Using Machine Learning Techniques for Soccer Results Prediction with Scikit-learn

Minh Vo
Computer Science
Earlham College
801 National Road West
Richmond, Indiana 47374
mbvo14@earlham.edu

Abstract— **Machine learning techniques have been widely known as the best tool for the purpose of prediction. Therefore, machine learning algorithms can also be explored and utilized in the problem of predicting sports results. This project aims to produce a framework that includes the procedures of taking in public data of the English Premier League (soccer) results, processing them with Machine Learning techniques, and determining results of matches. By making use one of the most common toolkits for machine learning, Scikit-learn, the framework classifies the results as home wins, ties, or away wins.**

I. Introduction

With the non-stop improvements in technology, more and more fields are trying to apply computer science to achieve their goals in a more efficient and less time consuming way. Sports are no outsiders to this group of fields. In sports, especially in soccer, technology has become an essential part. Soccer experts now make use of technology to evaluate a player's or a team's performances. Other than using their experience and their management abilities after many years being parts of the game, the soccer coaches also use statistics from data providers to improve their knowledge of their own players and teams so that they can come up with different strategies/tactics that bring them closer to the wins. Besides coaches, soccer analysts also make use of the data to predict results in the future as well as evaluate new talents emerging from the scene. This is where Machine Learning techniques can become useful. Machine Learning is one of the intelligent methodologies that have shown promising results in the domains of classification and prediction [10]. Therefore, in this project, a framework that uses Machine Learning to determine the results of soccer matches is designed.

II. Related Work

Buursma presented an algorithm of predicting match results based on previous results that could beat the bookmakers oods [4]. The approach used data identical to the data source that is used for this project. The author used the WEKA toolkit to process them, which is a popular machine learning toolkit. Buursma used the match histories and classifiers such as MultiClassClassifier and ClassificationViaRegression for the system. The strategies to win against the odds were also briefly discussed but not really well-explained. The author tried to come up the strategies to win against the odds, which is different from the goal of this project. Moreover, Buursma made use of WEKA, which is not the toolkit used in this work.

Constantinou et al. discussed a Bayesian network model to forecast football matches in which the subjective variables signified factors that were not caught by the data [5]. First, the authors introduced the sport and different approaches in predicting outcomes of matches, including the Poisson distribution for goal-based data analysis, regression models, and other machine learning techniques. Next, the authors discussed the data source (on Football-Data) and their model, which considered generic factors for both home and away teams, including strength, form, psychology, and fatigue. While Constantinou et al. made use of subjective information such as fatigue and psychological impact, this project does not take those materials into consideration.

Kumar discussed another approach to the problem of sports prediction [6]. The author tried to identify the players attributes that were related to ratings, that affected the result, that determined a set of team ratings and finally, how well the expert ratings could predict the next match result. The author used detailed

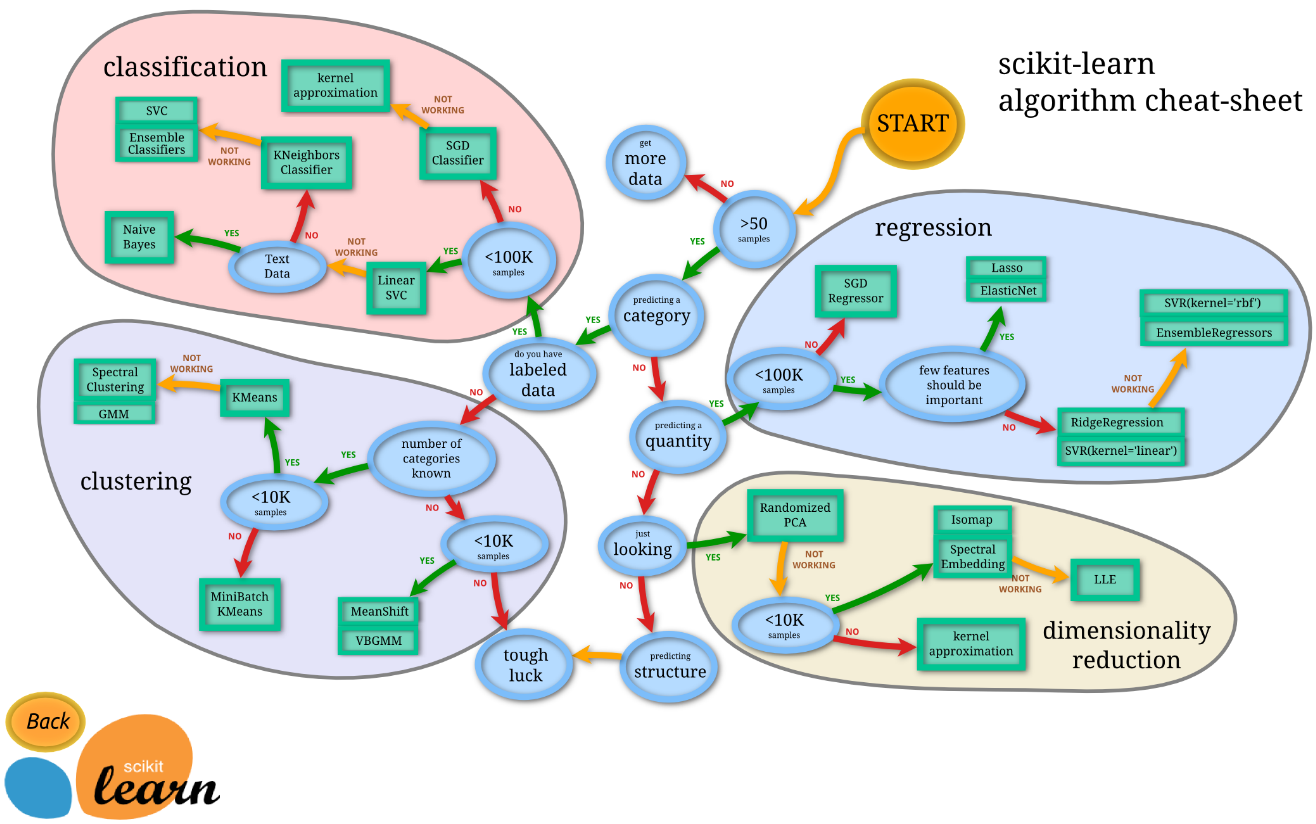

Figure 1: Often the hardest part of solving a machine learning problem can be finding the right estimator for the job. Different estimators are better suited for different types of data and different problems. The flowchart is designed to give users a bit of a rough guide on how to approach problems with regard to which estimators to try on your data [13].

player statistics from OPTA. This is an interesting approach as a team's strength may rely more on the starting players than on the rotation players. ”What will happen if an important player has to miss a game or is on a poor run of games?” This was also the initial approach to this project. However, due to the expense of acquiring players' data, the initial idea could not be worked on.

III. Background

A. Knowledge about The Sport and the English Premier League

To do any type of learning in a particular field, which is soccer in this case, one needs to understand the basic ideas of the sport as well as the league from which the data are being retrieved and used. The main characteristics about soccer and the English Premier League, of which the results are the data to be used, are explained in this section. Here, simple knowledge of the sport and the league that should be grasped before actually going into the more detailed statistics of the sport in the Data Understanding section is introduced.

Soccer (or football) is a sport that involves two teams, each of which has eleven players on the pitch, including one goalkeeper. The basic idea of soccer is that to win a match, a team needs to score more than the other team. Every team has one manager and a few coaches (decided by the manager), who come up with formations, strategies, training drills, etc., to increase the chance of winning. The English Premier League is the English soccer league that is at the highest order in the English soccer league system. It has been around for more than twenty years. The Premier League is often regarded the most exciting soccer league on the planet, with fierce competitions from the La Liga, which is the top-level League in the Spanish soccer league system. The English Premier League has 20 teams competing for the Championship. Every team plays each of the other 19 teams in the league twice per season, one at home and one away. This means that each team has to play 38 games per season.

B. Machine Learning

Machine Learning is the concept of learning from the data provided/given. It gives the computers the capability to learn without having to be set up manually, which means that the computer will determine the next steps to take from the past events/data. This idea is the same as the idea of how human learn from experiences (or mistakes) to make decisions in the future. The fundamental goal of machine learning is to generalize, or to induce an unknown rule from examples of the rule's application [12].

There are two main categories of Machine Learning: supervised learning and unsupervised learning. In supervised learning, a dataset is given as training data. The labeled training data is analyzed to create a function that can determine the next instances from examples of the right answers. Unsupervised learning, on the other hand, serves the purpose of attaining patterns in the data provided. This means that in unsupervised learning, the computer attempts to discover relationships in the data that is unlabeled.

There is one more category which is not as common as the two previous ones, it is reinforcement learning. Reinforcement learning can be considered the middle-ground between supervised learning and unsupervised learning. Reinforcement learning, or semi-supervised learning, takes in and uses both supervised and unsupervised data. A reinforcement learning program receives feedback for its decisions, but the feedback may not be associated with a single decision [12].

In sports in general, and in soccer in particular, data is usually carefully labeled. This means that data in soccer has some kind of meaningful tag, or label, such as number of shots, fouls, points, etc. The choice of machine learning category can also be looked at in more details from the cheat sheet that is provided by the Scikit-learn site (Figure 1). Therefore, the most common machine learning techniques in sports/soccer belong to the category of supervised learning.

Explained below are the four machine learning algorithms that are used in the project. These four algorithms are commonly used and suggested by researchers who have attempted to use machine learning for soccer prediction [4][5][7][8][9].

- Decision Tree:
  The way a decision tree works can be easily inferred from its name. A decision tree implements a tree-like graph of decisions and the consequences that come after those decisions.
  Each node in the tree specifies a test of some attribute of the instance, and each branch descending from that node corresponds to one of the possible values for this attribute [1]. Starting from the root, the attributes of the root will be examined. After that, each branch of the initial node will be another "root" for further attributes and branches and the previous process will be repeated. The decision tree algorithm will stop after all decisions have been made and the final result has been determined.
  The collection of a large number of decision trees can make up a random forest. This is an ensemble learning method for classification that operates by constructing a set of decision tree while training and then outputs the class which is the mode of the classes output by individual trees [6]. The random forest method's purpose is to avoid or overcome the habit of overfitting to the training set. Ulmer et al. used random forest because they believed that many results (from using Baseline, Naive Bayes, and Hidden Markov Model) did not correctly reflect the predicted matches due to the upsets in the data and thus, had high error rates [8].

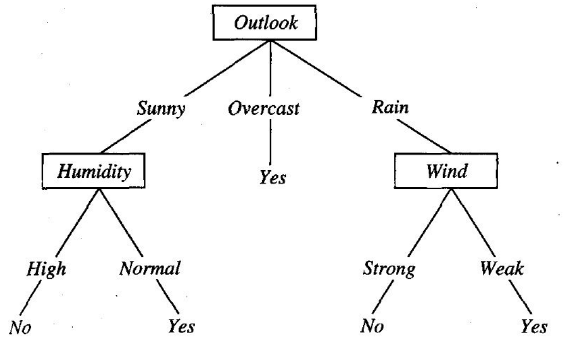

Figure 2: A decision tree for the concept PlayTennis [1].

An example is classified by sorting it through the tree to the appropriate leaf node, then returning the classification associated with this leaf (in this case, Yes or No). This tree classifies Saturday mornings according to whether or not they are suitable for playing tennis [1].

- Bayesian Network:
  A Bayesian network is a graphical probabilistic belief network that represents the conditional dependencies among uncertain variables, which can be both objective and subjective [5]. Bayesian networks provide a means for representing, displaying, and making the knowledge of experts in a given field available in a usable form [3].
  Looking into Bayesian learning methods, it is important to consider every training instance as each of those can increase or decrease the correctness of a hypothesis. Moreover, combining the newly examined data with the past observation is the way to establish a final and concrete evaluation of a hypothesis's correctness. From the hypotheses that have been examined/observed and evaluated, Bayesian networks will attempt to make predictions in a percentage format. New instances can be classified by combining the predictions of multiple hypotheses, weighted by their probabilities [1].
  The core and underlying factor of Bayesian networks is the Bayesian theorem, which supplies a calculation method of the posterior probability P(h|D). We need to know three pieces of information to make the calculation. P(h) is the prior probability of hypothesis h being a correct one. P(D) is the prior chance that training data D will be examined. P(D|h) is the probability of observing data D if hypothesis h holds [1]. With the three probabilities above, we can find the posterior probability is produced using the formula: $P(h|D) = \frac{P(D|h)P(h)}{P(D)}$ [1].
  Constantinou et al. presented a new Bayesian network model for forecasting the outcomes of soccer matches in the distribution form of {p(H), p(D), p(A)}; corresponding to home win, draw and away win [5].They made use of previous information (data from previous seasons), current information (data from the on-going season) and subjective information (expert's) to produce a non-symmetric Bayesian parameter learning procedure.
  One of the most common learner in the Bayesian learning methods is the naive Bayes learner. This is a simple probabilistic classifier based on applying Bayes theorem with strong independence assumptions [6]. This means that the naive Bayes classifier makes the assumption that all instances or characteristics are independent of each other in contributing to the probability.

- Logistic Regression:
  Logistic regression is used for classification. The goal in classification tasks is to find a function that maps an observation to its associated class or label [12]. The model is useful for problems in which the dependent variable is categorical.
  In logistic regression, the dependent variable does not have that infinite range. It only has a limited number of possible values. Instead of approximating the 0 and 1 values directly, thereby risking illegitimate probability values when the target is overshot, logistic regression builds a linear model based on a transformed target variable [2]. Moreover, in problems that require the dependent variables to belong to more than two possible outcomes, logistic regression is used. This method is called multinomial logistic regression.
  Multinomial logistic regression, provides another dimension for researchers in prediction problems. Timmaraju et al. used multinomial logistic regression as there were more than two possible outcomes [7]. Other researchers such as Buursma in and Tax and Joustra in also used LogitBoost, a boosting algorithm that is based on logistic regression [4][9].

- Support Vector Machine:
  Support vector machine is a powerful model for classification and regression. From a given training dataset that contains examples that are components of two different categories, support vector machines select one of the two categories to assign new instances. Support vector machines select a small number of critical boundary instances called support vectors from each class and build a linear discriminant function that separates them as widely as possible [2]. Then, the new instances will be mapped and to predict/decide which category they fit in, the only thing that needs to be considered will be the side of the gap to which they have been mapped. These systems transcend the limitations of linear boundaries by making it practical to include extra nonlinear terms in the function, making it possible to form quadratic, cubic, and higher-order decision boundaries [2].
  Although not many researchers have utilized the strength of support vector machine in sports pre-

diction, there has been evidence that support vector machine performs better than some other commonly used machine learning algorithms. Ulmer et. al. reported that support vector machine was one of the best performing models in their three-class classification problem, with lower error rate than that of the random forest as well as the naive Bayes models [8].

IV. Framework and Methodology

The framework for predicting soccer match results in the English Premier League include the following components: knowledge about the sport and the English Premier League, data understanding and preparation, feature extraction, training and testing, and finally, performance evaluation. This framework slightly follows the one proposed by Bunker and Thabtah [10].

A. Data Preparation and Understanding

Before going into learning Machine Learning techniques as well as deciding which methods to use, researchers have to get adequate understandings of how to obtain the data and what each set of data describes. There are a few things to be considered when you retrieve the data, which include the validity of the source, the correctness of the data, making sure that the data do not get lost during retrieving process, etc. After the right source has been found and the data have been retrieved, it is essential that you have good knowledge of what the data represent, or in other words, what they tell about the subject, in this case, soccer matches.

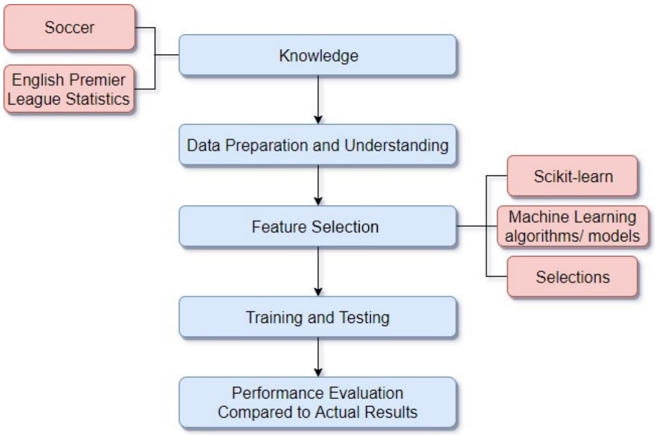
Soccer
English Premier League Statistics
Knowledge
Data Preparation and Understanding
Scikit-learn
Feature Selection
Machine Learning algorithms/ models
Selections
Training and Testing
Performance Evaluation Compared to Actual Results

Figure 3: Framework of the English Premier League match results prediction. This includes Knowledge about soccer and the English Premier League, Data Preparation and Understanding, Feature Selection, Training and Testing, Evaluation - Compared to Actual Results.

1) Data Preparation:

Data Preparation includes finding the appropriate data source as well as cleaning and preprocessing the data. The data may need cleaning and preprocessing as even if the data source is considered trustworthy, there can still be minor errors and inconsistent formatting.

Data Source:

Detailed statistics for specific players are very important in determining teams' strengths and values. Certain "key" players can turn the matches on their heads and bring their teams the more benefiting results. However, the data providers do not want to make this type of statistics public. Thus, in this project, results prediction is made from data of past results.

Data source provided by Football-Data site [11] is highly used for similar projects in the past [4][5][7][8][9]. The database contains all full-time and half-time score lines and the bookmaker odds for most of the matches [4]. The site provides data for matches in the English Premier League since season 1993 - 1994. However, from season 1993 - 1994 to season 2000 - 2001, the data only contain half-time results and full-time results, which are inadequate for the usage of Machine Learning. From season 2000 - 2001 until the current season, the site provides data with much more diversity than before. The data contains the number of shots, shots on target, fouls, corner kicks, cards, etc., which will be listed and explained in the Data Understanding section below. Thus, this data source is the perfect fit for the purpose of applying Machine Learning techniques to find patterns and features that may affect the future match outcomes. The data are provided in the form of CSV files.

Data Cleaning and Preprocessing:

The data can be easily read in Python using the pandas package as they are provided in the form of CSV files. However, any set of data can have errors that affect the solution. Thus, the cleaning process is very important. This dataset in particular contains a few rows of data that are NaN (not a number) values. These needs to be dropped as they can not be read by the machine learning algorithms. In the data, there are also numerous bad lines, which are lines that had too many fields (commas). Moreover, the data contains categorical variables, such as team names. While the system should predict the results in three categories, home wins, ties, or away wins, it should not be the case that any of the features be categorical. Hence, conversions from any categorical variables into integers have to be made by using Scikit's LabelEncoder. However, Scikit's algorithms and estimators may consider the integer values as ordered values, which is not the case for the team names. For example, a pair of teams are assigned values 0 and 1 may be considered as more similar than a pair of 0 and 5. Thus, those x possible values must then be turned into x binary features (with only one of them being active) with Scikit's OneHotEncoder.

2) Data Understanding:

Using Machine Learning techniques on public league data to predict results in soccer requires one to have more than just the basic knowledge above. Based on the personal knowledge about soccer, below are some explanations of the statistics that are provided in the data that is used in this project:

- Home Team/ Away Team: The home team gets to play on home field and gets more attendance at the game than the other team. On the other hand, the away team has to travel to the stadium of the other team to play. This can be described as an advantage for the home team as they get more mental encouragement from the home supporters. Thus, this plays an important role in predicting match outcomes [9].
- Total shots: This number is the record of the number of attempts to score of each team. The number of shots can tell which team attacks more, which may mean that that team is the better side in the match.
- Shots on target: Any goal attempt that goes into the net, would have gone into the net but for being stopped by a goalkeeper's save or would have gone into the net but for being stopped by a defender who is the last man [6].
- Corner kicks: A corner kick is a set-piece/free-kick taken from a corner of the field. The number of corner kicks may tell if a team is attacking a lot or not as they are taken on the opponent's half of the field.
- Fouls: Any infringement that is penalized as foul play by a referee [6]. This statistic can be interesting as it can reflect whether a team plays with a defensive mentality or not. The team that fouls more in a match (by a significant amount) is usually the team that plays defensively in order to to avoid losing to the opponent and thus, get at least one point.
- Yellow cards: A yellow card is usually given after a foul as a warning. This is decided by the referee.
- Red cards: A red card, like a yellow card, is usually given after a foul. A straight red card can be given to a player if the foul is very bad. Two yellow cards also result in a red card. A player who gets a red card will be removed from the match and his/her team will have to play with one fewer player for the rest of the game. This is a huge disadvantage in soccer.
- Results: Soccer results at league levels can be divided into three types. The first one is the home team wins. The second one is the away team wins. The final one is a tie, in which both teams score the same amount of goals, which can also be zero.

B. Feature Selections

1) Scikit-learn - The Tool:

Nowadays, there are a lot of tools that have been developed for the purpose of analyzing statistics and Machine Learning. A few outstanding softwares to be mentioned are The Shogun Machine Learning Toolbox and Waikato Environment for Knowledge Analysis (WEKA). The Shogun Machine Learning Toolbox does not provide as many common models and model-tuning, meta-methods as WEKA. On the other hand, WEKA provides users with several graphical user interfaces that make it easier for the users to use. However, this feature is also a disadvantage. This is because with the main usage being clicking and dragging instead of purely coding, it is hard to automate the whole process of machine learning as the users would have to choose everything manually. Thus, WEKA is not the best choice for this type of problem. On the contrary, Scikit-learn is perfect for the automation of the process.

Scikit-learn is a Python module integrating a wide range of state-of-the-art machine learning algorithms for medium-scale supervised and unsupervised problems [14]. The most important characteristic of the Scikit-learn is that it provides the users with the algorithms for machine learning tasks including classification, regression, dimensionality reduction, and clustering. The Scikit-learn official site also help users get a clearer ideas of what to do/which algorithms to choose according to their dataset and purposes by equipping a detailed flowchart, which is called Scikit-learn algorithm cheat-sheet. This cheat-sheet can be seen in figure 1 above. It also provides modules for extracting features, processing data, and evaluating models [12]. With Scikit-learn, creating scripts to run many lines of code that carry out the whole process of machine learning becomes easier.

2) Machine Learning Algorithms/ Models:

Recall from section III, this project uses a total of four machine learning algorithms: Decision Trees, Bayesian Networks (Gaussian Naive Bayes to be more specific), Logistic Regression, and Support Vector Machine. These four algorithms are all provided in the Scikit-learn module. The models from the algorithms can also be easily created thanks to Scikit-learn. The modeling process includes four main components, the set of selected features, fitting the model of the imported algorithm/estimator with the training data to make predictions, the training and testing process and finally, the calculation of the accuracy.

3) Selections:

After gaining basic understandings of the data as well as the machine learning algorithms/models, we have to carefully consider the selections in the dataset that will best serve the purposes. An accurate feature set makes it a lot easier to predict the outcomes of matches [4].

One way to select the features or statistics that can have positive influence on the accuracy of the predictions in this problem is to follow the basic understandings of the data in soccer. For example, as explained the Data Understanding section, home advantage (or away disadvantage), the goals scored and goals conceded can be considered deciding factors in the results of soccer matches.

Another way to approach the process of selections is to create feature subsets. Features in sport result data can be divided into several different subsets [10]. This can be done by separating match statistics and performance-related data. Match statistics include features such as the number of shots, goals, corners, cards, etc. Performance-related features may contain the points gained by each team in the last x matches, their standings on the table, etc. In this project, we will be looking at the performance of the algorithms when used with match statistics. The starting set of features is:

- Home Team
- Away Team
- Half-time Home Goals
- Half-time Away Goals
- Home Shots
- Away Shots
- Home Shots on Target
- Away Shots on Target
- Home Corner-kicks
- Away Corner-kicks
- Home Fouls
- Away Fouls
- Home Yellow Cards
- Away Yellow Cards
- Home Red Cards
- Away Red Cards

However, the selection process must not stop here. This set needs to be checked and based on the results, the list of features to be selected must be changed. This process needs to be done again and again so that the best features can be chosen. In this project, we will try to find the best set of features simply by adding and removing features to see the differences in the accuracy. The strategy to remove and add features is to do it in pairs. From the starting set of features, it can be easily seen that the set is a collection of pairs of features. There are 14 features, which can actually be divided into 7 pairs of items: teams, half-time goals, shots, shots on target, corner-kicks, fouls, yellow cards, and red cards. Each time we want to test a different set of features, we can make a different combination from any of the 7 pairs. The accuracy from different sets of features can be calculated from the training and testing process. If the removal of a pair increases the accuracy, we can keep testing other features without that pair of features. On the contrary, if the removal of a pair decreases the accuracy, we know what those features are essential to the model and thus, should be kept as part of the final selection of features.

C. Training and Testing

After an initial set of selections had been chosen/attained, the training and testing process can be executed. The procedure during which the machine learning algorithm calculate the probabilities for the matches is called training [4]. It is important to preserve the order of the training data for the sport prediction problem, so that upcoming matches are predicted based on past matches only [10]. Normally, a large percentage of the data will be used for training while the remaining data will be used for testing. In this project, the percentage of data used for testing is 10% as it is common that the more data the algorithms have to train with the better they perform. The data for training and testing can be easily split by using Scikit's train_test_split function with parameter test_size set to the wanted value (0.10 in this case).

D. Performance Evaluation - Compared to Actual Results

After all the results have been calculated and predicted, we will need to evaluate the performances of the algorithms, which make up the core of the framework. This is done by determining the outcomes of the matches based on the predictions of the algorithms and comparing those to the results in real life. This is actually also a part of training and testing. It can done by using Scikit-learn's accuracy_score function, which takes in the predicted values and the classifications from the testing set to return the accuracy of the predictions.

It is possible that a different set of features can work better with one algorithm than with the others. That is why the results of the Training and Testing process are used to determine the efficiency of the algorithms. From there, the best set of features and the best algorithm can be determined. The models can be evaluated as successful if their accuracies are more than 60%, as reasonable if they can predict more than 50% of the games right, and as totally unsuccessful if they cannot get more than 50% of the results right.

1) Results:

Below is the final (best) set of features:

- Home Team
- Away Team
- Half-time Home Goals
- Half-time Away Goals
- Home Shots on Target
- Away Shots on Target

With the set of features above, the algorithms achieve the following accuracies:

Algorithms	Accuracies
Gaussian Naive Bayes	34.64%
Decision Tree	60.21%
Support Vector Machine	61.57%
Logistic Regression	67.62%

Figure 4: Table of Results.

The algorithm that had the lowest accuracy was Gaussian Naive Bayes while the algorithm that had the best accuracy was Logistic Regression.

2) Discussions:

As stated above, Gaussian Naive Bayes was the least successful method out of the four algorithms in this project. The reason for this can be soccer being a game in which many things are related each other. For example, the number of shots on target may relate to the number of goals scored by a team, or the number of fouls may relate to the number of yellow cards and red cards. Moreover, soccer has an unclear characteristic that can be considered as streak. A team that has a winning streak tends to do well and get a positive result in the next game. Similarly, a team that has a losing streak tends to get a negative result in the following match. With all these relations, Gaussian Naive Bayes performing poorly at soccer prediction is not a surprising result. This is because Naive Bayes treats all instances or characteristics as independent of each other and disregards any relationships between the features. This is not the case in soccer and thus, gave a totally unsuccessful result.

The other three algorithms all performed reasonably well. While the differences were not huge, Logistic Regression is noticeably the best algorithm for the problem of soccer prediction with the specified feature set. Logistic Regression performed slightly better than Decision Tree because Decision Tree algorithm often needs large dataset. Otherwise, Decision Tree may lead to misleading/wrong results. On the other hand, for a relatively small dataset as the one in this project, Logistic Regression can give out better results, hence the slight difference in the accuracies between the two algorithms. To explain the lower accuracy of Support Vector Machine, we have to look at the dimension of the data. Support Vector Machine performs better with high-dimensional spaces while Logistic Regression is the better choice for lower number of dimensions. Data that is considered to have high-dimensional spaces has more features than data points, which is not the case for the above set of features. We have a total of 6 features while the number of data points we have can go up to a few thousands. Therefore, while all three algorithms perform well in this problem with the certain feature set, Logistic Regression still has the highest accuracy.

V. Conclusions and Future Work

In conclusion, a detailed framework for using machine learning algorithms to predict the match results in the English Premier League has been illustrated in this project. This problem is only one of many problems in the prediction domain, which is on the rise. In this project, The game-play statistics such as number of shots, shots on target, corner kicks, cards, etc. were explored. Out of fourteen different features, the ones that made up the best combination were home team, away team, half-time home goals, half-time away goals, home shots on target, and away shots on target. Moreover, the best performing algorithm is Logistic Regression. In addition to Logistic Regression, Decision Tree and Support Vector Machine also performed well. However, Gaussian Naive Bayes had a significantly lower accuracy.

Currently, I am only using the game-play statistics in this project. This means that predictions can only be made on existing matches. Thus, to improve the model in the future, I will need to manipulate the data and extract additional features such as the cumulative points gained and goals difference (goals scored - goals conceded) of a team during the current season to actually predict future results. Moreover, in order to turn the framework into a working product, I plan to create a simple interactive program that can predict the results of the match given the names of the home team and the away team, or the results of one team for the whole season. Another possible direction for the future is to predict the exact scores of the matches instead of categorizing the results as only home wins, ties, and away wins. Furthermore, this particular soccer prediction framework can be the foundation for other frameworks used to forecast results in different sports.

VI. Acknowledgements

This work was supported by the Earlham College Computer Science Department as part of the Senior Capstone Experience Course. Special thanks go to Charlie Peck who, as my Methods of Research and Dissemination Course instructor, led me in the right direction to determine the best topic for me as well as the necessary tools for the project. I would also like to thank David Barbella for his guidance and instructions as my topic advisor, which has kept me on track to complete this interesting Senior Capstone project.

References

[1] Mitchell, Tom M. *Machine Learning*. First edition, McGraw-Hill Education, 1997.

[2] Witten, Ian H., and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition*. Morgan Kaufmann, 2005.

[3] Joseph, Anito, et al. "Predicting Football Results Using Bayesian Nets and Other Machine Learning Techniques." Knowledge-Based Systems, vol. 19, no. 7, 2006, pp. 544-553.

[4] Buursma, D. "Predicting Sports Events from Past Results Towards Effective Betting on Football Matches." Conference Paper, Presented at 14th Twente Student Conference on IT, Twente, Holland, vol. 21, 2011.

[5] Constantinou, Anthony C., et al. "Pi-Football: A Bayesian Network Model for Forecasting Association Football Match Outcomes." Knowledge-Based Systems, vol. 36, 2012, pp. 322-339.

[6] Kumar, Gunjan. *Machine Learning for Soccer Analytics.* Cambridge University Press, MSc thesis, KU Leuven, 2013.

[7] Timmaraju, Aditya Srinivas, et al. *Game ON! Predicting English Premier League Match Outcomes.* 2013.

[8] Ulmer, Ben, et al. *Predicting Soccer Match Results in the English Premier League.* Ph. D. dissertation, 2013.

[9] Tax, Niek, and Yme Joustra. "Predicting The Dutch Football Competition Using Public Data: A Machine Learning Approach." Transactions on Knowledge and Data Engineering, vol. 10, no. 10, 2015, pp. 1-13.

[10] Bunker, Rory P., and Fadi Thabtah. "A Machine Learning Framework for Sport Result Prediction." Applied Computing and Informatics, 2017.

[11] Football Betting — Football Results — Free Bets — Betting Odds. http://www.football-data.co.uk/. Accessed 9 Nov. 2017.

[12] Hackeling, Gavin. *Mastering Machine Learning With Scikit-Learn*. Packt Publishing, 2014.

[13] *Scikit-learn: Machine Learning in Python.* http://scikit-learn.org/stable/. Accessed 16 Nov. 2017.

[14] Pedregosa, Fabian, et al. "Scikit-Learn: Machine Learning in Python." Journal of Machine Learning Research, vol. 12, Oct. 2011, p. 28252830.