IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 7, 2021, accepted October 29, 2021, date of publication November 2, 2021, date of current version November 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3124931

Empirical Comparisons for Combining Balancing and Feature Selection Strategies for Characterizing Football Players Using FIFA Video Game System

MUSTAFA A. AL-ASADI AND SAKIR TASDEMİR
Faculty of Technology, Department of Computer Engineering, Selçuk University, 42130 Konya, Turkey
Corresponding author: Mustafa A. Al-Asadi (masadi@lisansustu.selcuk.edu.tr)

ABSTRACT The process of modelling individual player performance using machine learning is a mature task in sports analytics. The most significant challenges in machine learning include class imbalance and high dimensionality problems. We conducted a comprehensive literature review and observed that both the issues have been studied independently. We found that feature selection addresses the dimensionality reduction problem by determining a subset of relevant features, while data sampling seeks to make the data more balanced by adding or removing instances. We also found out that efforts have been taken for studying the effect of the joint use of feature selection and balancing techniques. However, the prioritization of the feature selection and sampling is still difficult, and the relationship between them remains unclear. This paper presents a large-scale comparison of characterizing football players into nine positions by using FIFA video game data, whereas most of the previous studies in this field have focused on characterizing players into only three classes according to their positions. The proposed methodology for the study consists of three main steps. In the first step, the sampling technique is applied to deal with class imbalance, while the second step encompasses the feature selection technique, which deals with the high dimensionality problem. The third step combines feature selection and data sampling to deal with both the issues. We made the comparisons based on nine feature selection algorithms and three balancing techniques, and then we evaluated their performance using the random forest classifier. We found that 1) feature selection techniques did not improve the accuracy of the baseline model, 2) balancing techniques improved the accuracy compared to the baseline, and 3) the results showed superiority of the proposed methodology, involving the joint application of resampling and feature selection with data balanced by the random oversampling (ROS) method and synthetic minority oversampling technique (SMOTE), compared to the results obtained only through the use of a single technique and from the original imbalanced training set. Overall, the proposed methodology improved prediction accuracy compared to the baseline model. Moreover, the methodology provided a significant decrease in the number of features, from 29 to 10 features on average.

INDEX TERMS Class imbalance, data mining, data sampling, feature selection, FIFA video game, player characterizing.

I. INTRODUCTION

Football is regarded as the most popular sport in the world in the number of both spectators and players [1]. The popularity of football has increased in the last few years, making it an essential contributor to the global economy [2]. In fact, the revenue for European football clubs alone for 2017 was rated at $27 bn [3]. In football, creating an optimal lineup of players capable of winning over another list is a significant challenge since player positions require different skills [4]. Furthermore, there is no formula or scientific equation to identify a player's preferred position in the team. The coaches generally

The associate editor coordinating the review of this manuscript and approving it for publication was Emre Koyuncu.

do the assignment using their experiences and observations about the players [5], which exposes selection of players to many biases.

For the past two decades, machine learning has become an essential methodology for transforming football statistics into useful information to help teams and guide coaches in analyzing opponents and making better decisions in real time by using sensor-generated data. These data comprise videos from cameras to all types of physical measurements and human monitoring. However, research in the field of football analytics with machine learning techniques is limited. The main reason for this is the lack of a large-scale dataset for players, since collecting such rich information about players might be costly, making sensed data limited to teams with high purchasing power [5]. In football analytics, video games such as FIFA and Football Manager (FM) are considered as other sources of data. Since 2014, researchers and clubs have used video games as alternative sources for data. Shin and Robert used the FIFA video game data to predict match results, and they found that this data can be used in machine learning projects to make predictions with accurate results [6].

The website of SoFIFA classifies simulated players in the FIFA video game series into 14 positions and provides the player ratings on 29 different skills, where each skill is evaluated on a 0-to-100 scale. All of the previous studies, on the other hand, combined these 14 positions into only three classes according to playing roles: defense, midfield, and forward line. Combining these positions stems from two main reasons. First, most players can play in multiple positions, and second, the multi-class distribution of players in the SoFIFA data is highly skewed, leading to a problem in classification accuracy due to class imbalance and high dimensionality in the data. The instability in the classes leads to the problem of class imbalance, which may decrease the prediction performance and extend the training period.

The class imbalance problem is considered the most significant issue in data mining. When one of the two classes has more samples than the other, an imbalance problem occurs. In such a situation, most classifiers are biased towards the major classes and hence show abysmal classification rates on minor classes. Although the minority samples rarely occur, they are crucial to some areas, such as detecting fraud in banking operations, finding network intrusions, and diagnosing cancer [7].

Feature selection is a process of choosing a subset of relevant features so that the quality of prediction models can be maintained or improved. Moreover, feature selection is one of the essential data preprocessing steps in data mining. Data sampling seeks to make the data more balanced by adding or removing instances [8].

During the literature review, we identified that many data mining techniques are helpful but not sufficient. However, some papers suggest that applying two or more techniques may give better solutions for the class imbalance problem [9]–[12]. This paper presents a large-scale comparison for characterizing football players into nine positions using unbalanced data, namely FIFA video game data. Thus, this paper also explores the role of data mining techniques to improve the performance of machine learning algorithms via treating both high dimensionality and class imbalance problems in the used dataset. Therefore, we propose an approach combining feature selection and data sampling to solve the problems that exist in the above-mentioned data. The proposed methodology for the study comprises three main steps. The first step consists of applying the sampling technique to deal with class imbalance, while the second step covers the feature selection technique, which focuses on the high dimensionality problem. The third step combines feature selection and data sampling to resolve both the issues. We made the comparisons based on nine feature selection algorithms and three balancing techniques. Moreover, their performances were evaluated using the random forest (RF) classifier. Sections 2, 3, 4, 5, 6, 7, and 8 provide a literature review, the steps of the methodology of the study, the FIFA dataset used in the study, the elements of the study, the evaluation metrics, the research design, and the analysis of results, respectively. The last section of the paper encompasses the discussion and conclusions.

II. LITERATURE REVIEW

In this section, we briefly review the literature on characterizing football players' positions and the studies examining the influence of combining resampling and feature selection techniques to manage class imbalance.

A. CHARACTERIZING FOOTBALL PLAYERS' POSITIONS

Characterizing football players for their particular positions according to their skills and specific metrics has attracted coaches' and data scientists' attention. Therefore, most European clubs have enlisted data scientists or algorithms specialists to help them with this requirement. However, the issue of grouping and selecting players based on their individual skills and data using machine learning methods is an open field that has not seen much published research [13], [14]. To the best of our knowledge, ours is the first approach that characterizes players in nine positions in a match according to their skills by using a supervised learning approach.

On the other hand, most of the studies in this field focused on characterizing players into three categories according to their positions: defense, midfield, and forward line. We strongly believe that building a model according to nine positions using an extended set could reveal more insights about the characteristics of football players required in each position. Table 1 summarizes the most critical recent research about the machine learning-based characterization of football players' positions; the comparison has been made in terms of classifier, data type, highest accuracy, and number of instances, features, and positions classified.

TABLE 1. Summary of some of the most-critical recent research studies on characterizing football players' positions.

Ref.	Data base	Classifier	Highest Accuracy	N. of Instances	N. of Features	N. of Positions classified	Assessment
[15]	Real data (Iranian club)	Fuzzy rule-based classifier	z = 75.9%	20	18	4	The classification has been conducted for only four positions.
[16]	EA Sports' FIFA video game	C4.5, Backprop -ANN, ANN-RBF	94.12%	206	28	4	The classification has been conducted for only four positions.
[17]	EA Sports' FIFA video game	Neural network, Random Forests and Logistic regression.	79.1%	1700	40	3	The classification has been conducted for only three positions.
[14]	EA Sports' FIFA video game	Principal components analysis in conjunction with a model-based Gaussian clustering	95%	7705	40	4	The classification has been conducted for only four positions.
[18]	EA Sports' FIFA video game	Random Forests and Sequential Minimal Optimization (SMO)	81.5%	120	34	4	The classification has been conducted for only four positions.

B. STUDY OF THE INFLUENCE OF COMBINED RESAMPLING AND FEATURE SELECTION TECHNIQUE

As mentioned in the Introduction section, the imbalance problem occurs when one of the two classes has more samples than the other. In such a situation, most classifiers are biased towards the major classes and give poor classification rates for minor classes. The methods for the classification of the imbalanced dataset are divided into three main categories: the algorithmic approach, data-preprocessing approach (resampling), and feature selection approach. Each of these techniques has pros and cons [7], [19], [20]. Moreover, by means of the literature review, we observed that studies have examined the effects of combining resampling and feature selection to address the class imbalance. For combining the two techniques, researchers have investigated four different approaches: Approach 1 – feature selection and modelling based on original data; Approach 2 – feature selection based on original data and modelling based on sampled data; Approach 3 – feature selection based on sampled data and modelling based on original data; and Approach 4 feature selection and modelling based on sampled data. The supervised feature selection methods are strongly affected by the distribution of data (the imbalanced problem), such that it is natural to select the features on the sampled data and then have them modelled on the sampled data. We, therefore, present a comparative study using only the fourth approach and our results also confirm the abovementioned viewpoints. In other words, we assume that Approach 1, Approach 2, and Approach 3 were redundant comparative studies. We believe that the experiments conducted in our study will guide future practices in the categorization of class-imbalanced data. Table 2 summarizes the most critical recent research directed towards studying the influence of combining resampling and feature selection to tackle the class imbalance; the comparison has been made in terms of classifier, data type, approaches, and type of feature selection and data sampling algorithm.

III. STEPS OF METHODOLOGY

Before detailing the proposed methodology, we have reviewed the studies examining the influences of combining resampling and feature selection to solve class imbalance, shed light on the ideal method in modelling, and discussed the techniques used. The proposed methodology consists of three main steps. The first step encompasses applying the sampling technique to deal with class imbalance, while the second step covers the feature selection technique, which deals with the high dimensionality problem. The third step combines feature selection and data sampling to deal with both the issues. In addition to the main steps, two more steps were added to deal with other challenges in the dataset.

1) NORMALIZE AND DOUBLE DATA

The original dataset consisted of 17,981 players; however, some players had multiple positions. We, therefore, doubled the number of players to 27,251. The FIFA dataset contained a column that showed the player's preferred position. Afterwards, nine out of 14 play positions were assigned. Later, we performed data normalization on all the dataset features except for the attribute having the preferred position, to ensure consistency. Thus, finally the value of each feature ranges from 0 to 1. Moreover, some of the features have a '+/−' sign. For this reason, we did some calculations instead of keeping them as a string.

2) DATA SPLITTING AND EVALUATION MODELS

After cleaning the dataset, 80% of the data was randomly allocated to train the classifier, while the remaining 20% was used for testing. In classification problems, the simplest way to evaluate an algorithm's performance is to use different

TABLE 2. Summary of some of the most-critical recent research studies on the influence of combined resampling and feature selection technique on class imbalance.

Refs.	Application	Dataset	Approaches	Model	Highlights	Assessment
[21]	Music (Identifying the melodic track of a given MIDI file)	CL200, JZ200, KR200, CLA, JAZ and KAR	1- Feature Selection after Data Sampling (B1). 2- Feature Selection before Data Sampling (B2).	1-NN, SVM	- The benefits associated to resample training data are greater than those related to the use of feature selection. - The application of feature reduction methods on imbalanced data has low effectiveness'. - The B1 approach performs better than the B2 approach.	- Only one method for feature selection was considered: filter and the other method was PCA. - Only two methods for sampling were considered: SMOTE and RUS. - After sampling data in approach B1, the study did not focus on modeling data based on original or sampled data.
[22]	Different application fields.	Three cancer gene expression datasets and four datasets from the UCI ML Repository (Internet Advertisements dataset, Musk, Satimage-4 and Opt digits.	1- Feature selection based on sampled data, and modeling based on original data (A1). 2- Feature selection based on sampled data, and modeling based on sampled data (A2). 3- Feature selection based on original data, and modeling based on sampled data (A3).	NB, MLP, 5-NN, SVM, LR.	- A1 and A3 approaches perform, on average, better than the A2 approach.	- Only one method for feature selection is considered (filter). - Only one method for sampling was considered (RUS). - Researchers injected noise into the data set. Therefore, introducing the noise factor complicated the problem set. For the uncertainty of noise, the conclusions of researchers may not fit for the situation without noise.
[23]	Different application fields.	Steel Plates Faults, Statlog (Landsat Satellite), and Spam base, Optdigits, Musk, Semeion Handwritten Digit, and Internet Advertisements.	1- Data Sampling after Feature Selection (A1). 2- Feature Selection after Data Sampling (A2).	C4.5	- Feature selection before sampling is mostly better (A1, mostly, better than A2). - Under sampling performs better than oversampling (when the dataset is highly imbalanced).	- Only two methods for feature selection are considered: filter and wrapper. - Only two methods for sampling were considered: RUS and ROS. - After sampling data in approach A2, the study did not focus on modeling data based on original or sampled data. - The type of classifier was not considered, as only one classifier was used in all experiments.
[12]	Software defect prediction	Eclipse 2.1	1- Feature selection based on sampled data, and modeling based on original data (A1/ DS-FS-UnSam). 2- Feature selection based on sampled data, and modeling based on sampled data (A2/ DS-FS-Sam). 3- Feature selection based on original data and modeling based on sampled data (A2/ FS-DS).	NB, MLP, KNN, SVM, AND LR.	- A1 approach performs, on average, better than the A3 and A2 approaches when the RUS was employed. However, the A3 approach performs better than the other two approaches when the oversampling method (ROS or SMOTE).	- Only one method for feature selection is considered (filter). - Considered three methods for sampling: RUS, ROS and SMOTE
[11]	Tweet sentiment analysis	Sentiment140 corpus (5:95 and 20:80 positive: negative class ratios)	1- Feature selection based on sampled data, and modeling based on original data (A1/ DS-FS-UnSam). 2- Feature selection based on sampled data, and modeling based on sampled data (A2/ DS-FS-Sam). 3- Feature selection based on original data, and modeling based on sampled data (A2/ FS-DS).	KNN, C4.5, SVM, MLP, LR, NB, RBF.	- Overall, there is little difference between the three approaches. - The A3 approach performs better than the A1 approaches for the 5:95 dataset (highly imbalanced scenarios).	- Only one method for feature selection is considered (filter). - Only one method for sampling was considered (RUS).
[24]	Webspam detection	WEBSPAM-UK2006 and WEBSPAM-UK200	- Feature Selection after Data Sampling	C4.5	- The study focuses on constructing an ensemble decision tree classifier through data balancing and select several optimal feature subsets for each sub-classifier.	- The study focuses on one scenario for data balancing (feature selection after data sampling using only one method for feature selection and one method for sampling RUS).
[10]	Heart failure prediction	Hull-LifeLab clinical database	1- Feature Selection before Data Sampling (A1). 2- Feature Selection after Data Sampling (A2).	RF, J48	-Handling a high dimensional imbalanced dataset is valuable for all feature selection methods. -The resampling of the imbalanced class generates a good enhancement in performance results for all measurements.	- Only two methods for sampling were considered: RUS and ROS - Three methods for feature selection are considered: filter, wrapper, and Embedded. - After sampling data in approach A2, the study did not focus on modeling data based on original or sampled data.

NN= neural network, SVM= Support-vector machine, NB= Naive Bayes, MLP= multilayer perceptron, LR= Logistic regression, KNN= k-nearest neighbors, RBF= Radial basis function network, RF= Random Forest.

training and testing sets. In this technique, the original data is divided into two parts. The first part trains the model and makes predictions on the second part. Moreover, it evaluates the predictions for the expected results, and the split data size is often based on the dataset size. The prevalence of using the training data is between 70 and 80%, while for testing, it is between 20 and 30% [25].

IV. DATA SCREENING AND SAMPLE SELECTION

A. DATASET DESCRIPTION

The difficulty of obtaining large-scale reliable data and the cost problems related to this process were explained in the Introduction section. For these reasons, in this study, we aim to use the FIFA soccer video game data, which is commonly used in the literature. It has been used successfully to predict the results of football matches [6], [26], and we have seen that it was comparable or better than other sources of football data [6].

The EA Sports FIFA video game series system began in 2009. It offers detailed information including weekly updates about a broad set of European soccer players and their skills, which covers three aspects: physical, mental, and technical skills. This information is available on the official website of the game (http://sofifa.com/). The FIFA video game series system has resulted in a huge amount of fine-grained data, which has proven to be particularly useful for coaches, sports analysts, and football fans worldwide [27], [28]. In this study, we used the dataset for the FIFA18 game available on Kaggle.[1] It contains 17,980 cases, where each case relates to one football player. Each football player has more than 70 attributes. These attributes can be divided into personal attributes (e.g., age, nationality, and value), performance attributes (e.g., overall, potential, and stamina), and position (which is classified into 14 positions). For our analysis, we selected 29 continuous variables (player performance indicators on a scale of 0–100) and one categorical variable (player's position).

B. SAMPLE ANALYSIS

In this paper, we seek to characterize football positions based on individual player skills, covering three aspects: physical, mental, and technical skills [16], [29]. Table 3 summarizes these skills.

There are 11 different positions in a soccer team in general: goalkeeper, centre back, full back, wing back, centre midfielder, central attacking midfielder, central defensive midfielder, midfielder, winger, centre forward, and striker. These positions represent both the player's primary role and their operation area on the pitch [14]. The location of these positions on a football pitch is shown in Figure 1.

Previous studies showed that each position has different criteria. For instance, while the criterion 'ball control' is significant for central attacking midfielders, it is less critical for central defensive midfielders (see, e.g., [30]). Therefore, we also seek to find out the essential features required in the process of characterizing players.

[1]https://www.kaggle.com/thec03u5/fifa-18-demo-player-dataset

TABLE 3. Description of the main variables that are used in the performance analysis of a football player.

Physical Skills	Mental Skills	Technical Skills
Acceleration	Aggression	Ball Control
Agility	Composure	Crossing
Balance	Interceptions	Curve
Jumping	Marking	Dribbling
Reactions	Positioning	Finishing
Sprint Speed	Vision	Free Kick Accuracy
Stamina	-	Heading Accuracy
Strength	-	Long Passing
-	-	Penalties
-	-	Short Passing
-	-	Shot Power
-	-	Sliding Tackle
-	-	Standing Tackle
-	-	Volleys

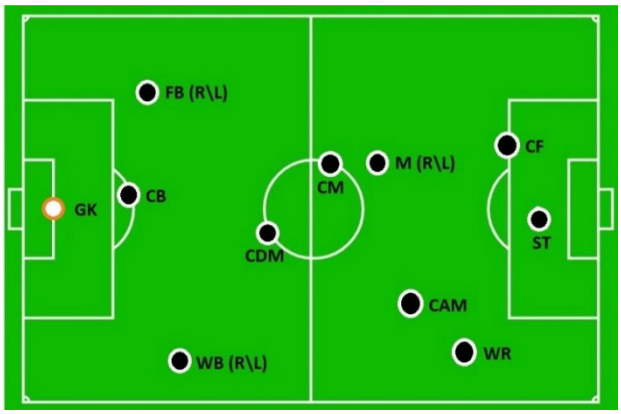

FIGURE 1. Positions of players on football (soccer) pitch.

Through data analysis, we found that the sample dataset had the following characteristics:

1) There are 14 positions for football players: goalkeeper (GK), centre back (CB), right and left full back (RB/LB), right- and left-wing back (RWB/LWB), centre midfielder (CM), central attacking midfielder (CAM), central defensive midfielder (CDM), right and left midfielder (RM/LM), right and left winger (RW/LW), centre forward (CF), and striker (ST).
2) A goalkeeper is a special position that differs from other positions in terms of some characteristics like 'overhead exit' and 'person-to-person battles'. So, we ignore this position as a separate position.
3) The original dataset consisted of 17,981 players, but since some players had multiple positions, we doubled the number of players to 27,251.
4) To avoid class overlapping, we considered nine primary positions out of 14, where previous studies [30], [31] indicated that the skills required for some positions are the same. For example, right and left full back, and right and left midfielder. Table 4 summarizes the nine primary positions.

TABLE 4. Summary of the nine main positions.

n.	Primary position	Comments
#0	Striker	Independent position
#1	Winger	RW and LW are considered as one position.
#2	Midfielder	RM and LM are considered as one position.
#3	Center Midfielder	Independent position
#4	Central Attacking Midfielder	Independent position
#5	Center Forward	Independent position
#6	Central Defensive Midfielder	Independent position
#7	Center back	Independent position
#8	Defender	Full backs (RB/LB) and Wing backs (RWB/LWB) are considered as one position. Moreover, in modern football, there's hardly any difference between Full backs and Wing backs.

Class overlapping is a critical problem in which data samples appear as valid instances of more than one class. Researchers have found that misclassification often occurs near class boundaries, where overlapping usually occurs as well. Therefore, the class overlapping problem may be responsible for noise in datasets [32], [33].

5- There are 29 relevant features for the prediction in players' position (see Table 3).
6- Imbalanced data. For example, among 27,251 players for nine positions, only 350 players were centre forward, which accounted for only 1.28% of the samples. Figure 2 shows the class imbalance ratios of data. The observations in Figure 2 are summarized in Table 5. Formula (1) represents the method to calculate each class's imbalance ratio [34].

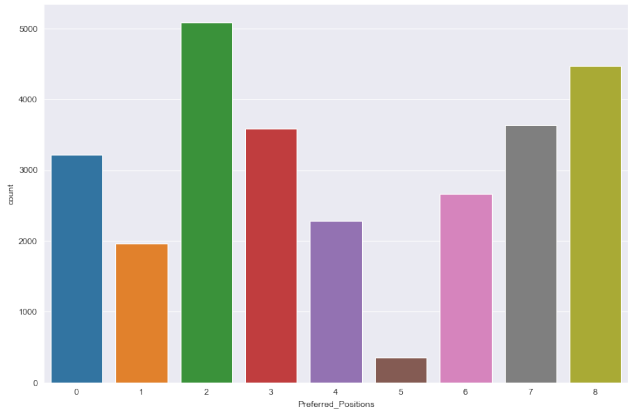

FIGURE 2. Distribution of players in the original dataset according to nine positions.

Label cardinality of D is the average number of labels of the examples in D:

$$LC(D) = \frac{1}{|D|}\sum_{i=1}^{|D|}|Y_i|,$$
$$\text{Imbalance ratio} = \frac{|Y_i|}{LC(D) - |Y_i|} \quad (1)$$

TABLE 5. Distribution of players in the dataset.

Class Label	N. of players	Imbalance ratio
0	3219	11.8%
1	1962	7.1%
2	5090	18.6%
3	3589	13.1%
4	2281	8.3%
5	350	1.28%
6	2663	9.7%
7	3630	13.3%
8	4467	16.3%
Total number of players	27251	

V. ELEMENTS OF COMPARATIVE STUDY

A. RESAMPLING TECHNIQUES

A dataset is entitled to be imbalanced if it contains more samples from one class than from the rest. Resampling techniques are considered one of the most commonly used means to deal with imbalanced datasets. Resampling techniques include removing examples from the majority class (undersampling) or duplicating examples from the minority class (oversampling), as shown in Figure 3. Therefore, in this paper, we present a comparative study about the influence of combining these resampling methods and three feature selection methods for tackling class imbalance.

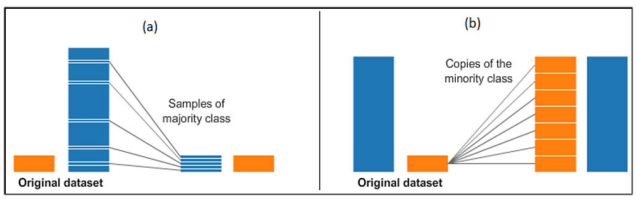

FIGURE 3. A general example for resampling techniques: (a) undersampling, (b) oversampling.

For this study, we selected the following resampling methods, which are among the most reported methods in the literature. Additionally, these methods have not been tested with feature selection methods, except in the study presented by [12], in which only one way has been used to feature selection (see Table 2).

1) RANDOM UNDERSAMPLING (RUS)

The RUS deletes examples in the majority class and can result in losing information invaluable to a model.

2) RANDOM ONDERSAMPLING (ROS)

The ROS duplicates examples from the minority class in the training dataset and can result in overfitting for some models [35].

3) SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE (SMOTE)

In the SMOTE method, each minority class sample is taken and synthetic samples are created by looking at any or all of

the sample's k neighbour. Thus, the minority class becomes oversampled. The main difference from other sampling methods is the synthetic samples' production, which is facilitated by looking at their nearest neighbours instead of copying and replicating the minority class samples. The main disadvantage of the SMOTE method is the noise it generates. Noises are often intricately intertwined with the other class; they confuse the model and are hard to predict [36].

B. FEATURE SELECTION

Feature selection is one of the main preprocessing steps in many machine learning applications. It is a process of selecting a subset of relevant features, reducing data dimensionality for use in model construction (so that prediction performance will be improved or maintained), and speeding up the learning process. Many features may be irrelevant or contain no useful information. Thus, their inclusion may negatively impact classification performance. Therefore, feature selection also helps data miners acquire a better understanding of their data by telling them about the necessary features and their correlation with each other [8], [37]. In contrast to other dimensionality reduction techniques, such as those based on projection (e.g., principal component analysis), feature selection techniques do not alter the variables' original representation. Thus, they preserve the original semantics of the variables, thereby offering the advantage of interpretability by a domain expert [38]. In this way, they can find out the required player performance attributes for each position, and a coach would have an objective criterion to select the players.

Feature selection techniques can be broadly categorized into three categories, depending on how they combine the feature selection search with the construction of a classification model: filter, wrapper, and embedded. The following subsections provide a brief explanation of each technique and the most prominent advantages, disadvantages, and algorithms used in this study.

1) FILTER METHODS

The random filter feature selection methods use statistical techniques to obtain a specific score and assign it to each feature. By only looking at the intrinsic properties of the data, filter methods can assess the relevance of features [39]. The selection of a subset of features is made as a preprocessing step; this means that after each feature's score is calculated, the low-scoring features are removed, and the remaining are used as predictors in the model construction [38], [40].

Thanks to its simplicity, filter feature selection methods are widely used in sports predictions [41]. Examples of this method and the usage areas in sports prediction are information gain, chi-squared, ANOVA [42], mutual information (MI) [43], correlation-based feature selection (CFS), INTERACT algorithm, ReliefF, and minimum redundancy maximum relevance (mRMR) [44].

In this study, we used CFS, chi-squared, MI, and mRMR as filter feature selection methods. The following subsections provide a brief explanation of each algorithm.

a: CORRELATION-BASED FEATURE SELECTION ALGORITHM (CFS)

This method uses the correlation-based heuristic evaluation function to determine the merit of a particular feature subset for predicting the class label and the level of correlation among them. In other words, the CFS is used to calculate subsets for the evaluation of features with the following basic hypotheses, which are based on the heuristic that 'Good feature subsets contain features highly correlated (predictive of) with the classification, yet uncorrelated (not predictive of) to each other'.

The heuristic uses the Pearson's correlation coefficient which can be calculated using the following formula:

$$Ms = \frac{k\overline{r_{Cf}}}{\sqrt{k + k\,(k-1)\,\overline{r_f}}} \tag{2}$$

where Ms is the merit of the current subset of features, k is the number of features, $\overline{r_{Cf}}$ is the mean of the correlations between each feature and the class variable, and $\overline{r_{ff}}$ is the mean of the pairwise correlations between every two features [12].

Correlation coefficients whose magnitude is between 0.7 and 0.9 indicate variables that can be considered highly correlated. Moreover, coefficients whose magnitudes are between 0.5 and 0.7 indicate variables that can be considered moderately correlated [45].

b: CHI-SQUARED (CS)

The chi-squared feature evaluation tells the significance of each of the original features. Based on this, the user can choose to keep the most-significant and discard the least-significant features. In the chi-squared feature selection, a feature's significance is measured by the chi-squared test statistic between the feature and the target class. Equation (3) is used to calculate the chi-squared statistic, where 'observed' is the actual number of class observations and 'expected' is the number of class observations that would be expected if there were no relationships between the feature and class. The sum is over each value of the feature since the chi-squared method requires that numeric features be discretized before calculating [46].

$$x^2 = \Sigma(observed - expected)^2/(expected) \tag{3}$$

A high chi-squared test score indicates that the feature and the target class are unlikely to be independent and that, therefore, we should keep the feature in our new dataset.

c: MUTUAL INFORMATION (MI)

The MI is another statistical method used in feature selection. It is the measure of how two variables (x, y) are mutually dependent. It evaluates the 'measure of data' gathered about one arbitrary variable through the other random variable. Equation 4 is used to calculate the MI between two discrete random variables x and y :

$$I(X, Y) = \Sigma_{x\epsilon X}\Sigma_{y\epsilon Y}p(x, y)\log(\frac{P(x, y)}{p\,(x)\,p(y)}) \tag{4}$$

where $p(x, y)$ is the joint probability function of X and Y, and $p(x)$ and $p(y)$ are the marginal probability distribution functions of X and Y, respectively.

For continuous random variables, the summation is replaced by a double integral as

$$I(X, Y) = \int_y \int_x p(x, y) \log\left(\frac{P(x, y)}{p(x)\, p(y)}\right) dx dy \tag{5}$$

d: MINIMUM REDUNDANCY MAXIMUM RELEVANCE (mRMR) TECHNIQUE

The mRMR is a feature selection approach that tends to select features with a high correlation with the class (output) and a low correlation among themselves. For continuous features, the F-statistics can be used to calculate correlation with the class (relevance), and the Pearson's correlation coefficient can be used to calculate the correlation among the features (redundancy). Thereafter, features are selected one by one by applying a greedy search to maximize the objective function, which is a function of relevance and redundancy. Two commonly used types of the objective functions are mutual information difference (MID) criterion and mutual information quotient (MIQ) criterion, which represent the difference or the quotient of relevance and redundancy [47].

2) WRAPPER METHODS

The wrapper feature selection methods generate several feature subsets evaluated according to their predictive power when used with a specific classifier [39]. As described by Saeys *et al.*, a search procedure in the space of possible feature subsets is defined, and various subsets of features are generated and evaluated. The evaluation of a specific subset of features is obtained by training and testing a specific classification model. A search algorithm is then 'wrapped' around the classification model to search the space for all feature subsets. The application of wrapper methods to high-dimensional datasets requires special attention since with the increase in number of features, the space of feature subsets grows exponentially and becomes computationally impossible. The heuristic search methods are used to guide the search for an optimal subset of features to tackle this problem [38], [40].

The two most common greedy searching techniques used to perform wrapper-style feature selection are sequential feature selection and recursive feature elimination (RFE). Sequential feature selection algorithms can be either forward as sequential forward selection (SFS) or backward as sequential backward elimination (SBE). In this study, we used SFS, SBE, and RFE as wrapper feature selection methods. The following subsections provide a brief explanation of each algorithm.

a: SEQUENTIAL FORWARD SELECTION (SFS)

The SFS starts from the empty set. It performs best when only a small number of features are involved. Nonetheless, the main disadvantage of SFS is that it cannot remove features that become insignificant after the addition of other features.

b: SEQUENTIAL BACKWARD ELIMINATION (SBE)

The SBE works in the opposite way to that of the SFS. The SBE starts with a full set of features. It works best with many features in the dataset [48].

c: RECURSIVE FEATURE ELIMINATION (RFE)

Given an external estimator that assigns weights to features (e.g., the coefficients of a linear model), the RFE aims to select features by recursively considering smaller and smaller sets of features. First, the estimator is trained on the initial set of features and the importance of each feature is obtained through any specific attribute. Then, the least important features are pruned from the current set of features. This procedure is recursively repeated on the pruned set until the desired number of features to select is eventually reached [49].

3) EMBEDDED METHODS

As for the wrappers, the embedded methods depend on a specific learning algorithm. Further, while the search and evaluation procedures are separated in the wrappers, the embedded method performs feature selection in the classifier construction using its internal parameters. Therefore, they are faster than the wrappers and are more efficient as they avoid the use of all the available data by not needing to divide the data into a training set and a test set [50].

Decision trees such as RF, extra tree, and XGBoost are popular approaches for embedded methods. Other embedded methods are the least absolute shrinkage and selection operator (LASSO) with the L1 penalty and ridge with the L2 penalty for constructing a linear model. These two methods shrink many features to zero or almost near zero [51]. In this study, we used RF and LASSO as embedded feature selection methods. The following subsections provide a brief explanation of each algorithm.

a: EMBEDDED-RANDOM FOREST (RF)

The feature evaluation approach based on the RF is known as embedded method [52]. It provides a variable importance criterion for each feature by computing the mean decrease in the classification accuracy for out-of-bag (OOB) data from bootstrap sampling [53]. Assuming bootstrap samples $b = 1, \ldots,$ B, the mean decreases in classification accuracy $\overline{D_j}$ for variable x_j as the importance measure is given by

$$D = \frac{1}{B} \sum\nolimits_{b=1}^{B} (R_b^{oob} - R_{bj}^{oob}) \tag{6}$$

where R_b^{oob} denotes the classification accuracy for OOB data ℓ_b^{oob} using the classification model T_b; and R_{bj}^{oob} is the classification accuracy for OOB data ℓ_{bj}^{oob} permuted the values of variable x_j in ℓ_b^{oob} ($j = 1, \ldots, N$). Finally, a z-score of variable x_j representing the variable importance criterion could be computed using the formula $x_j = \frac{\overline{D_j}}{s_j/\sqrt{B}}$, after the standard deviation s_j of the classification accuracy decrease has been calculated.

b: EMBEDDED-LEAST ABSOLUTE SHRINKAGE AND SELECTION OPERATOR (LASSO)

The LASSO is a powerful method that helps perform regularization (L1) and feature selection of the given data. It penalizes the beta coefficients in a model. The LASSO method limits the sum of the values of the model parameters, where the sum has to be less than the specific fixed value. This shrinks some of the coefficients to zero, indicating that a particular predictor or certain features will be multiplied by zero to estimate the target. During this process, the variables that have a non-zero coefficient after shrinking are selected to be a part of the model. It also adds a penalty term to the cost function with a lambda value tuned [51]. This is how the LASSO reduces the overfitting caused and helps in feature selection; it uses the following equation:

$$\sum_{i=1}^{M}\left(y_i-\hat{y}_i\right)^2=\sum_{i=1}^{M}\left(y_i-\sum_{j=0}^{p}w_j\times x_{ij}\right)^2+\lambda\sum_{j=0}^{p}\times\left|w_j\right| \quad (7)$$

When lambda (λ) is 0, the equation is reduced, leading to no elimination of the parameters. An increase in λ causes an increase in bias, and a decrease in λ causes an increase in variance.

C. CLASSIFICATION

The final step of our proposed methodology involves a supervised learning predictive model. The classification stage aims to characterize football players into nine positions. In this study, we used only one classifier in empirical comparisons as we seek to increase classification accuracy based on balancing techniques and feature selection regardless of the classifier. Therefore, the RF was selected for this task. The RF was chosen owing to its frequent use in the literature of characterizing players [17], [18] and data mining domains. Moreover, it is a relatively fast state-of-the-art algorithm [12], [54].

1) RANDOM FOREST (RF)

The RF is an ensemble classification approach that has proved its high accuracy and superiority. The RF consists of several uncorrelated decision trees. For a classification operation, the RF classifier creates a set of decision trees from a randomly selected subset of training data. It then collects the votes from different decision trees to decide the final class of the test target. The general architecture of the RF is shown in Figure 4.

The RF was first introduced in 1999 by Leo Breiman. In his studies, Breiman explored various methods of randomization of decision trees (sampling), for example, using bagging or boosting [55]. In bootstrap, the classifier creates new datasets from the original data and then calculates the average errors in these groups to estimate variance. (Unlike cross-validation sampling like hold out in which data is divided into two parts for training and testing.) As for the RF, its hallmarks mainly include [56]

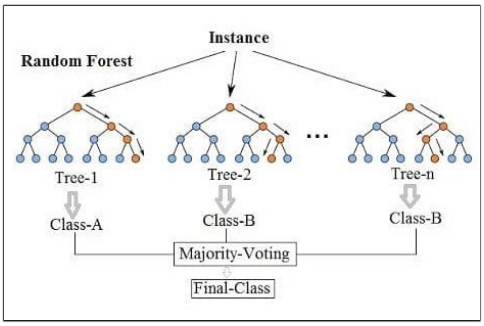

FIGURE 4. Random Forests-based classification process.

1) Bootstrap sampling (bagging) – randomly selecting number of samples with replacement.
2) Feature selection randomly – randomly selecting only a small number of *m* instances in each node's split.
3) Full-depth decision tree growing.
4) OOB error – calculating error on the samples not selected during bootstrap sampling.

VI. EVALUATION METRICS

In machine learning, several metrics are used to evaluate the performance of the classification models. Generally, statistical methods, such as hold-out (train-and-test split), cross-validation, and bootstrap, can be used with predictive models to get estimates of model performance using the training set [57]. Confusion matrix, classification report, and accuracy are considered as the most critical metrics for evaluating the classification models using the testing data [25].

A. HOLD-OUT (TRAIN AND TEST SPLIT)

In classification problems, the simplest way to evaluate the algorithm's performance is to use different training and testing sets. In this technique, the original dataset is split into two parts. The first part trains the algorithm and makes predictions on the second part and then evaluates predictions against the expected results. Generally, the size of the split data is based on the size of the dataset. It the common to use 70–90% of the data for training and 10–30% for testing [25]. In this study, we used the train-and-test split for splitting data. The samples (see Table 5) were randomly divided into 70% for training and 30% for testing.

B. CONFUSION MATRIX

A confusion matrix is a practical presentation of the accuracy of a model with two or more classes. The matrix displays predictions on the x-axis and accuracy outcomes on the y-axis. The matrix cells are the number of predictions made by the algorithm [25], as shown in Figure 5.

In the confusion matrix, true positives (TP) correspond to the number of correct positive predictions. Similarly, false positives (FP), true negatives (TN), and false negatives (FN) are the numbers of incorrect positive predictions, correct negative predictions, and incorrect negative predictions, respectively.

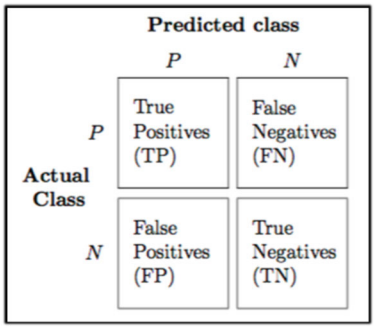

FIGURE 5. Confusion Matrix.

Like the previous studies [58], the minority class was considered positive, while the majority class was considered negative. Therefore, according to Tables 4 and 5, the centre forward position was regarded as positive (minority class) while the midfielder was considered as negative (majority class), which means

TP: The player is a midfielder and is classified as a midfielder.

FP: The player is a midfielder and is classified as a centre forward.

TN: The player is a centre forward and is classified as a centre forward.

FN: The player is a centre forward and is classified as a midfielder.

C. CLASSIFICATION REPORT

Classification report provides a convenient representation when working on classification problems to give you a quick idea of a model's accuracy using several measures derived from the confusion matrix for the model. The classification report displays the precision, recall, *F1*-score, and support (the number of actual occurrences of the class in the specified dataset). These metrics give a more profound intuition of the classifier behaviour over total accuracy, which can mask functional weaknesses in one type of binary or multi-class problem. In binary classification, the precision, recall, and *F1*-score are defined as shown in formulas (8), (9), and (10), respectively [59]. However, in multi-classification, it can compute the performance measures in the same way as it can define one class as positive and the other as unfavourable.

$$\text{Precision} = \frac{\text{TP}}{\text{FP} + \text{TP}} \tag{8}$$

$$\text{Recall} = \frac{\text{TP}}{\text{FN} + \text{TP}} \tag{9}$$

$$\text{F1 score} = 2 * \frac{(\text{precision} * \text{recall})}{(\text{recall} + \text{precision})} \tag{10}$$

Since we are dealing with an imbalanced class problem, recall is an important metric to consider. From the football point of view, having high values of FN is not good. Midfielders are usually good at defense and offence, unlike the centre forward players who are not required to be good at defense. This means, having too many FP is not as severe as the latter case.

D. CLASSIFICATION ACCURACY

A typical metric for measuring the performance of learning systems is the classification accuracy rate. It is the number of correct predictions made divided by the total number of predictions made. Classification accuracy is considered as the most popular evaluation metric for classification problems in machine learning [25]. However, empirical evidence shows that this measure is biased regarding the data imbalance and proportions of correct and incorrect classifications [60]. Therefore, these shortcomings have motivated the search for new measures such as precision, recall and *F1*-score. Classification accuracy is defined in formula (11):

$$\text{Accuracy} = \frac{(\text{TN} + \text{TP})}{(\text{TP} + \text{FP} + \text{FN} + \text{TN})} \tag{11}$$

VII. RESEARCH DESIGN

In this study, we aim to create a machine learning classifier to characterize football players' positions. Moreover, we seek to address the imbalance problem in the dataset.

A. RESEARCH QUESTIONS

The research questions for this study are as follows:

Research Question 1: Can machine learning algorithms make recommendations to improve team performance?

Research Question 2: Can data mining techniques improve the performance of machine learning algorithms?

To answer the first question, we discuss the implementation of the baseline algorithm in Section (8.1) and evaluation of its performance on our data. Thus, we explore one of the primary aspects of sports analytics in football using a supervised predictive model to characterize players according to nine positions. For the second question, in Sections (8.2) and (8.3), we discuss the importance of applying two preprocessing techniques, resampling and feature selection, to jointly reduce the complexity of training datasets and solve the class imbalance problem. We used three different algorithms of the following preprocessing techniques: RUS, ROS, and SMOTE. For resampling, nine methods for feature selection were used to evaluate the effectiveness of the various techniques and build a machine learning classifier.

B. DESIGN OF COMPARATIVE EXPERIMENTS

A Python module called scikit-learn was used to build machine learning models and execute feature selection algorithms. Moreover, a python toolbox called imbalanced-learn API was used to tackle the curse of imbalanced datasets. All models were created using the default parameters unless otherwise noted.

The design of comparative experiments was based on the 3×9 crossings of resampling and feature selection methods using the RF classifier, which produced four different combinations:

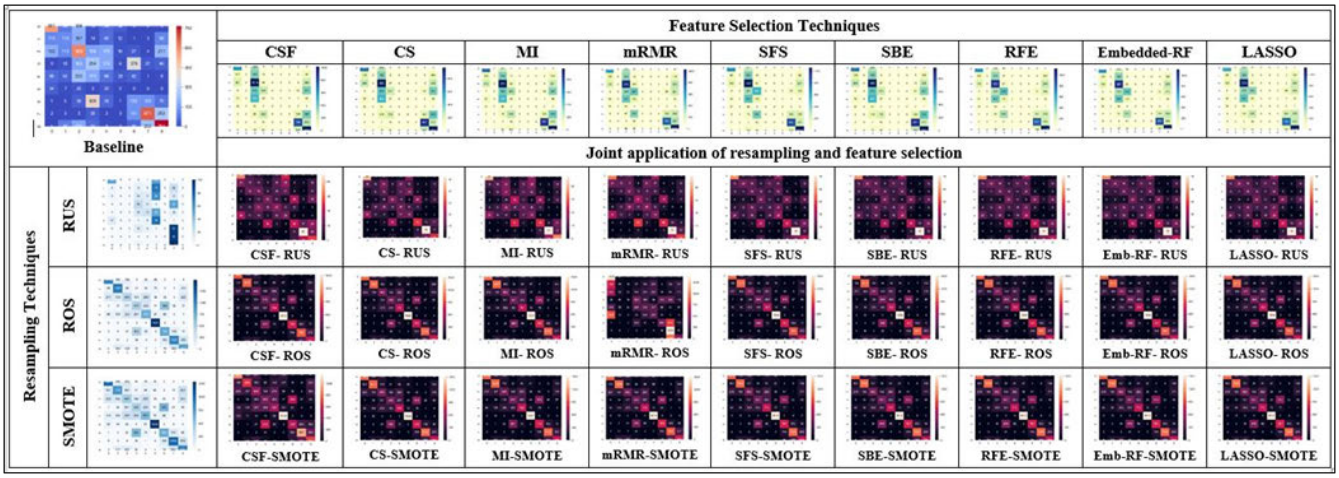

FIGURE 6. Averaged results of experiments in terms of the confusion matrix.

1) Baseline (one model).
2) Resampling versus baseline (three models).
3) Feature selection versus baseline (nine models).
4) Joint application of resampling and feature selection (27 models).

Figure 6 shows the results of four previous combinations in terms of the confusion matrix for the dataset, which used to evaluate the algorithm in Research Question 1 and to develop the predictive models for Research Question 2 (consisting of 40 models). In the next section, the results of these matrices is interpreted and clearly presented in terms of precision, recall, and *F1*-score.

VIII. ANALYSIS OF RESULTS

The results in Figure 6 were analyzed in the following three ways, organized from a low to a high level of detail. (Each comparative analysis involves all the possible cases obtained from the combination of a classifier [see 5.3], a data partition [see 6.1], and performance measures [see 6.2, 6.3, and 6.4].

A. BASELINE MODEL

A baseline is a simple procedure for making predictions on a specific predictive problem. The skill of this model provides the bedrock for the lowest acceptable performance of a machine learning model on the original dataset, by which all other models can be evaluated. If a model achieves performance at or below the baseline, it means that something is wrong or the model is not appropriate for your problem. Random forest is used to establish the baseline model in our experiments. The classification report results in terms of accuracy, precision and recall, which were summarized in Figure 7.

We acknowledge that tuning the algorithm's parameters can lead to better results, but we adopt the classification model's default parameters in all the experiments. Thus, we seek to maintain baseline performance as the basis for comparison. The focus of this study is not to examine the pros and cons of the used classification models. However, it focuses on investigating the joint influence of resampling and feature selection for tackling class imbalance.

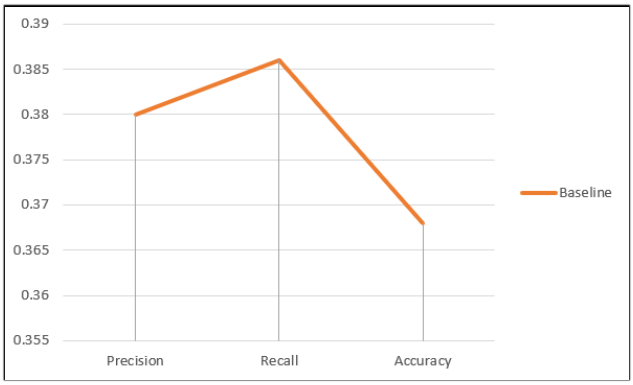

FIGURE 7. Summary regarding the performance of the baseline model (in terms of accuracy, precision, and recall).

B. LEVEL A (THE HIGHEST LEVEL OF ANALYSIS)

1) RESAMPLING VERSUS BASELINE (A1)

In this sublevel, the resampling techniques used in the study were analyzed. In the first column of Figure 6, the classification results obtained from resampled training sets were compared with those provided by the corresponding original training sets in terms of the confusion matrix.

Figure 8 shows the resampled sets after applying the three resampling techniques in terms of accuracy, precision and recall. Owing to random behaviour of the RUS, ROS, and SMOTE, the resampled sets were randomly divided into 70% for training and 30% for testing in each experiment involving these techniques. The results obtained from these experiments are summarized as follows: The use of resampling techniques improved accuracy, precision, and recall compared with baseline, and the ROS had a relative advantage compared to other methods of balancing.

TABLE 6. Attributes table for features that were selected using nine types of feature selection.

	Feature (skill)	CSF	CS	MI	mRMR	SFS	SBE	RFE	Emb. - RF	LASSO
1	Acceleration									
2	Aggression									
3	Agility					✓				
4	Balance						✓			
5	Ball control									
6	Composure									
7	Crossing					✓	✓	✓	✓	✓
8	Curve						✓			
9	Dribbling	✓	✓	✓		✓	✓		✓	
10	Finishing	✓	✓	✓		✓	✓	✓	✓	✓
11	Free kick accuracy					✓				
12	heading accuracy			✓		✓		✓	✓	
13	Interceptions	✓	✓	✓		✓	✓	✓	✓	✓
14	Jumping									
15	Long passing				✓	✓	✓	✓	✓	
16	Long shots	✓	✓	✓			✓			
17	Marking	✓	✓	✓	✓			✓	✓	✓
18	Penalties		✓							✓
19	Positioning	✓	✓	✓	✓		✓		✓	✓
20	Reactions				✓					
21	Short passing				✓	✓				
22	Shot power				✓					✓
23	Sliding tackle	✓	✓	✓	✓			✓	✓	✓
24	Sprint speed				✓					
25	Stamina				✓					
26	Standing tackle	✓	✓	✓	✓			✓	✓	✓
27	Strength							✓	✓	
28	Vision	✓				✓	✓	✓	✓	✓
29	Volleys	✓	✓	✓						✓
	Total	10	10	10	10	10	10	10	12	11

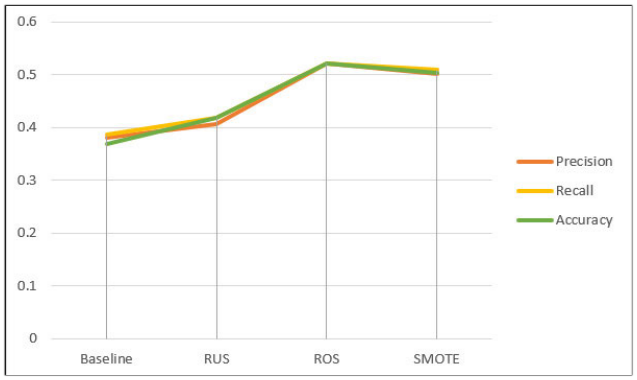

FIGURE 8. Summaries regarding the performance of resampling methods models (in terms of accuracy, precision and recall).

2) FEATURE SELECTION VERSUS BASELINE (A2)

In this sublevel, the feature selection techniques used in the study were analyzed. Table 6 shows the new subsets whose dimensionality was reduced by the nine feature selection techniques, where nine subsets were produced. The implementations of the feature selection techniques used are those included in the scikit-learn library with their default parameters except for some parameters set in advance. In the CSF, the most critical parameters are the correlations between each feature and the class variable. We set it to 0.5 in our experiments, which leads to a subset of 10 features being produced. For the chi-squared algorithm, we set ($k = 10$) for all the datasets to specify the 10 best features with highest chi-squared statistics. For the MI algorithm, the 10 features with the highest MI score were selected. For the mRMR algorithm, the MIQ criterion was used as an objective function to specify the 10 best features that have high correlation with the class. For SFS, SBE, and RFE algorithms, subsets of 10 features were generated and evaluated by the RF classifier. For the LASSO algorithm, we set alpha to 0.1 in our experiments, and a subset of 11 features was produced.

Analysing the results, in the first row in Figure 6, the classification results obtained from the training and test sets whose dimensionality was reduced by feature selection techniques are compared with those provided by the corresponding original training sets (baseline) in terms of the confusion matrix. The classification report results in terms of accuracy, precision, and recall are summarized in Figure 9, for sets whose dimensionality was reduced. The results obtained from this experiment can be summarized as follows: The use of feature selection techniques alone tends to deteriorate results for all models compared with the baseline model in terms of accuracy, precision, and recall; thus, the experiments demonstrate that the evaluated feature selection techniques did not improve the accuracy of the classifier.

TABLE 7. Attributes table for features that were selected using nine types of feature selection over RUS.

	Feature (skill)	CSF	CS	MI	mRMR	SFS	SBE	RFE	Emb. - RF	LASSO
1	Acceleration						✓			✓
2	Aggression									
3	Agility					✓	✓			
4	Balance					✓	✓			
5	Ball control									
6	Composure					✓	✓			
7	Crossing							✓	✓	
8	Curve					✓				
9	Dribbling			✓		✓	✓		✓	✓
10	Finishing	✓	✓	✓				✓	✓	✓
11	Free kick accuracy									
12	heading accuracy		✓					✓	✓	
13	Interceptions	✓	✓	✓		✓		✓	✓	✓
14	Jumping					✓	✓			
15	Long passing				✓			✓	✓	
16	Long shots		✓	✓		✓	✓			
17	Marking	✓	✓	✓	✓			✓	✓	✓
18	Penalties									
19	Positioning		✓	✓	✓	✓			✓	✓
20	Reactions				✓					✓
21	Short passing				✓					
22	Shot power				✓		✓			✓
23	Sliding tackle	✓	✓	✓	✓			✓	✓	✓
24	Sprint speed				✓		✓	✓		
25	Stamina				✓	✓	✓			
26	Standing tackle	✓	✓	✓	✓			✓	✓	✓
27	Strength		✓					✓	✓	
28	Vision			✓					✓	
29	Volleys	✓	✓	✓						
	Total	6	10	10	10	10	10	10	12	10

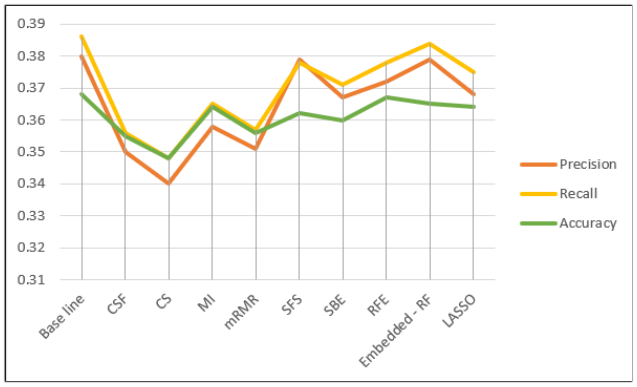

FIGURE 9. Summary regarding the performance of the feature selection methods (in terms of accuracy, precision and recall).

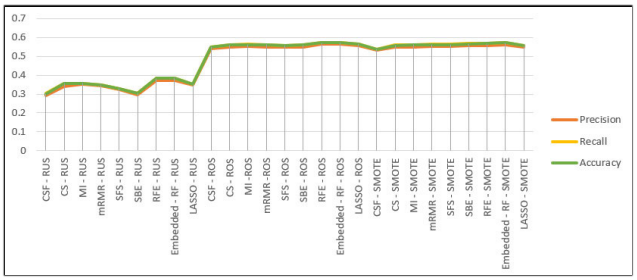

FIGURE 10. The results of the approaches that combine resampling and feature selection models (in terms of accuracy, precision and recall).

C. LEVEL B (A MIDDLE LEVEL OF ANALYSIS)

In this level, the classification results obtained from the joint application of resampling and feature selection were compared with baseline results in terms of the confusion matrix (see the second, third, and fourth rows in Figure 6). Experiments in this level included the following steps (which represent the main methodology proposed for this study):

1) Applying the sampling technique to deal with class imbalance.
2) Using the feature selection technique to deal with the high dimensionality problem.
3) Modelling the models based on sampled data and the new subset selected by feature selection techniques.

It is worth noting that applying feature selection to the balanced data produced subsets that differed slightly from those that resulted from applying feature selection to the unbalanced data (Tables 7, 8, and 9).

In Figure 11, the results of these experiments in terms of accuracy, precision, and recall are summarized in Figure 10.

Experimental comparisons with baseline model are made on the average basis of the average accuracy, precision, and recall for the nine feature selection methods over the

TABLE 8. Attributes table for features that were selected using nine types of feature selection over ROS.

	Feature (skill)	CSF	CS	MI	mRMR	SFS	SBE	RFE	Emb. - RF	LASSO
1	Acceleration						✓			✓
2	Aggression									
3	Agility					✓				
4	Balance									
5	Ball control									
6	Composure						✓			
7	Crossing							✓	✓	
8	Curve					✓				
9	Dribbling			✓			✓		✓	✓
10	Finishing	✓	✓	✓		✓		✓	✓	✓
11	Free kick accuracy									
12	heading accuracy		✓	✓				✓	✓	
13	Interceptions	✓	✓	✓			✓	✓	✓	✓
14	Jumping									
15	Long passing				✓	✓		✓	✓	
16	Long shots		✓	✓						
17	Marking	✓	✓	✓	✓	✓		✓	✓	✓
18	Penalties					✓				
19	Positioning	✓	✓	✓	✓		✓		✓	✓
20	Reactions				✓					✓
21	Short passing				✓		✓			
22	Shot power				✓					✓
23	Sliding tackle	✓	✓	✓	✓	✓	✓	✓	✓	✓
24	Sprint speed				✓		✓	✓		
25	Stamina				✓	✓	✓			
26	Standing tackle	✓	✓	✓	✓	✓	✓	✓	✓	✓
27	Strength		✓						✓	
28	Vision					✓		✓	✓	
29	Volleys	✓	✓	✓						
	Total	**7**	**10**	**10**	**10**	**10**	**10**	**10**	**12**	**10**

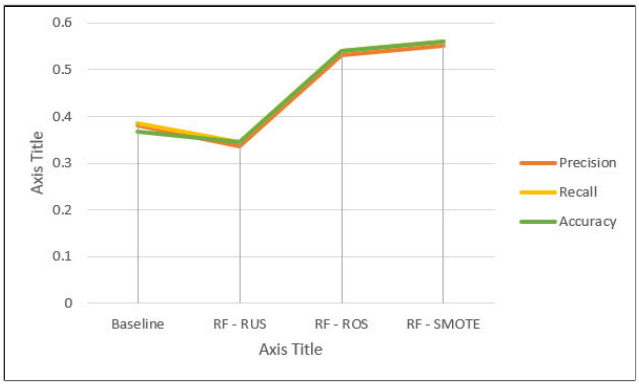

FIGURE 11. The results of the approaches that combine resampling and feature selection models (based on the average accuracy, precision, and recall for the nine feature selection methods).

three balancing methods RUS, ROS, and SMOTE. The following inference can be made from Figures 10 and 11:

- The use of feature selection with data balanced by the RUS method leads to deteriorating results for all models compared with the baseline in terms of accuracy, precision, and recall.
- The use of feature selection with data balanced by the ROS and SMOTE methods leads to improved accuracy, precision, and recall compared with the baseline model.
- The use of feature selection with data balanced by the ROS and SMOTE methods leads to improved accuracy, precision, and recall compared with the baseline. Thus, there is no single filter, wrapper, or embedded-based feature selection method that is the best. Therefore, the experimental comparisons with the baseline were made on the basis of the average accuracy, precision, and recall.

Appendix A shows the performance evaluation of all tested models for each class, in addition to the baseline.

D. LEVEL C (THE LOWEST LEVEL OF ANALYSIS)

At this level, the results of the proposed methodology for the joint application of resampling and feature selection analyzed in Level B were compared with the results obtained only by the use of a single technique and from the original imbalanced training set analyzed in Level A. The results of all the previous experiments of Levels A and B versus the baseline model are summarized in Figure 12. The results obtained from this figure can be summarized as follows:

- The results showed superiority of the proposed methodology, involving the joint application of resampling and feature selection with data balanced by the ROS and SMOTE methods, compared to the results obtained

TABLE 9. Attributes table for features that were selected using nine types of feature selection over SMOTE.

	Feature (skill)	CSF	CS	MI	mRMR	SFS	SBE	RFE	Emb. - RF	LASSO
1	Acceleration									✓
2	Aggression					✓				
3	Agility						✓			
4	Balance					✓				
5	Ball control									
6	Composure					✓				
7	Crossing							✓	✓	
8	Curve									
9	Dribbling			✓					✓	✓
10	Finishing	✓	✓	✓				✓	✓	✓
11	Free kick accuracy					✓				
12	heading accuracy		✓				✓	✓	✓	
13	Interceptions	✓	✓	✓			✓	✓	✓	✓
14	Jumping						✓			
15	Long passing				✓			✓	✓	
16	Long shots		✓	✓						
17	Marking	✓	✓	✓	✓	✓	✓	✓	✓	✓
18	Penalties			✓		✓	✓			
19	Positioning	✓	✓	✓	✓		✓		✓	✓
20	Reactions				✓					✓
21	Short passing				✓					
22	Shot power				✓	✓				✓
23	Sliding tackle	✓	✓	✓	✓			✓	✓	✓
24	Sprint speed				✓	✓	✓			
25	Stamina				✓					
26	Standing tackle	✓	✓	✓	✓			✓	✓	✓
27	Strength		✓			✓	✓	✓	✓	
28	Vision					✓	✓	✓	✓	
29	Volleys	✓	✓	✓						
	Total	**7**	**10**	**10**	**10**	**10**	**10**	**10**	**12**	**10**

only by the use of a single technique and from the original imbalanced training set analyzed in Level A.

- By comparing the results obtained from Figures 10 and 12, the most accurate model (embedded-RF feature selection and ROS) achieved an accuracy of 57.3%, precision of 56.4%, and recall of 57.4%. The model built using the RFE feature selection and ROS had a comparable accuracy and precision of 57.2% and 57.2%, respectively, and a recall of 56.3%.
- The proposed methodology improved prediction accuracy compared to baseline. Moreover, it produced a drastic reduction in the number of features, from 29 to 10 on average. This means these features, at least in a statistical sense, are the most influential factors for predicting player position.
- Based on the model that achieved the highest accuracy (embedded-RF feature selection and ROS) and Table 8, the most important attributes in characterizing a player's position are crossing, dribbling, finishing, heading accuracy, interceptions, long passing, marking, positioning, sliding tackle, standing tackle, strength, and vision.
- This model could be used as an initial model for characterizing football players according to the multivariate performance data; this information can be beneficial to coaches since it can be used as an objective criterion for evaluating a player.

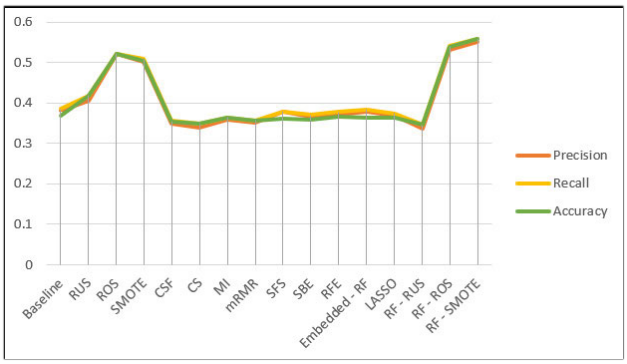

FIGURE 12. The results for all the experiments (in terms of accuracy, precision, and recall).

IX. DISCUSSION AND CONCLUSION

A. RESEARCH QUESTION 1

To answer the Research Question 1 (Can machine learning algorithms make recommendations to improve team performance?), we implemented the baseline algorithm using an RF classifier to characterize football player's positions and evaluate their performance on our data. Since the data used

TABLE 10. A review analysis of the comparison models.

	Class	Prec.	Rec.	F1-sc.	Supp.		Class	Prec.	Rec.	F1-sc.	Supp.		Class	Prec.	Rec.	F1-sc.	Supp.
BISE LINE MODEL	0	0.577	0.605	0.59	956	CSF	0	0.565	0.610	0.586	953	SFS	0	0.576	0.589	0.582	956
	1	0.262	0.2	0.227	604		1	0.271	0.2	0.230	604		1	0.246	0.175	0.205	604
	2	0.376	0.391	0.383	1503		2	0.393	0.407	0.399	1527		2	0.375	0.403	0.388	1503
	3	0.214	0.236	0.224	1070		3	0.221	0.230	0.225	1075		3	0.215	0.237	0.226	1070
	4	0.102	0.095	0.099	651		4	0.098	0.086	0.092	684		4	0.100	0.092	0.096	651
	5	0.011	0.011	0.011	90		5	0.022	0.020	0.021	101		5	0.035	0.033	0.034	90
	6	0.176	0.159	0.167	839		6	0.153	0.153	0.153	792		6	0.165	0.151	0.158	839
	7	0.609	0.599	0.604	1101		7	0.639	0.627	0.633	1096		7	0.613	0.599	0.606	1101
	8	0.528	0.559	0.543	1362		8	0.529	0.568	0.548	1344		8	0.533	0.565	0.549	1362
RUS	0	0.551	0.619	0.583	113	CS	0	0.539	0.548	0.543	956	SBE	0	0.569	0.576	0.573	956
	1	0.25	0.195	0.219	113		1	0.238	0.171	0.199	604		1	0.223	0.164	0.189	604
	2	0.286	0.264	0.275	106		2	0.337	0.360	0.348	1503		2	0.363	0.387	0.375	1503
	3	0.365	0.396	0.38	96		3	0.184	0.203	0.193	1070		3	0.204	0.225	0.214	1070
	4	0.221	0.232	0.227	99		4	0.055	0.049	0.052	651		4	0.100	0.097	0.098	651
	5	0.322	0.277	0.298	101		5	0.020	0.022	0.021	90		5	0.031	0.033	0.032	90
	6	0.471	0.434	0.452	113		6	0.117	0.099	0.107	839		6	0.143	0.129	0.135	839
	7	0.65	0.705	0.677	95		7	0.581	0.582	0.582	1101		7	0.594	0.579	0.586	1101
	8	0.546	0.651	0.594	109		8	0.473	0.518	0.494	1362		8	0.524	0.551	0.537	1362
ROS	0	0.703	0.664	0.683	1513	MI	0	0.569	0.571	0.570	956	RFE	0	0.570	0.582	0.576	956
	1	0.614	0.686	0.648	1524		1	0.251	0.195	0.220	604		1	0.238	0.177	0.203	604
	2	0.374	0.286	0.324	1548		2	0.353	0.379	0.366	1503		2	0.371	0.392	0.381	1503
	3	0.342	0.313	0.327	1530		3	0.191	0.207	0.199	1070		3	0.216	0.236	0.226	1070
	4	0.49	0.484	0.487	1540		4	0.079	0.071	0.075	651		4	0.096	0.092	0.094	651
	5	0.829	1	0.907	1495		5	0.031	0.033	0.032	90		5	0.021	0.022	0.022	90
	6	0.458	0.492	0.474	1564		6	0.142	0.119	0.130	839		6	0.149	0.130	0.139	839
	7	0.69	0.681	0.685	1491		7	0.603	0.606	0.604	1101		7	0.610	0.602	0.606	1101
	8	0.541	0.546	0.544	1538		8	0.481	0.523	0.501	1362		8	0.516	0.551	0.533	1362
SMOTE	0	0.626	0.656	0.641	1513	mRMR	0	0.520	0.526	0.523	956	Embedded - RF	0	0.586	0.593	0.589	956
	1	0.548	0.49	0.517	1524		1	0.242	0.182	0.208	604		1	0.251	0.187	0.214	604
	2	0.309	0.297	0.303	1548		2	0.346	0.369	0.357	1503		2	0.374	0.397	0.385	1503
	3	0.255	0.251	0.253	1530		3	0.202	0.219	0.210	1070		3	0.209	0.230	0.219	1070
	4	0.423	0.392	0.406	1540		4	0.083	0.078	0.081	651		4	0.102	0.095	0.098	651
	5	0.835	0.857	0.846	1495		5	0.023	0.022	0.023	90		5	0.010	0.011	0.011	90
	6	0.413	0.43	0.422	1564		6	0.147	0.129	0.137	839		6	0.161	0.145	0.153	839
	7	0.666	0.67	0.668	1491		7	0.590	0.559	0.574	1101		7	0.625	0.615	0.620	1101
	8	0.502	0.559	0.529	1538		8	0.488	0.543	0.514	1362		8	0.536	0.565	0.550	1362

TABLE 10. *(Continued.)* **A review analysis of the comparison models.**

	Class	Prec.	Rec.	F1-sc.	Supp.		Class	Prec.	Rec.	F1-sc.	Supp.		Class	Prec.	Rec.	F1-sc.	Supp.
LASSO	0	0.560	0.576	0.568	956	mRMR - RUS	0	0.442	0.442	0.442	120	Embedded – RF - RUS	0	0.558	0.642	0.597	120
	1	0.239	0.166	0.196	604		1	0.234	0.161	0.190	112		1	0.250	0.152	0.189	112
	2	0.363	0.390	0.376	1503		2	0.208	0.214	0.211	98		2	0.227	0.224	0.226	98
	3	0.215	0.237	0.226	1070		3	0.248	0.291	0.268	103		3	0.252	0.301	0.274	103
	4	0.098	0.089	0.094	651		4	0.243	0.245	0.244	102		4	0.245	0.245	0.245	102
	5	0.031	0.033	0.032	90		5	0.235	0.253	0.244	95		5	0.299	0.274	0.286	95
	6	0.136	0.120	0.128	839		6	0.426	0.351	0.385	114		6	0.380	0.360	0.369	114
	7	0.594	0.583	0.589	1101		7	0.583	0.733	0.649	101		7	0.635	0.723	0.676	101
	8	0.525	0.565	0.544	1362		8	0.450	0.450	0.450	100		8	0.495	0.530	0.512	100
CSF - RUS	0	0.439	0.417	0.427	120	SFS - RUS	0	0.492	0.483	0.487	120	LASSO - RUS	0	0.559	0.517	0.537	120
	1	0.188	0.134	0.156	112		1	0.282	0.196	0.232	112		1	0.205	0.143	0.168	112
	2	0.209	0.194	0.201	98		2	0.196	0.194	0.195	98		2	0.186	0.194	0.190	98
	3	0.202	0.194	0.198	103		3	0.185	0.214	0.198	103		3	0.232	0.252	0.242	103
	4	0.242	0.225	0.234	102		4	0.173	0.137	0.153	102		4	0.258	0.225	0.241	102
	5	0.177	0.211	0.192	95		5	0.237	0.295	0.263	95		5	0.241	0.295	0.265	95
	6	0.308	0.281	0.294	114		6	0.356	0.316	0.335	114		6	0.385	0.351	0.367	114
	7	0.545	0.663	0.598	101		7	0.628	0.752	0.685	101		7	0.619	0.693	0.654	101
	8	0.317	0.400	0.345	100		8	0.321	0.360	0.340	100		8	0.400	0.480	0.436	100
CS - RUS	0	0.535	0.575	0.554	120	SBE - RUS	0	0.430	0.483	0.455	120	CSF - ROS	0	0.683	0.622	0.651	1580
	1	0.182	0.107	0.135	112		1	0.275	0.196	0.229	112		1	0.608	0.638	0.622	1573
	2	0.196	0.194	0.195	98		2	0.204	0.194	0.199	98		2	0.345	0.285	0.312	1549
	3	0.248	0.243	0.245	103		3	0.159	0.165	0.162	103		3	0.306	0.282	0.293	1481
	4	0.214	0.216	0.215	102		4	0.183	0.186	0.184	102		4	0.458	0.485	0.471	1453
	5	0.210	0.232	0.220	95		5	0.218	0.200	0.209	95		5	0.837	0.990	0.907	1527
	6	0.391	0.395	0.393	114		6	0.250	0.263	0.256	114		6	0.459	0.485	0.472	1586
	7	0.642	0.782	0.705	101		7	0.603	0.693	0.645	101		7	0.660	0.680	0.670	1463
	8	0.425	0.450	0.437	100		8	0.330	0.340	0.335	100		8	0.492	0.475	0.483	1531
MI - RUS	0	0.535	0.508	0.521	120	RFE - RUS	0	0.552	0.575	0.563	120	CS - ROS	0	0.706	0.640	0.672	1580
	1	0.225	0.143	0.175	112		1	0.195	0.152	0.171	112		1	0.621	0.653	0.636	1573
	2	0.216	0.245	0.230	98		2	0.268	0.306	0.286	98		2	0.359	0.299	0.326	1549
	3	0.295	0.320	0.307	103		3	0.265	0.291	0.278	103		3	0.302	0.289	0.295	1481
	4	0.202	0.176	0.188	102		4	0.219	0.206	0.212	102		4	0.462	0.489	0.475	1453
	5	0.211	0.242	0.225	95		5	0.312	0.263	0.286	95		5	0.836	0.997	0.909	1527
	6	0.386	0.342	0.363	114		6	0.394	0.342	0.366	114		6	0.465	0.468	0.466	1586
	7	0.628	0.703	0.664	101		7	0.630	0.743	0.682	101		7	0.673	0.710	0.691	1463
	8	0.424	0.530	0.471	100		8	0.500	0.570	0.533	100		8	0.513	0.486	0.499	1531

TABLE 10. *(Continued.)* **A review analysis of the comparison models.**

	Class	Prec.	Rec.	F1-sc.	Supp.		Class	Prec.	Rec.	F1-sc.	Supp.		Class	Prec.	Rec.	F1-sc.	Supp.
MI - ROS	0	0.708	0.653	0.680	1580	RFE - ROS	0	0.723	0.649	0.684	1580	CS - SMOTE	0	0.706	0.651	0.677	1580
	1	0.602	0.653	0.627	1573		1	0.619	0.666	0.642	1573		1	0.610	0.643	0.626	1573
	2	0.374	0.292	0.328	1549		2	0.383	0.317	0.347	1549		2	0.361	0.303	0.330	1549
	3	0.308	0.284	0.296	1481		3	0.331	0.305	0.317	1481		3	0.305	0.287	0.296	1481
	4	0.458	0.482	0.470	1453		4	0.460	0.493	0.476	1453		4	0.462	0.485	0.473	1453
	5	0.836	0.988	0.906	1527		5	0.836	0.991	0.907	1527		5	0.841	0.988	0.908	1527
	6	0.473	0.503	0.487	1586		6	0.478	0.484	0.481	1586		6	0.463	0.479	0.471	1586
	7	0.684	0.703	0.693	1463		7	0.679	0.702	0.690	1463		7	0.676	0.705	0.691	1463
	8	0.519	0.501	0.510	1531		8	0.550	0.538	0.544	1531		8	0.512	0.486	0.499	1531
mRMR - ROS	0	0.698	0.644	0.670	1580	Embedded – RF - ROS	0	0.722	0.660	0.690	1580	MI - SMOTE	0	0.699	0.645	0.671	1580
	1	0.609	0.646	0.627	1573		1	0.618	0.655	0.636	1573		1	0.606	0.654	0.629	1573
	2	0.359	0.283	0.317	1549		2	0.377	0.307	0.339	1549		2	0.362	0.287	0.320	1549
	3	0.320	0.294	0.307	1481		3	0.318	0.299	0.308	1481		3	0.317	0.295	0.306	1481
	4	0.455	0.490	0.472	1453		4	0.473	0.506	0.489	1453		4	0.465	0.497	0.480	1453
	5	0.834	0.991	0.906	1527		5	0.837	0.995	0.909	1527		5	0.838	0.993	0.909	1527
	6	0.483	0.511	0.497	1586		6	0.485	0.482	0.484	1586		6	0.471	0.489	0.480	1586
	7	0.671	0.684	0.678	1463		7	0.688	0.712	0.700	1463		7	0.675	0.699	0.678	1463
	8	0.514	0.496	0.505	1531		8	0.545	0.541	0.543	1531		8	0.510	0.487	0.498	1531
SFS - ROS	0	0.704	0.646	0.674	1580	LASSO - ROS	0	0.705	0.639	0.670	1580	mRMR - SMOTE	0	0.708	0.644	0.674	1580
	1	0.614	0.645	0.629	1573		1	0.608	0.652	0.629	1573		1	0.618	0.664	0.640	1573
	2	0.360	0.305	0.330	1549		2	0.372	0.305	0.336	1549		2	0.362	0.291	0.322	1549
	3	0.319	0.300	0.309	1481		3	0.320	0.290	0.304	1481		3	0.317	0.291	0.304	1481
	4	0.456	0.483	0.469	1453		4	0.473	0.506	0.489	1453		4	0.468	0.496	0.481	1453
	5	0.840	0.991	0.909	1527		5	0.839	0.988	0.907	1527		5	0.836	0.991	0.907	1527
	6	0.469	0.484	0.476	1586		6	0.473	0.499	0.486	1586		6	0.474	0.505	0.489	1586
	7	0.657	0.678	0.668	1463		7	0.677	0.692	0.648	1463		7	0.670	0.679	0.675	1463
	8	0.512	0.485	0.498	1531		8	0.526	0.511	0.518	1531		8	0.512	0.500	0.506	1531
SBE - ROS	0	0.694	0.631	0.661	1580	CSF - SMOTE	0	0.545	0.562	0.553	1580	SFS - SMOTE	0	0.707	0.641	0.672	1580
	1	0.605	0.657	0.630	1573		1	0.482	0.411	0.444	1573		1	0.616	0.661	0.637	1573
	2	0.362	0.281	0.317	1549		2	0.296	0.287	0.291	1549		2	0.364	0.298	0.328	1549
	3	0.317	0.301	0.309	1481		3	0.228	0.238	0.233	1481		3	0.316	0.295	0.305	1481
	4	0.457	0.480	0.468	1453		4	0.321	0.311	0.316	1453		4	0.461	0.490	0.475	1453
	5	0.839	0.991	0.909	1527		5	0.712	0.756	0.733	1527		5	0.839	0.990	0.908	1527
	6	0.471	0.487	0.479	1586		6	0.355	0.334	0.344	1586		6	0.473	0.499	0.486	1586
	7	0.672	0.692	0.682	1463		7	0.629	0.661	0.644	1463		7	0.675	0.707	0.690	1463
	8	0.519	0.512	0.515	1531		8	0.422	0.460	0.440	1531		8	0.519	0.482	0.500	1531

TABLE 10. *(Continued.)* **A review analysis of the comparison models.**

	Class	Prec.	Rec.	F1-sc.	Supp.		Class	Prec.	Rec.	F1-sc.	Supp.
SBE - SMOTE	0	0.712	0.649	0.679	1580	Embedded – RF - SMOTE	0	0.714	0.657	0.684	1580
	1	0.616	0.652	0.633	1573		1	0.615	0.655	0.634	1573
	2	0.357	0.292	0.321	1549		2	0.380	0.305	0.338	1549
	3	0.320	0.291	0.305	1481		3	0.325	0.297	0.311	1481
	4	0.473	0.502	0.487	1453		4	0.463	0.490	0.476	1453
	5	0.838	0.997	0.910	1527		5	0.837	0.991	0.908	1527
	6	0.474	0.503	0.488	1586		6	0.479	0.508	0.493	1586
	7	0.690	0.712	0.701	1463		7	0.689	0.706	0.698	1463
	8	0.521	0.500	0.510	1531		8	0.545	0.530	0.537	1531
RFE - SMOTE	0	0.709	0.652	0.679	1580	LASSO - SMOTE	0	0.703	0.638	0.669	1580
	1	0.618	0.648	0.633	1573		1	0.621	0.648	0.634	1573
	2	0.366	0.298	0.329	1549		2	0.351	0.294	0.320	1549
	3	0.316	0.292	0.303	1481		3	0.315	0.298	0.306	1481
	4	0.471	0.502	0.486	1453		4	0.454	0.480	0.467	1453
	5	0.835	0.991	0.907	1527		5	0.839	0.993	0.909	1527
	6	0.470	0.482	0.476	1586		6	0.464	0.471	0.467	1586
	7	0.682	0.698	0.690	1463		7	0.678	0.699	0.689	1463
	8	0.543	0.541	0.542	1531		8	0.504	0.496	0.500	1531

in the study were unbalanced, the accuracy of the model did not exceed 37%.

B. RESEARCH QUESTION 2

To answer the Research Question 2 (Can data mining techniques improve the performance of machine learning algorithms?), we examined the importance of applying two preprocessing techniques, re-sampling and feature selection, to jointly reduce the complexity of training datasets and solve the class imbalance problem by making empirical comparisons; a total of 40 predictive models were tested. The proposed methodology for the study consisted of three main steps. The first step consisted of applying the sampling technique to deal with class imbalance; the second step consisted of the feature selection technique, which dealt with the high dimensionality problem, and the third step combined feature selection and data sampling to deal with both the issues.

Our approach goes beyond the studies presented in Table 2. We offer a comprehensive study in which we uses nine selection algorithms based on the main feature selection algorithms – filter, wrapper, and embedded in addition to three methods for data balancing. We trained models using the RF as an objective function for each position. Based on the experiments, we concluded that 1) feature selection techniques did not improve the accuracy of the baseline model, 2) balancing techniques improved accuracy compared to the baseline, and 3) the results showed superiority of the proposed methodology, involving the joint application of resampling and feature selection with data balanced by the ROS and SMOTE, compared to the results obtained only through the use of a single technique and from the original imbalanced training set.

Overall, the proposed methodology improved the prediction accuracy compared to the baseline, and an accuracy of more than 57% was reported. Moreover, the proposed methodology provided a significant decrease in the number of features, from 29 to 10 on average. This means these features, at least in a statistical sense, are the most influential for predicting player position. This information can be beneficial to coaches since these features can be used as an objective criterion for evaluating a player. Moreover, this model could be used as an initial model for characterizing football players according to the multivariate performance data.

On the other hand, regarding player position, our approach goes beyond the studies presented in Table 1, which were limited to classifying players into the three central positions (defender, midfielder, and attacker). In contrast, we sought to find the specific role in those positions (e.g., centre midfielder or central attacking midfielder).

This study supports the concept that specific performance indicators define each position of players in football. Additionally, we believe that the quantitative analysis of the multivariate performance data using machine learning methods (like classification) is an essential step in this process.

Finally, our study has shown that the data collected from video games such as FIFA could improve prediction quality. Furthermore, these games can also be used as an essential source for retrieving sports data and executing artificial intelligence analyses.

X. FURTHER STUDIES

Further studies can improve the performances reported in this study. New experiments could easily be set up by merely replacing techniques regarding data balancing, feature selection, or classification algorithms. In this sense, we intend to expand the applied options of the different feature selection techniques by trying ReliefF [44] as a filter method, bidirectional elimination (Stepwise Selection) [61] as a wrapper method, and regularized L2 logistic regression as an embedded method. We will also plan more experiments on datasets and other applications to test whether the proposed approach can be used more generally and robustly.

APPENDIX A

See Table 10.

CONFLICT OF INTEREST STATEMENT

On behalf of all authors, Mustafa A. AL-ASADI certifies that the submission is an original study and is not under review at any other publication. There is no financial interest to report.

REFERENCES

[1] L. Cotta, "Using FIFA soccer video game data for soccer analytics," in *Proc. Workshop Large Scale Sports Anal.*, 2016, pp. 1–4.

[2] R. Asif, "Football (soccer) analytics: A case study on the availability and limitations of data for football analytics research," *Int. J. Comput. Sci. Inf. Secur.*, vol. 14, no. 11, p. 516, 2016.

[3] R. Vroonen, "Predicting the potential of professional soccer players," in *Proc. Mach. Learn. Data Mining Sports Anal.*, 2017, pp. 1–10.

[4] M. J. Fry and J. W. Ohlmann, "Introduction to the special issue on analytics in sports, Part I: General sports applications," *Interfaces*, vol. 42, no. 2, pp. 105–108, Apr. 2012.

[5] L. Cotta, "Usin g FIFA soccer video game data for soccer analytics," in *Proc. Workshop Large Scale Sports Anal.*, 2016, pp. 1–4.

[6] J. Shin and R. Gasparyan, "A novel way to soccer match prediction," Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, Final Project CS229, 2014.

[7] R. Longadge and S. Dongre, "Class imbalance problem in data mining review," 2013, *arXiv:1305.1707*.

[8] K. Gao, T. M. Khoshgoftaar, and A. Napolitano, "Combining feature subset selection and data sampling for coping with highly imbalanced software data," in *Proc. SEKE*, 2015, pp. 439–444.

[9] S. S. Pattanayak and M. Rout, "Experimental comparison of sampling techniques for imbalanced datasets using various classification models," in *Progress in Advanced Computing and Intelligent Engineering*. Singapore: Springer, 2018, pp. 13–22.

[10] M. A. Khaldy, "Resampling imbalanced class and the effectiveness of feature selection methods for heart failure dataset," *Int. Robot. Autom. J.*, vol. 4, no. 1, pp. 1–10, Feb. 2018.

[11] J. D. Prusa and T. M. Khoshgoftaar, "Comparing approaches for combining data sampling and feature selection to address key data quality issues in tweet sentiment analysis," in *Proc. 29th Int. Flairs Conf.*, 2016, pp. 608–613.

[12] K. Gao, T. M. Khoshgoftaar, and A. Napolitano, "Aggregating data sampling with feature subset selection to address skewed software defect data," in *Proc. Int. J. Softw. Eng. Knowl. Eng.*, Dec. 2015, pp. 1531–1550.

[13] V. Gaurav and G. Chakraborty, "Scouting in soccer with applied machine learning," Oklahoma State Univ., Stillwater, OK, USA, 2019, Paper 3828.

[14] C. Soto-Valero, "A Gaussian mixture clustering model for characterizing football players using the EA Sports' FIFA video game system," *Revista Int. Ciencias Deporte*, vol. 13, no. 49, pp. 244–259, 2017, doi: 10.5232/ricyde2017.04904.

[15] M. Tavana, "A fuzzy inference system with application to player selection and team formation in multi-player sports," *Sport Manage. Rev.*, vol. 16, no. 1, pp. 97–110, 2013.

[16] R. Obiedat, "Identification of players positions in a multi-agent game using artificial neural networks and C4.5 algorithm: A comparative study," *Sci. Res. Essays*, vol. 8, no. 17, pp. 682–688, 2013.

[17] V. Rao and A. Shrivastava, "Team strategizing using a machine learning approach," in *Proc. Int. Conf. Inventive Comput. Informat. (ICICI)*, Nov. 2017, pp. 1032–1035.

[18] K. Apostolou and C. Tjortjis, "Sports analytics algorithms for performance prediction," in *Proc. 10th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Jul. 2019, pp. 1–4.

[19] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya, "A survey on addressing high-class imbalance in big data," *J. Big Data*, vol. 5, no. 1, p. 42, Dec. 2018.

[20] M. Lango, "Tackling the problem of class imbalance in multi-class sentiment classification: An experimental study," *Found. Comput. Decis. Sci.*, vol. 44, no. 2, pp. 151–178, 2019.

[21] R. Martín-Félez and R. A. Mollineda, "On the suitability of combining feature selection and resampling to manage data complexity," in *Proc. Conf. Spanish Assoc. Artif. Intell.* Seville, Spain: Springer, 2009.

[22] A. A. Shanab, T. M. Khoshgoftaar, R. Wald, and J. Van Hulse, "Comparison of approaches to alleviate problems with high-dimensional and class-imbalanced data," in *Proc. IEEE Int. Conf. Inf. Reuse Integr.*, Aug. 2011, pp. 234–239.

[23] H. Yin and K. Gai, "An empirical study on preprocessing high-dimensional class-imbalanced data for classification," in *Proc. IEEE IEEE 17th Int. Conf. High Perform. Comput. Commun.*, Aug. 2015, pp. 1314–1319.

[24] X. Y. Lu, M.-S. Chen, J.-L. Wu, P.-C. Chang, and M.-H. Chen, "A novel ensemble decision tree based on under-sampling and clonal selection for web spam detection," *Pattern Anal. Appl.*, vol. 21, no. 3, pp. 741–754, Aug. 2017.

[25] J. Brownlee, *Machine Learning Mastery With Python*, vol. 527. Melbourne, VIC, Australia: Machine Learning Mastery, 2016, pp. 100–120.

[26] D. Prasetio and D. Harlili, "Predicting football match results with logistic regression," in *Proc. Int. Conf. Adv. Inform., Concepts, Theory Appl. (ICAICTA)*, Aug. 2016, pp. 1–5.

[27] A. S. Markovits and A. I. Green, "FIFA, the video game: A major vehicle for soccer's popularization in the United States," *Sport Soc.*, vol. 20, nos. 5–6, pp. 716–734, 2017.

[28] K. Bandyopadhyay and S. S. N. Mitra, "FIFA World Cup and beyond: Sport, culture, media and governance," *Sport Soc.*, vol. 20, nos. 5–6, pp. 547–554, 2017.

[29] L. Yaldo and L. Shamir, "Computational estimation of football player wages," *Int. J. Comput. Sci. Sport*, vol. 16, no. 1, pp. 18–38, Jul. 2017.

[30] M. Flegl *et al.*, "Personnel selection in complex organizations: A case of Mexican football team for the 2018 World Cup in Russia," *Revista del Centro de Investigación de la Universidad La Salle*, vol. 13, no. 49, pp. 43–66, 2018.

[31] E. O. Ozceylan, "A mathematical model using AHP priorities for soccer player selection: A case study," *South Afr. J. Ind. Eng.*, vol. 27, no. 2, pp. 190–205, Aug. 2016.

[32] H. Xiong, J. Wu, and L. Liu, "Classification with class overlapping: A systematic study," in *Proc. Int. Conf. E-Bus. Intell.*, 2010, pp. 491–497.

[33] S. Gupta and A. Gupta, "Handling class overlapping to detect noisy instances in classification," *Knowl. Eng. Rev.*, vol. 33, p. 45, Jul. 2018.

[34] J.-H. Seo and Y.-H. Kim, "Machine-learning approach to optimize smote ratio in class imbalance dataset for intrusion detection," *Comput. Intell. Neurosci.*, vol. 2018, Nov. 2018, Art. no. 9704672.

[35] F. Sağlam, S. C. Turan, and M. A. Ceng!'z, "Boosting !'le Gözlemlerin Doğru tahmin edilebilirlik seviyesi tespiti ve dengesiz Sınıf probleminde sentetik veri Üretiminde Kullanımı," in *Proc. 6th Int. Multidisciplinary Studies Congr.*, Samsun, Turkey, 2019.

[36] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, 2002.

[37] G. Xu, Y. Zong, and Z. Yang, *Applied Data Mining*. Boca Raton, FL, USA: CRC Press 2013.

[38] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[39] T. Abeel, T. Helleputte, Y. Van de Peer, P. Dupont, and Y. Saeys, "Robust biomarker identification for cancer diagnosis with ensemble feature selection methods," *Bioinformatics*, vol. 26, no. 3, pp. 392–398, 2010.

[40] A.-C. Haury, P. Gestraud, and J.-P. Vert, "The influence of feature selection methods on accuracy, stability and interpretability of molecular signatures," *PLoS ONE*, vol. 6, no. 12, Dec. 2011, Art. no. e28210.
[41] R. P. Bunker and F. Thabtah, "A machine learning framework for sport result prediction," *Appl. Comput. Inform.*, vol. 15, no. 1, pp. 27–33, 2018.
[42] X. Jin *et al.*, "Machine learning techniques and chi-square feature selection for cancer classification using SAGE gene expression profiles," in *Proc. Int. Workshop Data Mining Biomed. Appl.* Berlin, Germany: Springer, 2006.
[43] L. T. Vinh, N. D. Thang, and Y.-K. Lee, "An improved maximum relevance and minimum redundancy feature selection algorithm based on normalized mutual information," in *Proc. 10th IEEE/IPSJ Int. Symp. Appl. Internet*, Jul. 2010, pp. 395–398.
[44] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, J. M. Benítez, and F. Herrera, "A review of microarray datasets and applied feature selection methods," *Inf. Sci.*, vol. 282, pp. 111–135, Mar. 2014.
[45] M. M. Mukaka, "A guide to appropriate use of correlation coefficient in medical research," *Malawi Med. J.*, vol. 24, no. 3, pp. 69–71, 2012.
[46] R. Spencer, "Exploring feature selection and classification methods for predicting heart disease," *Digit. Health*, vol. 6, Mar. 2020, Art. no. 2055207620914777.
[47] M. Radovic, "Minimum redundancy maximum relevance feature selection approach for temporal gene expression data," *BMC Bioinf.*, vol. 18, no. 1, pp. 1–14, 2017.
[48] L. Ladha and T. Deepa, "Feature selection methods and algorithms," *Int. J. Comput. Sci. Eng.*, vol. 3, no. 5, pp. 1787–1797, 2011.
[49] C. M. Salgado and S. M. Vieira, "Machine learning for patient stratification and classification Part 3: Supervised learning," in *Leveraging Data Science for Global Health*. Cham, Switzerland: Springer, 2020, pp. 169–198.
[50] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, Mar. 2003, pp. 1157–1182.
[51] V. Fonti and E. Belitser, "Feature selection using lasso," *VU Amsterdam Res. Paper Bus. Anal.*, vol. 30, pp. 1–25, Mar. 2017.
[52] M. Pal and G. M. Foody, "Feature selection for classification of hyperspectral data by SVM," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 5, pp. 2297–2307, May 2010.
[53] A. Verikas, A. Gelzinis, and M. Bacauskiene, "Mining data with random forests: A survey and results of new tests," *Pattern Recognit.*, vol. 44, no. 2, pp. 330–349, 2011.
[54] P. Manghi, L. Candela, and G. Silvello, "Digital libraries: Supporting open science," in *Proc. 15th Italian Res. Conf. Digit. Libraries (IRCDL)*, vol. 988, Pisa, Italy. Cham, Switzerland: Springer, Jan./Feb. 2019.
[55] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[56] R. Jiang, "A random forest approach to the detection of epistatic interactions in case-control studies," *BMC Bioinf.*, vol. 10, no. 1, p. S65, 2009.
[57] M. Kuhn and K. Johnson, *Applied Predictive Modeling*, vol. 26. New York, NY, USA: Springer, 2013.
[58] J. Tanha, Y. Abdi, N. Samadi, N. Razzaghi, and M. Asadpour, "Boosting methods for multi-class imbalanced data classification: An experimental review," *J. Big Data*, vol. 7, no. 1, pp. 1–47, Dec. 2020.
[59] V. Van Asch, *Macro-and Micro-Averaged Evaluation Measures*. Belgium, Brussels: CLiPS, 2013, pp. 1–27.
[60] F. Provost and T. Fawcett, "Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions," in *Proc. 3rd Int. Conf. Knowl. Discovery Data Mining*, 1997, pp. 1–6.
[61] A. Jovic, K. Brkic, and N. Bogunović, "A review of feature selection methods with applications," in *Proc. 38th Int. Convention Inf. Commun. Technol., Electron. Microelectron. (MIPRO)*, May 2015, pp. 1200–1205.

MUSTAFA A. AL-ASADI received the master's degree in computer engineering from Selçuk University, Konya, Turkey, in 2018, where he is currently pursuing the Ph.D. degree in computer engineering. His master's thesis was on the subject of decision support systems for football team management by using machine learning techniques. His research interests include machine learning, deep learning, predictive models, data analysis, data mining, and pattern recognition.

SAKIR TASDEMİR received the master's and Ph.D. degrees from Selçuk University, in 2004 and 2010, respectively. He is currently the Dean of the Faculty of Engineering and the Head of the Computer Engineering Department, Selçuk University. He has authored many publications in international journals. His research interests include decision support systems, expert systems, image processing, artificial intelligence, and computer aided systems. He is a member of an editorial board of several journals.