataset is a composite one with real and synthetically generated instances together. Synthetic data generation is done as 10 synthetic instances for one player. This makes 210 synthetically generated instances and 21 real instances. So, the size of the dataset becomes 231 instances in total.

While playing in Hit/it, the players may perform well or bad according to their mood on that training day. For this reason, 6 average reaction times of real players are randomly generated within a range of +/-250 ms to the original values. For the synthetic generation of coach evaluations data, real players of different groups

Table 6. Calculated scores for D-M-F positions for U13 players of ALFA.

P#	D value	M value	F value
1	19.6	24	24.65
2	23.5	29.3	29.65
3	24.85	30.65	31.45
4	22.3	28.05	28.3
5	28.6	34.05	34.75
6	30.3	36.15	37.5
7	24.7	29.7	30.7
8	34	43.3	44.9
9	23.65	30.2	30.8
10	34.7	43.2	44.7
11	20.15	25.25	25.95
12	29.05	36.25	37.85
13	28.45	36.3	36.7
14	28.2	35	36.3
15	24.25	30.35	31.9
16	19.45	23.7	24.95
17	26.95	33.35	35.1
18	26.75	33.8	35.75
19	28.45	35.55	37.3
20	26.55	34.7	35.9
21	28.95	36.8	39.1

are used instead of randomly generating. In this way, these data is still golden standard. To apply player selection and team formation on preprocessed data, the following 28 features are added to the dataset as input:

- 18 features obtained from the evaluation of each player with the mentioned 18 criteria (Table 3).
- 3 features from scores calculated by preprocessing coach evaluations as given in Table 6.
- 6 features of reaction times from Hit/it as the raw data of workout 'sequence'.
- 1 feature from preprocessed Hit/it data as the normalized average reaction times in the workouts they performed (normalized column of Table 4).

Output data:

The position information of each instance (D-M-F) is the output of the proposed model as the class.

4. Machine Learning

There are seven ML algorithms applied for the solution of the problem proposed in the study. Considering the classification of supervised machine learning algorithms, there are seven different categories: artificial neural networks (ANN), support vector machines (SVM), decision trees (DT), logistic regression, Bayes theorem,

random forest (RF) and classification and regression tree (CART). In this study, one algorithm is used for each of the mentioned machine learning categories. To verify the reliability of these algorithms, each of them are also used with a smaller version of the same dataset (without Hit/it data) to show the necessity of Hit/it data.

Artificial neural networks (ANN) simulate the working mechanism of the human brain and perform basic functions such as learning, remembering, and generating new information [34]. An ANN consists of nodes, also called neurons, weighted connections between these neurons that can be adapted during the learning process of the network and an activation function that defines the output value of each node depending on its input values. Every neural network consists of different layers. The input layer receives information from external sources, such as attribute values of the corresponding data entry, the output layer produces the output of the network and hidden layers connect the input and the output layer with one another. The information passes through the nodes in a forward direction and the final outputs are computed [35]. Then for each output, error values are computed and propagated to each neuron in the backward direction. Later, the weights are updated to get better results. This forward-backward propagation continues until reaching minimal error values [36]. Multilayer perceptron (MLP) algorithm is used in the WEKA environment as an ANN strategy, which uses backpropagation. ANN can be considered compatible with the dataset used in this study because as stated in [37], ANNs can be trained on small datasets with minimal tuning and also large neural networks with hundreds of parameters per training observation are able to generalize well on small data sets.

Support vector machine (SVM) is a supervised machine learning algorithm that can be used for classification and regression [38]. It is based on dividing the groups in a plane for classification into two by drawing a boundary. The place where this boundary will be drawn is that the two groups must be the farthest place to their members. SVM is a boundary that best separates two classes (hyperplane/line). In WEKA, sequential minimal optimization (SMO) algorithm is used as the SVM algorithm. SMO decomposes the problem into a series of binary problems for standard SVM to be applied [39]. It splits the problem into smaller subproblems by using heuristics and this speeds up the training process.

Decision tree (DT) is one of the popular techniques which is used for classification and regression problems [40–42]. Constructing a DT begins with a set of cases, or data to be classified and creates a tree data structure that can be used to classify new cases by splitting the data into branches. The data given to feed a DT is used for both modeling and testing. Each internal node of a decision tree contains a test, the result of which is used to decide what branch to follow from that node [43]. In this study, logistic model tree (LMT) is chosen as a DT algorithm, which uses a combination of a tree structure and logistic regression models resulting in a single tree [44]. It basically consists of a standard decision tree structure with logistic regression functions at the leaves.

As a representative of the logistic regression algorithms, logistic is used in this study. Logistic in WEKA builds and uses a multinomial logistic regression model [45]. Logistic regression is a powerful classification that predicts probabilities directly. The linear regression model gets the input and predicts the output by estimating initial weight values for each input. The algorithm tries to minimize the cost function iteratively using the gradient descent algorithm [46].

Naive Bayes (NB) is based on Bayes theorem [47]. This algorithm is one of the most important classification algorithms because it is very easy to construct and does not need any complicated iterative parameter estimation schemes [48]. NB is also known to work well with smaller dataset as used in this study [49].

Random forest (RF) is a combination of tree predictors that builds many classification trees as a forest of

random decision trees [50]. RF builds multiple decision trees and merges them to get a more accurate and stable prediction. It can be used for both classification and regression problems. RF adds additional randomness to the model while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity that generally results in a better model[5].

CART (classification and regression trees) was introduced by [51] which is an algorithm used to build both classification and regression trees. It constructs the classification tree with the binary splitting of the attribute. It can be used for both continuous and discrete variables. In WEKA, the SimpleCART algorithm is used in this study for classification.

5. Experimental work and results

The chosen ML algorithms are used to make classification for the 231 instances of the given dataset. For each algorithm on WEKA environment, cross-validation (CV) is used to evaluate ML models. In this study, k-fold cross-validation procedure is applied with k=10. In order to observe the misclassified instances, a preprocessing step is needed. In this step, the AddID filter is applied to the data to add an ID attribute as the first attribute to the instances in the dataset. Next, the ML algorithms are executed in WEKA under FilteredClassifier. For the ID attribute not to affect the results, the filter option is given as "Remove" with the "first" parameter. Then the algorithms are executed by choosing the option "first-last" for the output predictions.

After the algorithms are executed the classification results are obtained. The percentages for correctly classified instances in the dataset for each ML algorithm are given in Table 7. According to the results, the best performing algorithm in classification is random forest, followed by MLP and LMT. In order to understand how much we gain from the proposed ML models, we applied same algorithms on the baseline model, which is the smaller version of our dataset containing D-M-F values only (without Hit/it performances). The comparative results for the classification of two different datasets (dataset without Hitit values vs. whole dataset with Hit/it performances) are given in Table 8. The amount of gain obtained in all algorithms by adding Hit/it data is remarkable. For this reason the comparison with the baseline model implies that the Hit/it attributes used in the dataset to train the ML algorithms are informative and essential for classification.

Table 7. ML algorithm performances for player selection phase.

Algorithm	Percentage (%)	Kappa statistic	Mean absolute error	Root mean squared error	Relative absolute error (%)	Root relative squared error (%)
MLP	92.6407	0.8896	0.0604	0.1953	13.5861	41.4318
SMO	89.1775	0.8377	0.2492	0.3168	56.0503	67.1925
LMT	90.4762	0.8571	0.0772	0.232	17.3589	49.2011
Logistic	85.7143	0.7857	0.0957	0.3057	21.5211	64.8405
Naïve Bayes	79.6537	0.6948	0.143	0.3338	32.1618	70.8022
RF	93.9394	0.9091	0.1207	0.1953	27.1573	41.4228
SimpleCART	79.2208	0.6883	0.1554	0.351	34.9629	74.448

[5]Yiu T. (2019). Understanding Random Forest [online]. Website: https://towardsdatascience.com/understanding-random-forest-58381e0602d2 [accessed 07.03.2021]

Table 8. Comparison results for baseline model-excluded vs. included Hit/it performance values.

Algorithm	Dataset without Hit/it values	Complete dataset
MLP	76.1905	92.6407
SMO	51.0823	89.1775
LMT	80.9524	90.4762
Logistic	70.1299	85.7143
Naïve Bayes	53.2468	79.6537
RF	81.8182	93.9394
SimpleCART	76.1905	79.2208

5.1. Analysis of player selection and team formation

The confusion matrices for the ML algorithms are given in Table 9 for these classification results of the training set. The best three algorithms (MLP, LMT, and RF) are marked with '*'. With this classification phase, the players are classified to D-M-F positions. For every algorithm, the classification results for the positions of the real players of U13 team are given in Table 10. As can be seen from the table, in four ML algorithms (MLP, LMT, Logistic, and RF) the members of U13 team were correctly classified. Naive Bayes has the worst performance with 4 incorrectly classified players. For all classification results, the best players of each class should be chosen to create the lineup for the team.

Table 9. Confusion matrices of the training set results for the ML algorithms.

MLP*				SMO				LMT*				Logistic			
D	M	F	classified as	D	M	F	classified as	D	M	F	classified as	D	M	F	classified as
72	0	5	D	74	0	3	D	70	1	6	D	65	3	9	D
2	73	2	M	2	71	4	M	1	71	5	M	3	68	6	M
6	2	69	F	10	6	61	F	5	4	68	F	4	8	65	F
Naive Bayes				RF*				SimpleCART							
D	M	F	classified as	D	M	F	classified as	D	M	F	classified as				
71	1	4	D	75	0	2	D	65	2	10	D				
13	52	12	M	1	73	3	M	1	67	9	M				
12	5	60	F	7	1	69	F	13	13	51	F				

Table 10. Player predictions for D-M-F positions of real U13 players.

	D	M	F
MLP	1-2-3-4-5-6-7	8-9-10-11-12-13-14	15-16-17-18-19-20-21
SMO	1-2-3-4-5-6-7	8-9-10-11-12-13-14-20	15-16-17-18-19-21
LMT	1-2-3-4-5-6-7	8-9-10-11-12-13-14	15-16-17-18-19-20-21
Logistic	1-2-3-4-5-6-7	8-9-10-11-12-13-14	15-16-17-18-19-20-21
Naïve Bayes	1-2-3-4-5-6-7-9-11-15-16	8-10-12-13-14	17-18-19-20-21
RF	1-2-3-4-5-6-7	8-9-10-11-12-13-14	15-16-17-18-19-20-21
SimpleCART	1-2-3-4-5-6-7	8-9-10-11-12-13-14-20	15-16-17-18-19-21

In ALFA, a lineup is composed of 4 defenders, 3 midfielders and 3 forwards. Among the players selected to these positions as the classification results of the algorithms, the best 4 players as defenders, the best 3 players as midfielders and the best 3 players as forwards are determined (using Table 6). Lineups generated from the classification results of the algorithms and the lineup suggested by the coach are given in Table 11.

The lineups generated from the classification results of the algorithms show that, all algorithms are able to find same combination of defenders, which differ from the coach's defenders with only one player. The combination of midfielders is the same with that of the coach's for every algorithm. For forwards, the algorithms are able to find two different combinations, where one of them is already the same as the coach's. The incorrectly classified players have no effect on the lineups of the algorithms because their D-M-F values do not affect the rankings of the best 4 defenders, 3 midfielders and 3 forwards (i.e. the results of naive Bayes algorithm).

Table 11. Predicted lineups for each ML algorithm compared with the golden standard's ideal lineup.

	D	M	F
Coach's lineup	6-5-3-2	8-10-13	21-19-18
MLP	6-5-3-7	8-10-13	21-19-20
SMO	6-5-3-7	8-10-13	21-19-18
LMT	6-5-3-7	8-10-13	21-19-20
Logistic	6-5-3-7	8-10-13	21-19-20
Naïve Bayes	6-5-3-7	8-10-13	21-19-20
RF	6-5-3-7	8-10-13	21-19-20
SimpleCART	6-5-3-7	8-10-13	21-19-18

5.2. The evaluation of team formation

The evaluation of the team formation is done in two ways. First, the generated lineups with the ML algorithms are compared with the ideal lineup of the coach. Second, the lineups generated by the ML algorithms are compared with real match lineups for 20 consecutive matches played in 2019–2020 football season. These lineups work as unseen test data in the study which is never used as training data.

To make a comparison with a proper statistical analysis method, two different techniques are used. One of them is the Jaccard similarity, which depends on the principle of calculating the similarity between sets by looking at the number of common elements of two data sets [52]. The Jaccard similarity coefficient for the comparison of coach's ideal lineup and lineups of SMO and SimpleCART algorithms is given as 0.8182 (9 common elements). The Jaccard similarity coefficient for the comparison of coach's ideal lineup and lineups of other five algorithms is 0.6667 (8 common elements). In this study, it is not sufficient to evaluate only the number of common elements between the best lineups suggested by ML algorithms and the lineups of unseen match data. This is because it is important that a player not only ranks in the lineup, but also in what position he plays in the team. Moreover, for all lineups having the same number of common players Jaccard similarity would produce the same similarity coefficient just because the number of common elements is equal, which is not acceptable.

The other technique is Pearson correlation, which is also known as the product-moment correlation coefficient (PMCC). PMCC is a value between –1 and 1 that indicates if two variables are linearly related. It is a test that measures the association between the variables [53]. The lineup comparisons are done with Pearson

correlation using Wessa online software[6].

The comparison results of ML algorithms with that of the coach's ideal lineup are given in Table 12 for Pearson correlation. In addition to the correlation coefficient value, some other analysis results are also given in the table. Determination value, which is the square of the Pearson correlation coefficient, is also important because it explains how differences in one variable can be explained by a difference in a second variable. The t-test is a parametric test technique examining the difference between the means of two samples [54]. Two-sided p, also called confidence level, tests whether a sample is greater than or less than a certain range of values. It is used for testing statistical significance, where 1-sided p-value indicates that the critical area of a distribution is 1-sided so that it is either greater than or less than a certain value, but not both. Confidence level value for p-value is 0.005 for 2-sided p and 0.0025 for 1-sided p. P values obtained from this study are less than the confidence level values, which show the reliability of the ML algorithms used. The results of Pearson correlation show that the ML algorithms SMO and SimpleCART have the closest lineup to the coach's.

Table 12. Comparison of predicted lineups for each ML algorithm with Pearson correlation.

	MLP	SMO	LMT	Logistics	Naive Bayes	RF	SimpleCART
Correlation	0.9716	0.9748	0.9716	0.9716	0.9716	0.9716	0.9748
Determination	0.9440	0.9502	0.9440	0.9440	0.9440	0.9440	0.9502
t-test	11.6222	12.3627	11.6222	11.6222	11.6222	11.6222	12.3627
p-value (2-sided)	2.7346e-06	1.7080e-06	2.7346e-06	2.7346e-06	2.7346e-06	2.7346e-06	1.7080e-06
p-value (1-sided)	1.3673e-06	8.5403e-07	1.3673e-06	1.3673e-06	1.3673e-06	1.3673e-06	8.5403e-07

The comparison results of ML algorithms with the real match lineups are given in Table 13. Lineups given in the table belong to 2019–2020 season for Altınordu U14 team because the U13 team of 2018–2019 season should be evaluated with the next season's performances after they were trained with Hit/it. The lineups data can be accessed from Turkish Football Federation web site for all teams throughout seasons [7]. The lineup data of each match is compared with the lineups generated by all seven ML algorithms. For each algorithm, an average value of 20 comparisons is calculated in the last row of the table. When the results are examined, MLP, LMT, logistics, naive Bayes and RF are detected as the better performing algorithms with the percentage of 89.36%. On the other hand, SMO and SimpleCART algorithms also performed close to the other five with a performance of about 88.89%.

6. Conclusion

In this study, the data obtained from the training performances of players of the U13 team in the Altınordu Football Academy was used to make player selection and lineup prediction as a suggestion to the coach. The workout in Hit/it tool was played by the infrastructure players continuously during the season and their improvement was observed. The main purpose of Hit/it is to help improving young athletes before they reach the professional leagues because among the infrastructure players, the best performing ones are chosen to be sent to other football teams in Turkey.

[6]Wessa.Net Wessa: Free Statistics and Forecasting Software [online]. Website https://www.wessa.net/rwasp_correlation.wasp [accessed 08.03.2021].

[7]Turkish Football Federation (2021). Futbol Bilgi Bankası [online]. Website https://tff.org/default.aspx?pageID=322 [accessed 24 June 2020].

Table 13. Match lineup comparisons for U13 2019–2020 season for each ML algorithm.

Date	Lineup	MLP	SMO	LMT	Logistics	NaiveBayes	RF	SimpleCART
8.9.2019	1-3-6-5-7-13-8-15-21-19	0.8801	0.8703	0.8801	0.8801	0.8801	0.8801	0.8703
15.9.2019	2-4-6-5-10-13-8-18-21-19	0.9041	0.8992	0.9041	0.9041	0.9041	0.9041	0.8992
22.9.2019	2-4-6-5-10-13-8-18-21-19	0.9041	0.8992	0.9041	0.9041	0.9041	0.9041	0.8992
29.9.2019	1-4-6-5-10-13-8-18-21-17	0.8869	0.8894	0.8869	0.8869	0.8869	0.8869	0.8894
6.10.2019	2-6-10-8-5-4-13-21-18-19	0.8648	0.8583	0.8648	0.8648	0.8648	0.8648	0.8583
13.10.2019	1-3-6-5-10-13-8-15-21-19	0.8657	0.8563	0.8657	0.8657	0.8657	0.8657	0.8563
20.10.2019	1-3-5-7-12-9-8-18-19-17	0.9015	0.9008	0.9015	0.9015	0.9015	0.9015	0.9008
27.10.2019	2-3-6-5-10-13-8-18-21-17	0.8938	0.8965	0.8938	0.8938	0.8938	0.8938	0.8965
24.11.2019	2-4-6-5-10-13-8-15-18-19	0.8782	0.8630	0.8782	0.8782	0.8782	0.8782	0.8630
1.12.2019	2-4-6-5-10-13-8-18-21-19	0.9041	0.8992	0.9041	0.9041	0.9041	0.9041	0.8992
15.12.2019	1-4-6-5-10-13-8-18-21-17	0.8869	0.8894	0.8869	0.8869	0.8869	0.8869	0.8894
22.12.2019	1-4-6-5-10-13-8-18-21-17	0.8869	0.8894	0.8869	0.8869	0.8869	0.8869	0.8894
29.12.2019	1-4-6-5-10-13-8-18-19-21	0.9023	0.8877	0.9023	0.9023	0.9023	0.9023	0.8877
5.1.2020	2-3-5-4-12-8-14-17-18-19	0.9216	0.9140	0.9216	0.9216	0.9216	0.9216	0.9140
8.1.2020	2-4-5-6-13-8-10-21-18-17	0.9192	0.9228	0.9192	0.9192	0.9192	0.9192	0.9228
12.1.2020	6-1-5-4-3-13-10-8-18-19	0.7603	0.7348	0.7603	0.7603	0.7603	0.7603	0.7348
9.2.2020	2-4-5-6-13-8-10-21-18-19	0.9286	0.9247	0.9286	0.9286	0.9286	0.9286	0.9247
16.2.2020	4-3-2-6-13-8-14-20-18-19	0.9522	0.9507	0.9522	0.9522	0.9522	0.9522	0.9507
1.3.2020	3-4-2-6-13-11-12-15-18-17	0.9042	0.9019	0.9042	0.9042	0.9042	0.9042	0.9019
8.3.2020	5-4-2-6-13-10-8-21-18-17	0.9268	0.9308	0.9268	0.9268	0.9268	0.9268	0.9308
	Avgs	0.8936	0.8889	0.8936	0.8936	0.8936	0.8936	0.8889

The training data provides a more consistent set of data than the match data about the players because a player may perform poorly in a match, suffer a disability, or may not be preferred by his coach in the top 11. This should not affect the player's overall performance, which should be extended throughout the year. On the other hand, the training data of the team was collected for an entire season and by monitoring this data, it could be observed how the player improves himself by performing the same exercises during consequent training. This observation gained a quantitative value, especially when working with a data recording tool such as Hit/it for training. This data was combined with the coach evaluations for the players, that were converted into a compatible format to feed ML algorithms in this study.

The real data used for this study was small because it was the data of the players of a single football team (excluding three goalkeepers), which was a small dataset in size. Indeed, the proposed algorithms could have been applied to a bigger version of the dataset with the data of other Ux teams, which would extend the dataset size. However, the infrastructure teams are determined by the age of the athletes and they cannot be merged for team formation process. U13 team of ALFA was taken as a case study for this paper and in order that this situation did not create a disadvantage for our study, synthetic data generation was done. 10 synthetic data was generated for one real (231 instances in total), based on the data of U13 team only.

The ML algorithms were used for position classifications of the players and the best lineups of the algorithms are generated after this classification. Among these algorithms, SMO and SimpleCART produced the closest results to the coach's lineup. The other five had worse classification performance, which was also remarkably reliable for team formation. RF was also one of the most successful algorithms in terms of

classification. The fact that it was the best algorithm in team-building made RF the most preferred algorithm in this study. For the performance percentages of the algorithms that produced close results, the players they misclassified may vary. Therefore, differences could be observed in the lineups after these classifications. LMT and SMO, for example, showed close performances in classifying players by position, and also the similarity percentages of the lineups of these algorithms to the ideal lineup of the coach were close (97.16% and 97.48%, respectively).

In the second step of the evaluation for team formation, it was seen that comparing not only with one ideal lineup of the coach but also with real match lineups gave reliable results. The unseen match data came from the lineups of 20 consecutive matches played in 2019–2020 football season. Among the ML algorithms applied, MLP, LMT, logistics, naive Bayes and RF performed the best for unseen real match data with 89.36%.

The results obtained in this study showed that ML algorithms were reliable for player classification problems and if it was supported with other input data like coach evaluations and quantitative data to represent player skills and capabilities for each position, the classification results could also be used for other purposes like lineup suggestions to the coaches. Since this study resulted in similar results to that of the coach's lineup with the mentioned components of the dataset, it is not necessary to use match data for position predictions and team formation.

This was the first study in which Hit/it data was used for classification. As a continuation of this study, the player classification and lineup suggestion for the coach will be integrated with Hit/It software to be used every week before the matches.

Acknowledgments

This research was supported by Performan.nz under the author's academic consultancy. We thank our colleague Hakan Şumnulu from Performa.nz, Namet Ateş, the training coordinator of the Altınordu Football Academy and the technical staff of the U13 team who provided us data, insight, and expertise that greatly assisted the research, although they may not agree with all of the conclusion of this paper.

References

[1] Arnason A, Sigurdsson SB, Gudmundsson A, Holme I, Engebretsen L et al. Physical fitness, injuries, and team performance in soccer. Medicine & Science in Sports & Exercise 2004; 36: 278-285. doi: 10.1249/01.MSS.0000113478.92945.CA

[2] Boon BH, Sierksma G. Team formation: matching quality supply and quality demand. European Journal of Operational Research 2003; 148: 277-292. doi: 10.1016/S0377-2217(02)00684-7

[3] Frank E, Hall MA, Witten IH. The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques". 4th ed. Burlington, MA, USA: Morgan Kaufmann Publishers Inc., 2016.

[4] Zahradnik D, Korvas P. The Introduction into Sports Training. Masaryk University Press, Brno, Czech Republic, 2012.

[5] Jürgen P, Arnold B. Application of neural networks to analyze performance in sports. In: 8th Annual Congress of the European College of Sport Science ECCS; Salzburg, Austria; 2003. p. 342.

[6] Baca A, Kornfeind P. Rapid feedback systems for elite sports training. Pervasive Computing IEEE 2006; 5(4): 70-76. doi: 10.1109/MPRV.2006.82

[7] Novatchkov H, Baca A. Machine learning methods for the automatic evaluation of exercises on sensor-equipped weight training machines. Procedia Engineering 2012; 34: 562-567. doi: 10.1016/j.proeng.2012.04.096

[8] Owusu G. AI and computer-based methods in performance evaluation of sporting feats: an overview. Artificial Intelligence Review 2007; 27 (1): 57-70. doi: 10.1007/s10462-008-9068-3

[9] Pernek I, Hummel KA, Kokol P. Exercise repetition detection for resistance training based on smartphones. Personal Ubiquitous Computing 2013; 17 (4): 771-782. doi: 10.1007/s00779-012-0626-y

[10] Vales-Alonso J, López-Matencio P, Gonzalez-Castaño FJ, Navarro-Hellín H, Baños-Guirao PJ et al. Ambient intelligence systems for personalized sport training. Sensors 2010; 10 (3): 2359-2385. doi: 10.3390/s100302359

[11] Fisher I, Rauter S, Yang X-S, Ljubic K, Fisher Jr I. Planning the sports training sessions with the bat algorithm. Neurocomputing 2015; 149: 993-1002. doi: 10.1016/j.neucom.2014.07.034

[12] Severini TA. Analytic methods in sports: using mathematics and statistics to understand data from baseball, football, basketball, and other sports. Boca Raton, FL, USA: CRC Press, 2014.

[13] Liu F, Shi Y, Najjar L. Application of design of experiment method for sports results prediction. Procedia Computer Science 2017; 122: 720-726. doi: 10.1016/j.procs.2017.11.429

[14] Leung CK, Joseph KW. Sports data mining: predicting results for the college football games. Procedia Computer Science 2014; 35: 710-719. doi: 10.1016/j.procs.2014.08.153

[15] Bunker RP, Thabtah F. A machine learning framework for sports results prediction. Applied Computing and Informatics 2019; 15: 27-33. doi: 10.1016/j.aci.2017.09.005

[16] Park YJ, Kim HS, Kim D, Lee H, Kim SB et al. A deep learning-based sports player evaluation model based on game statistics and news articles. Knowledge-Based Systems 2017; 138: 15-26. doi: 10.1016/j.knosys.2017.09.028

[17] Yanpeng Z. Hybrid kernel extreme learning machine for evaluation of athletes' competitive ability based on particle swarm optimization. Computers and Electrical Engineering 2019; 73: 23-31. doi: 10.1016/j.compeleceng.2018.10.017

[18] Novatchkov H, Baca A. Fuzzy logic in sports: a review and an illustrative case study in the field of strength training. International Journal of Computing Applications 2013; 71 (6): 8-14. doi: 10.5120/12360-8675

[19] Ofoghi B, Zeleznikow J, MacMahon C, Dwyer D. Supporting athlete selection and strategic planning in track cycling omnium: a statistical machine learning approach. Information Sciences 2013; 233: 200-213. doi: 10.1016/j.ins.2012.12.050

[20] Tavana M, Azizi F, Azizi F, Behzadian M. A fuzzy inference system with application to player selection and team formation in multi-player sports. Sport Management Review 2013; 16: 97-110. doi: 10.1016/j.smr.2012.06.002

[21] Pehlivan NY, Unal Y, Kahraman C. Player selection for a National Football Team using fuzzy AHP and fuzzy TOPSIS. Journal Of Multiple-Valued Logic And Soft Computing 2019; 32 (5-6): 369-405.

[22] Braha D. Partitioning tasks to product development teams. In: International Design Engineering Technical Conferences of American Society of Mechanical Engineers (DETC'02 ASME); Montreal, Canada; 2002.

[23] Durmusoglu M, Kulak O. A methodology for the design of office cells using axiomatic design principles. Omega 2008; 36: 633-652. doi: 10.1016/j.omega.2005.10.007

[24] Schumaker RP, Solieman OK, Chen H. Predictive modeling for sports and gaming. Sports data mining, Springer 2010; 55-63. doi: 10.1007/978-1-4419-6730-5_6

[25] Seif El-Nasr M, Drachen A, Canossa A. Game Analytics: Maximizing the Value of Player Data. London, UK: Springer, 2013.

[26] Qader MA, Zaidan BB, Zaidan AA, Ali SK, Kamaluddin MA et al. A methodology for football players selection problem based on multi-measurements criteria analysis. Measurement 2017; 111: 38-50. doi: 10.1016/j.measurement.2017.07.024

[27] Eberhart R, Kennedy J. A new optimizer using particle swarm theory. In: 6th International Symposium on Micro Machine and Human Science; Nagoya, Japan; 1995. pp. 39-43.

[28] Storn R, Price K. Differential Evolution: A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. Technical Report TR-95-012. Berkeley, CA, USA: International Computer Science Institute, 1995.

[29] Karaboga D, Basturk B. Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems. In: Melin P, Castillo O, Aguilar LT, Kacprzyk J, Pedrycz W (editors). Foundations of Fuzzy Logic and Soft Computing. IFSA 2007. Lecture Notes in Computer Science, Vol 4529. Berlin, Germany: Springer, 2007. doi: 10.1007/978-3-540-72950-1_77

[30] Agarwalla P, Mukhopadhyay S. Efficient player selection strategy based diversified particle swarm optimization algorithm for global optimization. Information Sciences 2017; 397-398: 69-90. doi: 10.1016/j.ins.2017.02.027

[31] Agarwalla P, Mukhopadhyay S. Hybrid advanced player selection strategy based population search for global optimization. Expert Systems with Applications 2020; 139: 112825. doi: 10.1016/j.eswa.2019.112825

[32] Karsak E. A fuzzy multiple-objective programming approach for personnel selection. In: International Conference on Systems, Man, and Cybernetics; Nashville, TN, USA; 2000.

[33] Maanijou R, Mirroshandel SA. Introducing an expert system for prediction of soccer player ranking using ensemble learning. Neural Computing and Applications 2019; 31: 9157-9174. doi: 10.1007/s00521-019-04036-9

[34] Caudill M. Neural network primer: part I. AI Expert 1989; 2 (12): 46-52.

[35] Karray FO, Silva CD. Soft Computing and Intelligent Systems Design: Theory, Tools, and Applications. New York, NY, USA: Addison Wesley Pearson Press, 2004.

[36] Rojas R. Neural Networks: A Systematic Introduction. Berlin, Germany: Springer-Verlag, 1996.

[37] Olson M, Wyner A, Berk R. Modern neural networks generalize on small data sets. Advances in Neural Information Processing Systems 2018; 31: 3619-3628.

[38] Cortes C, Vapnik V. Support vector networks. Machine Learning 1995; 20: 273-297. doi: 10.1023/A:1022627411411

[39] Wang Z, Xue X. Multi-class support vector machine. In: Ma Y, Guo G (editors). Support Vector Machines Applications. Springer, Cham, Switzerland: Springer International Publishing, 2014, pp. 23-24. doi: 10.1007/978-3-319-02300-7_2

[40] Quinlan JR. Induction of decision trees. Machine Learning 1986; 1 (1): 81-106. doi: 10.1023/A:1022643204877

[41] Chen YL, Hsu CL, Chou SC. Constructing a multi-valued and multi-labeled decision tree. Expert Systems with Applications 2003; 25(2): 199-209. doi: 10.1016/S0957-4174(03)00047-2

[42] Chou PA. Optimal partitioning for classification and regression trees. IEEE Transactions on Pattern Analysis and Machine Intelligence 1991; 13 (4): 340-354. doi: 10.1109/34.88569

[43] Salzberg SL. Book Review: C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kauffman Publishers, Inc., 1993. Machine Learning 1994 (16): 235-240. doi: 10.1023/A:1022645310020

[44] Landwehr N, Hall M, Frank E. Logistic model trees. Machine Learning 2005; 59 (161). doi: 10.1007/s10994-005-0466-3

[45] Greene WH. Econometric Analysis. 7th ed. Boston, MA, USA: Pearson Education, 2012.

[46] Snyman J. Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms. In: Pardalos PM, Hearn DW (editors). Applied Optimization, Vol. 97. New York, NY, USA: Springer, 2005.

[47] Bayes M, Price M. An essay towards solving a problem in the doctrine of chances. By the late rev. Mr. Bayes, F. R. S. Communicated by Mr. Price, in a letter to John Canton, A. M. F. R. S. Philosophical Transactions 1763; 53: 370-418.

[48] Wu X, Kumar V, Quinlan JS, Ghosh J, Yang Q et al. Top 10 algorithms in data mining. Knowledge and Information Systems 2008; 14: 1-37. doi: 10.1007/s10115-007-0114-2

[49] MacAllister A. Investigating the use of Bayesian networks for small dataset problems. PhD, Iowa State University, ames, IA, USA, 2018.

[50] Ho TK. Random decision forests. In: Proceedings of the 3rd International Conference on Document Analysis and Recognition; Montreal, QC, Canada; 1995. pp. 278-282.

[51] Breiman L, Friedman JH, Olshen RA, Stone CJ. Classification and regression trees. Boca Raton, FL, USA: CRC Press, 1984.

[52] Jaccard P. Étude comparative de la distribution florale dans une portion des Alpes et des Jura. Bulletin de la Société Vaudoise des Sciences Naturelles 1901; 37: 547-579 (in French). doi: 10.5169/seals-266450

[53] Pearson K. II. Mathematical contributions to the theory of evolution. II. Skew variation in homogeneous material. Proceedings of the Royal Society of London 1985; 57: 340-346. doi: 10.1098/rspl.1894.0147

[54] Student. The probable error of a mean. Biometrika 1908; 6 (1): 1-25. doi: 10.2307/2331554