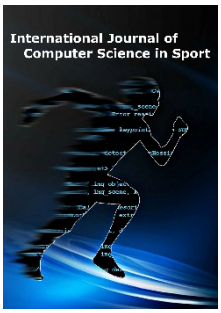
International Journal of Computer Science in Sport

Volume 19, Issue 2, 2020

Journal homepage: http://iacss.org/index.php?id=30

sciendo

DOI: xxx



A Critical Comparison of Machine Learning Classifiers to Predict Match Outcomes in the NFL

Ryan Beal, Timothy J. Norman and Sarvapali D. Ramchurn

University of Southampton, University Rd, Southampton, SO17 1BJ

Abstract

In this paper, we critically evaluate the performance of nine machine learning classification techniques when applied to the match outcome prediction problem presented by American Football. Specifically, we implement and test nine techniques using real-world datasets of 1280 games over 5 seasons from the National Football League (NFL). We test the nine different classifier techniques using a total of 42 features for each team and we find that the best performing algorithms are able to improve one previous published works. The algoriothms achieve an accuracy of between 44.64% for a Guassian Process classifier to 67.53% with a Naïve Bayes classifer. We also test each classifier on a year by year basis and compare our results to those of the bookmakers and other leading academic papers.

KEYWORDS: MACHINE LEARNING, SUPERVISED LEARNING, FOOTBAL, NFL

Introduction

The prediction of match outcomes across all sports has long been a challenge that researchers and punters across many fields have aimed to solve. Early works by Harville (1977) and Dixon & Coles (1997) show statistical models to predict American Football and Soccer respectively and set a high benchmarks that are challenging to improve on. This is likely due to the many uncertainties that exist when trying to predict the outcomes of teams of humans competing against each other at the highest levels. Any model that aims to make a prediction of this must consider many of these factors, these include: team strength, player configurations, health of players, location of the match (home or away), the weather, and team tactics.

In recent years, many domains (including sport) have seen the improvement of artificial intelligence (AI) techniques and many of the applications of AI in team sports are discussed in Beal, Norman, & Ramchurn (2019). These new approaches allow algorithms to learn as they are fed more data and they are able to interpret more of the uncertainties that need to be considered in team sports. This can make AI a key tool for the match outcome prediction problem.

The prediction of match outcomes is key to many stakeholders in sports. It can help teams select their tactics as well as bookmakers to set odds and punters to place their bets. The sports betting market is a huge worldwide industry with the most common bets being placed on match outcomes. The worldwide sports gambling market is projected to grow to $565 Billion by 2022.[1] In American Football punters not only bet on match outcomes but also bet on the points spread, which is also touched on in this paper.

In this paper, we aim to evaluate how a set of supervised machine learning classification techniques perform when predicting match outcomes in American Football. We specifically focus on 5 seasons (2015-2019) in the NFL. This comparison allows us to identify which machine learning technique is best for this problem as we use a consistent, comprehensive feature set to test across all techniques. The feature set is also scraped from online freely available open-source data, therefore allowing others to reapply and improve on the work in this paper. We give real-world benchmarks for new work to compare against while also identify the top techniques to use.

Against this background, we give a comparison of nine machine learning techniques over the five-season period. Our work advances the state of the art in the following ways:

1. We have mathematically defined the machine learning problem of match outcome prediction for American Football.
2. We present a novel application and comparison of nine well known machine learning classification techniques and how they perform when predicting match outcomes.
3. We find that we are able to achieve an accuracy of up to 67.53% across 5 seasons of real-world data from the NFL. This sets a new baseline for match outcome prediction in the NFL.

Our results show the varying abilities of machine learning models to predict match outcomes results for the NFL, this leads us to a discussion on how these can be used and how the results can be improved in the future. Our results provide a new accuracy baseline when predicting match outcomes in the NFL.

[1] https://www.businesswire.com/news/home/20190606005537/en/Global-Gambling-Market-Reach-565-Billion-2022.

The rest of this paper is structured as follows. In Section 2 we give a background to American Football and the literature. In Section 3 we define the problem and in Section 4 we outline our feature set. Next, in Section 5 we discuss the machine learning methods tested in this paper and in Section 6 we run experiments on these. Finally, in Section 7 we discuss our findings and Section 8 we conclude.

Background

In this section we provide a background to the research presented in this paper. Firstly, we give a brief description of the game of American football and its rules. Next, we discuss the past literature for outcome prediction across all sports and the leading papers. Finally, we specifically discuss the current approaches to match outcome prediction in American Football and the NFL.

Overview of the Game

American Football is mainly played in the US and Canada and the main professional league is the National Football League (NFL) in the US. The NFL contains 32 teams from across the country who each play 16 games across a regular season. The top teams then go on to compete in a play-off tournament, the final of this is called the SuperBowl with the winner of this being the overall winner of the league that season. There is also a strong following of college football in the US with teams competing in the NCAA (National Collegiate Athletic Association) league. American Football makes up an estimated 13% of the global sports market.[2]

In American Football, teams aim gain yards and to score more points than their opponents. Points can be scored in a number of ways: a touchdown is worth 6 points this is followed by an extra play worth 1 point for a field goal or 2 points for another score, a field goal in normal play is worth 3 points and a safety is worth 2 points. The game is played over 60 minutes over four quarters of 15 minutes each with 11 players on each team on the field at one time from a squad of 45. The average scoring frequency is a score every 9 minutes (Beal, Norman, & Ramchurn, 2019).

Sports Outcome Prediction

Predicting sports match outcomes is a complex problem that must factor in a number of considerations about two teams made up from human players. These include but are not limited to: team strengths, team tactics, player moods/morale, player health, team form, location of the game and the weather conditions. Sports match outcomes can usually be classified into 3 classes home win, draw/tie and away win (although in many sports, such as American Football, the draw/tie is an increasingly rarer outcome). There are many examples of works across many different fields (e.g., mathematics, statistics, economics, computer science and AI) that aim to predict match outcomes. Many of these works also look at the score of the game and points scored by each team (referred to as points spread by bookmakers). There is an in-depth review of the work and issues in this domain discussed in detail in Beal, Norman, & Ramchurn (2019).

Early examples of statistical approaches of this are shown in soccer by Maher (1982), which describes an initial model to assign probabilities to each game outcome (home win, draw and away win). This was built on in Dixon & Coles (1997), which is still one of the leading models for prediction in soccer. One significant improvement on Maher was the use of home team

[2] https://medium.com/sportyfi/how-big-is-the-sports-industry-630fba219331.

advantage which is discussed in Clarke & Norman (1995), here the value of home advantage is calculated. Other improvements on these models are shown in Dixon & Robinson (1998) and Crowder, Dixon, Ledford, & Robinson (2002).

There are also a number of examples that aim to improve on the baselines set by Dixon & Coles (1997) for soccer by using artificial intelligence techniques such as Joseph, Fenton, & Neil (2006) who use Bayesian methods to make their predictions. Constantinou, Fenton, and Neil (2012) test their model across two seasons of the English Premier League showing some promising results. Their model considers variables such as team strength, form, psychology and health of players. Joseph, Fenton, and Neil (2006) also test a decision tree and a K-nearest neighbor model.

As well as the works described for soccer, there are many notable papers for other sports such as Yang & Swartz (2004) using a two-stage Bayesian model for Baseball, Sankaranarayanan, Sattar, & Lakshmanan (2014) uses data-mining and machine learning to predict cricket games and (McCabe & Trevathan, 2008) uses Neural Networks to predict games of Rugby League. In the following section we go into more detail regarding the current literature for match outcome prediction in American Football.

American Football Prediction

Turning to American football, (Harville, 1980) presents a birth-process statistical model to predict NFL games. This work uses a linear approach to create a baseline for NFL predictions in American Football, building on work that was originally tested on college and high school American Football (Harville, 1977). More recently, Boulier & Stekler (2003) evaluate the use of "Power Scores" (published in the New York Times) in the NFL between 1994-2000 and find that their model was able to improve on the accuracy of the predictions made by human experts. However, it was not able improve on the bookmakers' accuracy.

Moving away from statistical approaches, there are a number of applications of AI techniques to predict NFL games. Work in Glickman & Stern (1998) uses a Bayesian state-space model (tested on 1993 NFL season). Their main focus is on predicting the points spread but by doing so also predict the match outcomes. They produce good results for this when compared against the "Las Vegas betting line" but were unable to outperform it. They achieve an accuracy of 58.2% whereas the Las Vegas accuracy (at the time) was 63%. There is also a prediction method presented in Landers & Duperrouzel (2018) which is applied to "Pick'em" style online competitions. Their model uses 28 features with an average perceptron and a boosted decision tree classifier algorithm.

They test their model over three NFL seasons and find the decision tree gives the most accurate average accuracy at 58%. This is compared to Boulier & Stekler (2003) which achieves 61% and to the bookmakers who achieve 65.8% accuracy. These accuracies give good baselines to compare the results from this paper to. In the next section, we define the problem of NFL match outcome prediction.

Problem Definition

In this section we mathematically define the problem of predicting the NFL match outcomes. This will set out the basis for our prediction techniques that are described in Section 5 and define the problem for future papers.

Match Outcomes

Here we define the match outcome problem that we aim to solve with our machine learning techniques. As discussed in Section 2 both teams are aiming to score points against one another with the team with the most winning the game. The outcome of the game can be separated into 3 possible classes: the home team winning, the away team winning and a tie. The tie in NFL games is much less likely as if the game ends the fourth quarter with both teams on equal points the game goes into overtime (OT) to help end in result (since 2012 there have been just 8 ties).[3] Thereofore, in this paper we do not aim to predict ties and just consider two outcome classes.

We find a probability of each of the possible scorelines that a game will end in, then sum to find the overall probability for each outcome class. This is broadly discussed in (Beal, Norman, & Ramchurn, 2019) for all sports match outcomes, we define this for the NFL problem below:

$$p(outcome) = \sum_{i \in \mathbb{Z}} \sum_{j \in \mathbb{Z}} p(H = i, A = j)$$
$$outcome = \begin{cases} homewin\ if\ H > A \\ awaywin\ if\ H < A \end{cases} \quad (1)$$

Where, H is the home points scored and A is the away points. Here, we sum the probabilities of each possible scoreline for each of the outcome classes (defined using H and A). To solve the problem that we have defined we collect a set of features X that help to quantify some of the key drivers of variables that impact of the outcome of an NFL game. These features will be discussed in detail in Section 4. Using this feature set we can train a machine learning model that is trained to classify the features into possible outcomes $y=\{homewin, awaywin\}$. This can be defined as $y = \phi(X)$, where y is the predicted outcome, ϕ is the prediction model used and X is our feature set. When training our machine learning models, we will use a training set Y of actual match results for the features in X. We discuss the machine learning techniques we test in this paper in Section 5.

In the next section, we highlight the features that are used in our models and how these are calculated from the historic datasets.

Feature Set

In this section we discuss the features that make up the feature set X we use in the machine learning models described in this paper. For all of the features that we discuss, we will use both the recent numbers (current season average up to that game) as well as the historic numbers (the average across the most recent completed season). This means that overall, we have 42 features for each team. Below, we provide the title of the feature, its type and a brief description of what it represents or how it is calculated.

- **Points Scored:** Total number of points that a team has scored in the season.
- **Yards Gained:** Total number of yards that a team has gained throughout the season.
- **Offensive Plays:** Total number of offensive plays a team has run in a season.
- **Possession Lost:** The combined number of offensive fumbles and turnovers in a season.

[3] https://operations.nfl.com/the-rules/nfl-overtime-rules/.

- **Passing Completions:** Total number of completed passing plays.
- **Passing Yards:** Total number of passing yards gained in a seasons.
- **Passing Touchdowns:** Total number of passing touchdowns in a season.
- **Rushing Completions:** Total number of completed rushing plays in a season.
- **Rushing Yards:** Total number of rushing yards gained in a seasons.
- **Rushing Touchdowns:** Total number of rushing touchdowns in a season.
- **Expected Points Scored:** Expected offensive points from the season.
- **Points Conceded:** Total number of points that a team has conceded in the season.
- **Yards Conceded:** Total number of yards that a team has lost in the season.
- **Defensive Plays:** Total number of defensive plays that a team has faced in a season.
- **Possession Gained:** The combined number of defensive fumbles and turnovers that have been caused in a season.
- **Passing Yards Conceded:** Total number of passing yards lost in a seasons.
- **Passing Touchdowns Conceded:** Total number of passing touchdowns scored against the team in a season.
- **Rushing Yards Conceded:** Total number of rushing yards lost in a seasons.
- **Rushing Touchdowns Conceded:** Total number of rushing touchdowns scored against the team in a season.
- **Expected Defensive Points:** Expected defensive points conceded in the season.
- **Extra Points Made:** Total number of extra points made in a season though field goals and point after touchdown (PAT) attempts.

As well as these for each team, we will also use a home advantage coefficient which will value the advantage that a team gains from playing at home. This is calculated using the techniques discussed in Clarke & Norman (1995). Meaning altogether the feature set X contains 85 features (42 regarding the performance of each team and one feature for home advantage).

In the next section we discuss the details of the machine learning classification techniques that we use with these features to predict the NFL match outcomes.

Machine Learning Classification Techniques

In this section we provide the details of the supervised machine learning classification methods that we test in this paper. For each method we give a background of how it will be used and the mathematical notation of its application to our features X and training set Y.

Support Vector Machines (SVM)

A support vector machine (SVM) algorithm (Suykens & Vandewalle, 1999), is used for two-group classification problems (e.g., home win or away win). An SVM uses training data to fit a hyperplane in the feature set that acts a decision boundary between the two classes. The hyperplane is fit by maximising the margins between the different classes. With SVM's different kernels can be selected to help learn the hyperplane decision boundary. In this paper

we have focused on using a radial basis function (RBF) kernel.[4] An RBF kernel as discussed in (Han, Qubo, & Meng, 2012), fits a decision boundary based on the distance a point is from a number of centres. The formula for the RBF model is shown below.

$$f(x) = \sum_{j=1}^{M} \lambda \emptyset(|x - m_j|) \quad (2)$$

Where, f(x) is our match outcome class prediction, M is the number of centres, λ is the weight for each centre, m_j is the centre point and x is the feature set for a given game.

Nearest Neighbours

The nearest neighbours classifier algorithm (Cover & Hart, 1967), classifies new samples by calculating the distance to the nearest training case. It then makes a prediction based on what classification that training case was. To calculate the distance to the nearest points the Euclidean Distance Formula (Danielsson, 1980) is used. This is as follows:

$$dist(p, q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \quad (3)$$

Where, p and q are two sets of features (x) corresponding to a game. This method is simple and can be effective, however, the algorithm gets significantly slower as the number of examples and/or predictors/independent variables increase.

Gaussian Process

A Gaussian process (Bonilla, Chai, & Williams, 2008) is a non-parametric classification method based on a Bayesian methods. It assumes some a distribution on the underlying probabilities and the classification is then determined as the one that provides a good fit for the observed features, while at the same time guaranteeing smoothness. In a Gaussian process, any point $x \in \mathbb{R}$ is assigned a random variable f(x) where the joint distribution of these is Gaussian. This gives the equation below.

$$p(f|X) = \mathcal{N}(f|\mu, K) \quad (4)$$

Where, $f = \big(f(x_1) \dots f(x_n)\big)$, $\mu = (m(x_1) \dots m(x_n))$ (m is the mean fucntion and m(x)=0) and $K_{ij} = k(x_i, x_j)$. Therefore, a Gaussian process is a distribution over functions whose shape is defined by K and if features x_i, x_j are similar then their prediction output $f(x_i)$ and $f(x_j)$ is also similar.

Decision Tree

The decision tree classifier (Breiman, Stone, & Olshen, 1984) uses a series of nodes and edges to determine a number of questions that lead to a given class (in our case match outcome) at a leaf node. This can therefore make a model easy to interpret and be used in a white box model to better understand what is happening. To select the order that attributes are in the tree certain criteria are calculated. Examples of these include entropy, information gain, gain ration and the Gini Index (shown below).

[4] Other kernels were tested in Section 6 and RBF gave the highest accuracy.

$$Gini = 1 - \sum_{i-1}^{C} (p_i)^2 \quad (5)$$

where p_i is the proportion of samples that belongs to class C for a given node.

Random Forest

A random forest model from (Breiman L. , 2001) is formed with a collection of different tree predictors where x is the feature set for a given game, $h(x, \Theta)$ is the individual tree's output and Θ is a random vector generated, independent of the past random vectors but with the same distribution.

$$f(z) = \frac{\sum_{k=1}^{K} h(x, \theta_k)}{F} \quad (6)$$

The outcome prediction f(x) is given by taking an average of the collection of tree predictor outputs. In the equation, F is the number of trees in the forest.

AdaBoost

Adaptive Boosting aka AdaBoost (Freund, Schapire, & Abe, 1999) helps to combine multiple weaker classifiers into a single stronger classifier. AdaBoost works by putting more weight on difficult to classify instances and less on those already handled well. The final equation for classification can be represented as below.

$$f(x) = sign(\sum_{m=1}^{M} \theta_m f_m(x)) \quad (7)$$

Where, f(x) is out match outcome prediction, f_m is the m^{th} classifier and θ_m is the corresponding weight.

Naive Bayes

A Naive Bayes classifier (Rish, 2001) is a machine learning technique based on Bayes theorem. Using Bayes theorem, we can calculate the probability of something happening given something else. Therefore, this can be modified to predict the probability of an outcome given the feature set. This can be written as the equation below.

$$p(y|x) = \frac{P(x|y)P(y)}{P(x)} \quad (8)$$

Where, y is the match outcome class and x is our features for this game. We also must consider two assumptions with this classifier. Firstly, that the feature predictors are independent and secondly, that all the predictors have an equal effect on the outcome.

Quadratic Discriminant Analysis (QDA)

QDA (Srivastava, Gupta, & Frigyik, 2007) is used to find a non-linear decision boundary between classifiers. In a QDA we assume that the covariance matrix can be different for each class and so, we will estimate the covariance matrix Σ_k separately for each class in y. We calculate the quadratic discriminant function in the first equation below and define the classification rule as the second equation.

$$\delta_{y(x)} = \frac{1}{2}\log|\Sigma_y| - \frac{1}{2}(x - \mu_y)^T \sum_y^{-1} (x - \mu - y) + log\pi_y \quad (9)$$

$$f(x) = \underset{y}{arg\max}\, \delta_y(x) \tag{10}$$

Here we aim to find class y (home or away win) which maximises the quadratic discriminant function δ to give the prediction f(x).

Neural Networks

Neural networks (Beale, Demuth, & Hagan, 1996) are a machine learning technique modelled to solve problems in a similar way to how the human brain works. They can recognise patterns in feature data to classify match outcome prediction. Neural networks are structured with an input layer, hidden layers and output layers (shown in Figure 1).[5]

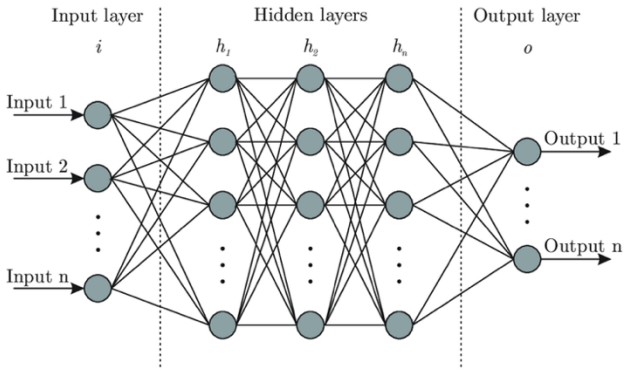


Figure 1. Neural Network Structure

We feed our feature data through the input layers and hidden layers and a prediction is outputted. Neural networks are trained by using error back propagation (Bohte, Kok, & La Poutre, 2002), where the internal parameters in the network are updated based on the output error in comparison to training examples in a labelled dataset. The final classification can then be given by the equation below.

$$f(x) = b + \sum_i w_i x_i \tag{11}$$

Where, b is the bias and each weight w_i is learnt with the back propagation for the corresponding features x_i.[6]

Experiments

In this section we outline the experiments that we have run to compare the performance of the machine learning classifiers that we have discussed in Section 5. We evaluate the accuracy of the models overall as well as across each of the 5 most recent NFL season (1280 games in total). We then compare against the leading published work in the area and the bookmaker's accuracy. All tests in this section are run using data collected from *pro-football-reference.com* and we use the SciKit-Learn Python library for our machine learning models.[7]

[5] https://www.kdnuggets.com/2019/11/designing-neural-networks.html.

[6] It is worth noting that in this paper we test feed-forward neural networks although there are a number of different types of neural network techniques that could be tested (eg., recurrent, convolutional and LSTMs).

[7] https://scikit-learn.org/.

Experiment 1: Overall Accuracy

In our first experiment we implement and test the machine learning methods from Section 5 using the feature set discussed in Section 4 complied from NFL data across the seasons between 2015-2020. We focus on the accuracy of each model which in this case is the share of correctly predicted outcomes. Our preliminary results showed that some of the test models may show signs of over-fitting. Particularly, SVM and Gaussian process methods where they show to be fitted perfectly for the training data. However, when for the test data-set they have much lower results in comparison to methods with lower training scores. The neural network seems to underperform which may be due to the smaller dataset as usually neural network and other deep learning methods require very large datasets. In terms of test scores, the decision tree method performs best with a test accuracy of 65.76% and this is tested further in the next table of results where we further explore the model accuracies by looking into their results in more detail.[8]

For each of the methods that we test, we randomly divide the historic dataset using a train-test split of 70% to 30% with a cross-validation approach for 10 folds. The results from this are shown in Table 1 where the standard deviation is show for the results across the 10 folds in the cross-validation approach . We show the accuracy, precision, recall and f1-score as well as their variations from the 10-fold test. The results show that Naive Bayes is the best performing technique with an accuracy of 67.53% and an f1-score of 0.67. AdaBoost and Random Forest also both performed well with accuracies of 66% and 64% respectively. These will be tested further in following experiments.

ML Method	Accuracy	Precision	Recall	F1-Score
SVM RBF	0.5537 (±0.11)	0.2769 (±0.06)	0.5 (±0)	0.3556 (±0.05)
Nearest Neighbours	0.5748 (±0.07)	0.5683 (±0.08)	0.5677 (±0.08)	0.5668 (±0.08)
Gaussian Process	0.4464 (±0.09)	0.2232 (±0.04)	0.5 (±0)	0.3080 (±0.04)
Decision Tree	0.6352 (±0.07)	0.6332 (±0.08)	0.6251 (±0.07)	0.6223 (±0.07)
Random Forest	0.6431 (±0.09)	0.6439 (±0.09)	0.6243 (±0.09)	0.6154 (±0.10)
AdaBoost	0.6635 (±0.05)	0.6588 (±0.04)	0.6567 (±0.04)	0.6559 (±0.04)
Naïve Bayes	**0.6753 (±0.05)**	**0.6731 (±0.06)**	**0.6732 (±0.06)**	**0.6706 (±0.06)**
QDA	0.5451 (±0.05)	0.5582 (±0.07)	0.5540 (±0.06)	0.5364 (±0.05)
Neural Network	0.6071 (±0.06)	0.6351 (±0.11)	0.5942 (±0.10)	0.5529 (±0.16)

Table 1. Comparison of ML Performance

Finally, in Figure 2 we show the receiver operating characteristic (ROC) curve, which is a graphical method to show the ability of a binary classifier system to predict the match outcomes. These show that again Naive Bayes is the best preforming method.

[8] Hyperparameters: KNN - 3 neighbors, Decision Tree - max depth = 5, Random Forest - max_depth = 5 and 10 trees, Neural Network structure is a multi-layer perceptron with ‘relu’ activation fucntion and 100 layers. These have been optimised using a GridSeach method (part of the sklearn package).

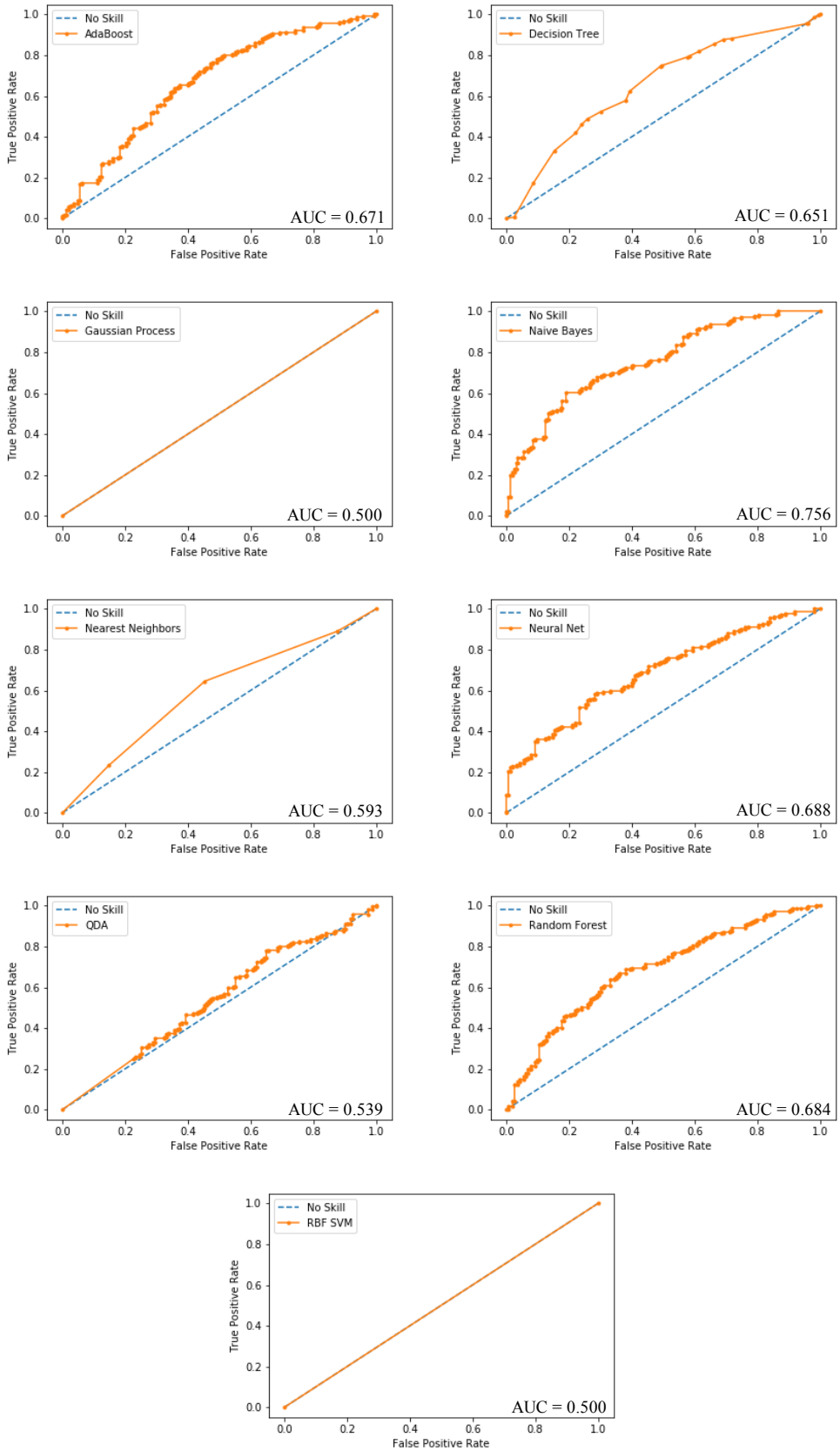


Figure 2. Machine Learning Methods ROC-Curves

Experiment 2: Year by Year

In this test, we take the top 3 performing methods that we found in the previous experiment. We show how Naive Bayes, AdaBoost and Random Forest performs across the 2015-2019 seasons and how the accuracies change year on year. The results from these are shown in Figure 3.

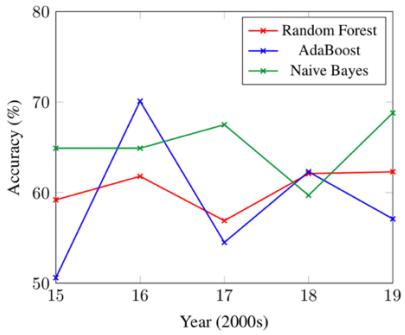


Figure 3. Year by Year Comparison

This shows that the Naive Bayes method continues to be the best performing algorithm in 60% of the years we tested, however the AdaBoost is the best performing in 2016 and joint best performing in 2018 with Random Forest. This suggests that an ensemble learning (Zhang & Ma, 2012) approach using all 3 of these methods could be a good idea to improve the prediction consistency across the seasons.

Experiment 3: Leading Work and Bookmaker Comparison

In this experiment, we compare the results from the top machine learning algorithms to the other leading academic work and the accuracy of the bookmakers. We look at the results from (Landers & Duperrouzel, 2018), who use decision trees on their feature sets and to work by (Boulier & Stekler, 2003) who use power scores in their models. We then also collect bookmaker's data over the past 5 seasons from *oddschecker.com* taking the bookmakers favourite as their prediction.[9] The results from this are shown in Figure 4.

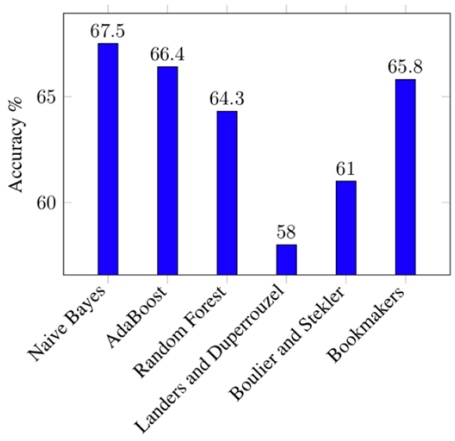


Figure 4. Model Match Outcome Prediction Accuracy Comparison.

[9] It is worth noting that bookmakers odds are set using a mixture of statistical models and market demand.

We see that the Naive Bayes outperforms Landers & Duperrouzel (2018) by 9.5%, Boulier & Stekler (2003) by 6.5% and the bookmakers by 1.7%. However, as we saw in the last experiment, the Naive Bayes would not be consistently above the bookmaker's accuracy so may not turn a profit against them. However, with some work and the use of ensemble learning we may be able to make this more consistent. We also may find that by using the pre-match bookmakers' odds as features in our models may help us to improve our models further and town profits in the betting markets.

Discussion

Our experiments find that the best performing algorithm is the Naive Bayes approach, this is able to achieve an accuracy of 67.53% and an f1-score of 0.6706. We believe this performance is due to its strong performance when features are independent, and also when dependencies of features from each other are similar between features. The next best performing approach we found was AdaBoost which performs highly with and accuracy of 66.35% and f1-score of 0.6559. This is due to the ability to adjust for non-linear relationships between features and outcomes. Finally, the third best performing approach was Random Forest at 64.31% accuracy and 0.6154 f1-score. This is due to that Random Forest is usually robust to outliers and can handle them automatically. We showed in Experiment 3 that these methods can outperform other published work in this area, we see our Naive Bayes model outperforms the Landers & Duperrouzel (2018) by 9.5% who use a Boosted Decision Tree method to make their predictions. They see a similar result to that in our decision tree test with a different feature set and we would expect a Naïve Bayes approach with their features to improve the accuracy seen in their paper. We also see that we can improve on the model that uses "power scores" in Boulier & Stekler (2003) by 6.5%. Boulier & Stekler achive a high-level accuracy using probit regressions based on power scores published in *The New York Times*[10] back to 1994. It would be interesting to combine the power scores from this model with the results in this paper to further boost the accuracy as well as using a larger training set of games from outside of the 1280 games across the 5 seaosns that we tested.

The worst performing method was Gaussian Process with an accuracy of 44.64% and f1-score of 0.3080. For the match outcome prediction this is very poor and would show that the model is no better than randomly selecting a winner. This performance is likely due to the model overfitting to the training data. Therefore, when new unseen data is given to the model it was unable to provide a good prediction. We also saw the SVM with an RBF kernel perform poorly, with an accuracy of 55.37% and f1-score of 0.3556. This again is likely due to overfitting as similarly to the Gaussian Process approach the SVM saw 100% accuracy for the training set.

In Experient 2 we explored how the top 3 approaches perform on a season by season basis. We see that even though Naive Bayes was the best approach across the 5 seasons that we tested, on a season by season basis it is top in 60% of years. We find that the accuracy of Naïve Bayes does drop off and is out performed by AdaBoost for 2 seasons (as well as Random Forest which was the joint best performing method in 2018). This may be due to the Naïve Bayes drawing samples from the population that are not fully representative. This coud be imporoved using a larger dataset (more games from across more seaosns). We may also find that AdaBoost results vary from season due to noise in the feature set or overfitting. Due to these variation it would be interesting to see if an ensemble learning approach with these three methods (as well as others) with some weighting on our confidence of the method prediction,

[10] https://www.nytimes.com/section/sports/football.

would provide more accurate and more consistent results across the 5 seasons. By doing this we could then evaluate how the methods compete against bookmakers and if they would turn a profit.

The features used to test our models (outlined in Section 4) were selected as they allow us to evaluate the team's offensive and defensive abilities for rushing and passing both in the short and long term. They are also easily obtainable from a good open data source. However, we may find that there are some features that can help us add to and improve what we have presented in this paper.[11] For example, prior results in games between the opponents and more player specific data in each team. We also discussed in Experiment 3 how the bookmakers are able to be consistent in predicting outcomes accurately. Therefore, by using their pre-match odds as features in our models may help improve the accuracy.

Further Work

It would be of interest to compare and contrast which features are best to use for match outcome prediction. This could help to reduce noise in the models if there are features that are not needed. This could be done by using feature selection techniques such as Pearson's Correlation or ANOVA (Dash & Liu, 1997). We would also want to test the pre-match odds as features to see how this affects the model accuracy.

We also would aim to provide a comprehensive test of ensemble learning methods and the different approaches that we could use on this. This paper would provide good comparison to any ensemble methods and it would be interesting to see if they can help improve accuracy and consistency. Part of this would involve learning the weighting we assign to each model that we use. We would also like explore further work on points spread regression and a comparison of results to the bookmaker's odds for the spread in games.

Finally, one element that is overlooked in match outcome predictions across all sports is the human element. There are many human factors that can change what is likely to have an impact on a match outcome. These include but are not limited to: team morale, new signings, new coaches, form, weather and other external factors. If these could be incorporated into a model by looking at match reports or human opinion through the use of natural language processing (NLP) then this again may help improve model accuracy.

Conclusion

In this paper we have implemented, tested and discussed a comparison of nine machine learning algorithms for predicting the match outcome of games in the NFL over a 5-season period. We find that the Naive Bayes method is the best preforming over the tested period, showing an accuracy of 67.53%. We also see that Random Forest and AdaBoost methods also perform very well with accuracies of 64.31% and 66.35% respectively. This would be able to better the leading published work and over some of the seasons we tested compete with the bookmaker's average. When testing these on a season by season basis though we find that the best performing switches between Naive Bayes and AdaBoost, which leads us to conclude that an ensemble learning approach with the best performing algorithms would being more consistency over a longer period of time.

[11] Although we still have shown a good comparison of ML methods with a comprehensive feature set, we believe even with new features the best performing methods would still stay the same.

Acknowledgements

We would like to thank the reviewers for their comments. This research is supported by the AXA Research Fund and the EPSRC NPIF doctoral training grant number EP/S515590/1.

References

Beal, R., Norman, T., & Ramchurn, S. (2019). Artificial intelligence for team sports: a survey. *The Knowledge Engineering Review, 34*.

Boulier, B., & Stekler, H. (2003). Predicting the outcomes of National Football League games. *International Journal of Forecasting*, 257-270.

Clarke, S., & Norman, J. (1995). Home ground advantage of individual clubs in English soccer. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 509-521.

Constantinou, A., Fenton, N., & Neil, M. (2012). pi-football: A Bayesian network model for forecasting Association Football match outcomes. *Knowledge-Based Systems*, 322-339.

Crowder, M., Dixon, M., Ledford, A., & Robinson, M. (2002). Dynamic modelling and prediction of English Football League matches for betting. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 157-168.

Dixon, M., & Coles, S. (1997). Modelling association football scores and inefficiencies in the football betting market. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 265-280.

Dixon, M., & Robinson, M. (1998). A birth process model for association football matches. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 523-538.

Glickman, M., & Stern, H. (1998). A state-space model for National Football League scores. *Journal of the American Statistical Association*, 25-35.

Han, S., Qubo, C., & Meng, H. (2012). Parameter selection in SVM with RBF kernel function. *World Automation Congress 2012*, 1-4.

Harville, D. (1977). The use of linear-model methodology to rate high school or college football teams. *Journal of the American Statistical Association, 72*, 278-289.

Harville, D. (1980). Predictions for National Football League games via linear-model methodology. *Journal of the American Statistical Association*, 516-524.

Joseph, A., Fenton, N., & Neil, M. (2006). Predicting football results using Bayesian nets and other machine learning techniques. *Knowledge-Based Systems*, 544-553.

Landers, J., & Duperrouzel, B. (2018). Machine learning approaches to competing in fantasy leagues for the NFL. *IEEE Transactions on Games*, 159-172.

Maher, M. (1982). Modelling association football scores. *Statistica Neerlandica*, 109-118.

McCabe, A., & Trevathan, J. (2008). Artificial intelligence in sports prediction. *Fifth International Conference on Information Technology: New Generations* (S. 1194-1197). IEEE.

Sankaranarayanan, V. V., Sattar, J., & Lakshmanan, L. (2014). Auto-play: A data mining approach to ODI cricket simulation and prediction. *Proceedings of the 2014 SIAM International Conference on Data Mining*, (S. 1064-1072).

Suykens, J., & Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural processing letters*, 293-300.

Yang, T., & Swartz, T. (2004). A two-stage Bayesian model for predicting winners in major league baseball. *Journal of Data Science*, 61-73.