Math 400 Research Project: Predicting Football Games

Noah Giles

December 12, 2019

Abstract

The goal of this paper is to use different model to come up with a ensemble model that can accurate depict who was going to win a football game. Overall the Random Forest Classifier had the best prediction of who would win a football game. The data set used is a combination of an elo ranking system kept by fivethrityeight sports and a Casinos odds book. Inside of these data sets 5 variables were used to help find a way to predict the game outcome. There are many factors in a football game that one can not account for thus the accuracy will not be able to pass beyond a certain threshold consistently.

1 Introduction

Every Year millions of people watch American football. Football is one of the most watched sports in America. Since becoming a major sport gambling on football games has become very popular. Many casinos use different factors to determine the odds for a football game. In this research project the points line and the over under line will both be used to help predict who will win a football game. The other varible that will be used is the Elo rating system. This system is based of the Elo chess rating system and is a fairly simplistic rating system.

2 Offense

The Goal of the offense is to advance the ball at least 10 yards in 4 attempts. The goal of the offense is to advance the ball into the end zone. The offense has 4 opportunities per 10 yards to advance the ball. The offense has a pivotal role in winning a football game. Since the offense is the main reason why a team score having a bad offense will undoubted

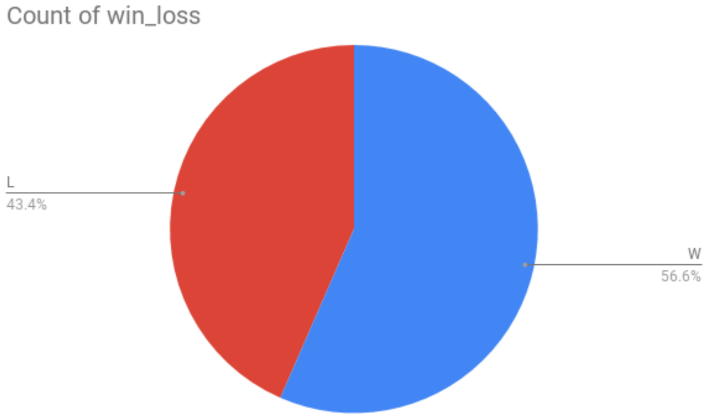


Figure 1: Total wins and loss for home teams

influence the success of your team. The most important player on offense is the Quarterback. The quarterback has the ball in his hands all the time. This aspect makes him the most important player on the field.The mentality that offensive players approach the game with is a more patient strategy. Every play it takes almost all 11 people on the field to create a positive play. Rarely is there instances when one person makes a mistake and it results in a positive play.

3 Defense

The Goal of the defense is to prevent the offense from scoring points. The defense can either do these by taking the ball away from the offense or stopping the offense from advancing the ball at least 10 yards in 4 tries. On defense the only thing that matters is how many points you can keep the opposing offense from scoring. Defense has a very different mentality than offense. Most defensive mentality consist of the "bend do not break" mentality. This mentality shapes there strategy because all it takes is one play or player to make a positive play and keep the opposing team from scoring. The margin for error on defense between individual players is drastically increased because since it takes an offense to be practically perfect to achieve a positive play; conversely on defense one player can make a huge impact on the game. This is why defenses strive to create better match-ups for their players.

4 Variable: Elo Difference

This variable tracks the difference in rankings. Each time is assigned an Elo rating with the home team having an added home field advantage. The elo rating system works by creating an even playing field for all teams. When one team beats another there elo rating will increase. Some of the factors that go into this rating are how much a team favored wins by. If the team that is heavily favored wins a close game their elo score has a chance to decrease because that team should have won by a bigger margin.

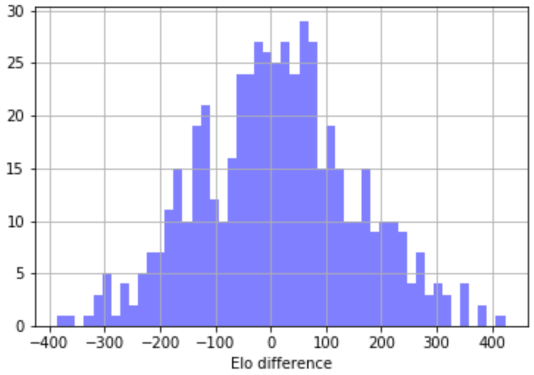


Figure 2: Frequency distribution with Elo Difference (negative elo means home team was not favored to win the game)

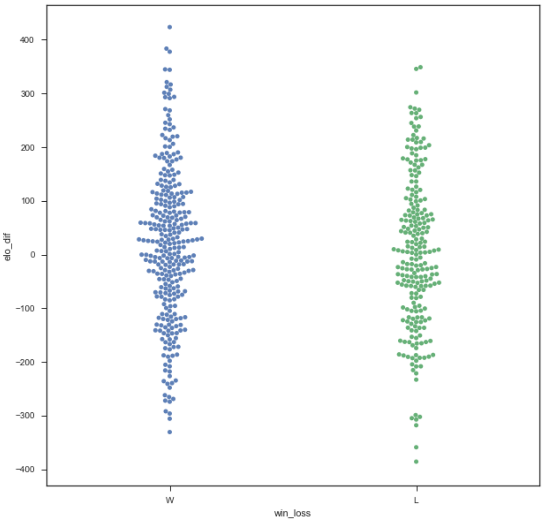


Figure 3: This graph shows the complex problem using one variable because each dot represents a game and which team was favored. If the dots were on the negative y axis the away team was favored. If the dots are on the positive y axis the home team was favored. The challenge is focusing on the dots above 0 that are below 100 because these are viewed as the closer games and can go either way.

5 Variable: Points Line

The Points line is the outcome of Vegas casino models. This point line favors one team and suggest how many points each team should win by. This data can be very useful because there is a lot that goes into the point line. For instance how many points an offense scored in the game before and how many points the defense also gave up in the game before. These statistics play a role in how well the team is playing at a certain time. Teams tend to have streaks when they are playing better that the Elo rating might miss because they originally were ranked so low.

6 Variable: Over Under

The over under variable tracks how many points Vegas thinks will be scored in the game. This variable combined with the point line can be very insightful. If the point line is high and the over under is also high this means the team is heavily favored to win by a lot.

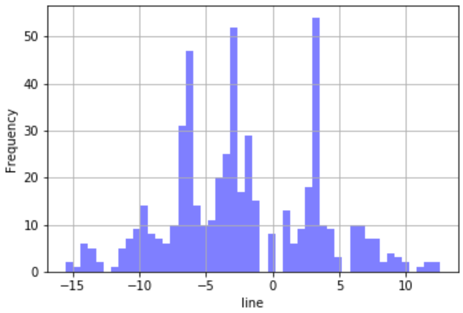


Figure 4: Frequency distribution with Point line (negative line means home team was favored to win the game

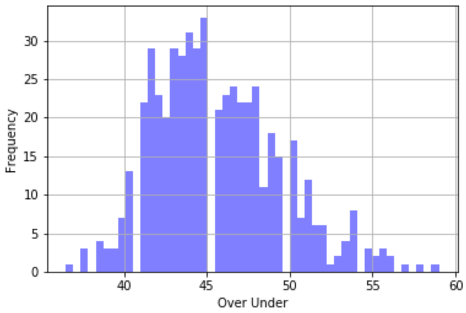


Figure 5: Frequency distribution with Point line (negative line means home team was favored to win the game

7 Variable: Elo Probability

The Elo Probability is how confident the home team is to winning that specific game. This estimate does not however include a home field advantage quantifier. Thus this is purely based off of the elo score before the home advantage multiplier was applied.

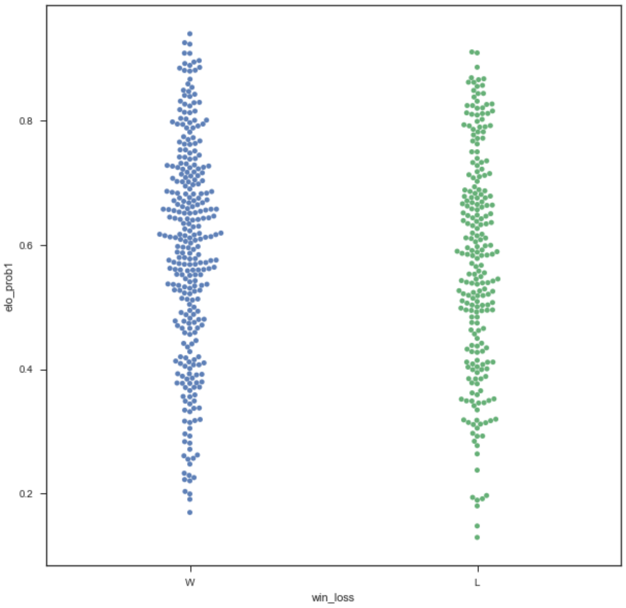


Figure 6: This graph shows the complex problem this variable alone. Each dot represents a game and which team was favored. If the dots were below .5 the home team was not favored.

8 Variable: Neutral site Game

This Variable keeps track of the location of the game of the game was in a neutral site location this means that both teams would have to travel. A neutral site game is in important factor because home advantage is a huge factor in who will win games. Just on average the home team wins more than away team just based on Figure 1. Teams that often play at neutral site games are often in the super bowl and or more recently in London. Since there is a limited sample size in the data for neutral site games having this element in there when there is a neutral site game aids the model in predicting the outcome.

9 Model: Naive Bayes Classifier

The Naive Bayes Classifier Model is a probabilistic model that uses Bayes theorem. This model looks at each dependant variable individually separate from the rest. It then determines the probability given this data point to determine where the outcome lands. For example it looks at the elo difference and determines the probability that having a higher or lower contributes to a win. This model is very fast and accurate and mild efficient in predicting the outcome.

	Win	Loss
Win	15	49
Loss	10	60

This table shows the confusion matrix. Where the win number in each column matches up is where the model predicted it to be a win and it actual was. The bottom left corner is where it predicted to be a win but it was a loss. This is a false positive. The top right is were it predicted a loss but it was actually a win this is a false negative. The bottom left is where the model predicted a loss and it was a loss. This model however was not a good estimate for this test data. It was 55 percent accurate. This however is misleading because using cross validation the average accuracy score was 64 percent. This model will on average predict the outcome of a football game 64 percent of the time. Cross Validation uses multiple random sets of data from the data set and test it against the model to determine how accurate it is. In this cross validation test I used 10 instances of training and testing the data and this leads to becoming close to the true mean of the model accuracy.

10 Logistic Regression

Logistic Regression is one of the most basic forms of regression this simple comes of with coefficients for each variable and weighs them. In this model a specific score should classify the data points as a win or a loss.

	Win	Loss
Win	24	40
Loss	17	53

The model performed poor on the training data scoring a 57 percent but the cross validation saved this model because it received a 64.7 percent cross validation score. This model is a helpful predictor on who will win a football game.

11 Random Forest Classifier

A random forest classifier is a set of classifier trees. Inside of each tree exist many nodes and branches. Each node contains an if statement deciding which way it should be classified.

Once the data point reaches the bottom of a node, it is thus classified as a win or a loss. This model can be very useful in classifying data that is multi-faceted. This aspect has cause it to be a very good predictor for football game outcomes.

	Win	Loss
Win	28	36
Loss	17	53

The model performed poor on the training data scoring a 60 percent but the cross validation saved this model because it received a 66.7 percent cross validation score. This model had the highest cross validation score. Therefore on average it is the best predictor of who will win a game.

12 K-Neighbors Classifier

This model looks at the training data and groups data points together and looks to see whats around it. This method is very simplistic and overall I found that it has a lot of trouble figure out which of the middle data points it should be classified in. The parameters are looking for the 14 nearest neighbors to determine where it should be classified at.

	Win	Loss
Win	34	30
Loss	19	51

The confusion matrix showed that the model was 63 percent accurate. This however is poor considering it has a very low cross validation score of 60 percent. The only upside of this model is that it does not have as many over predicting losses when it should be a win.

13 Neural Network Classifier

This classifier works by having layers of nodes that look at different combinations of data inputs to perform. The neural network in this model has 5 hidden layers. Each layer is supposed to be for each specific variable in the model. The total amount of nodes in the model is 117. The activation function used in the model was relu. Relu stands for rectified linear unit. This activation function for any negative bias values is 0.

	Win	Loss
Win	44	20
Loss	19	51

Based on the confusion matrix the model performed with 70 percent accuracy. This however was also misleading because the cross validation score was 56 percent. This model however not the best overall. It has insightful information in the harder to predict data points. The middle ground in any model is the hardest to predict. Understanding that this model is not over fitting extremely to the win side or loss side is an important factor. The other model usually favor one side or another but this model does not over categorize one side of data which is why, however does not overall have a high accuracy score, it can provide insightful information.

14 Voting Classifier

The Voting classifier model a way to combine all the models input and create a ensemble model to predict output. The model in this classifier are Logistic Regression, Random Forest, Naive Bayes, Nueral Network and K-nearest neighbors. This model combines all the prediction of each on and submits a final prediction based on how each model classified it. This is the democracy model because each one has a vote but some votes can also be weighed more than others. In this model the neural network is weighed more than the others to have the final say. The neural network, however does not have have the best cross validation score, helps predict the middle values better than the other model. Thus when there is a tie the model will however vote with the other models. This aspect gives the Neural Network the final say but the other models can overrule it if there is a tie. This aspect helps the model predict the outcome.

	Win	Loss
Win	45	19
Loss	21	49

The reason the neural network was chosen to be the ultimate weight was because it minimizes the error in the middle in each model there was an over arching them that it would over predict a loss. The optimal model would be to have a model that predicts win and loss equally thus because if one model classified it as a win but its accuracy when predicting a win is at 50 percent there is no reason to use that model. Human intuition can do a better job. The voting classifier however performed very well. The confusion matrix shows it has a 70 percent accuracy on the training data. It also has a 63 percent cross validation score which is in the average percentage of the other models. This however is a better model because of the fact that it does not over predict to one side

15 Conclusion

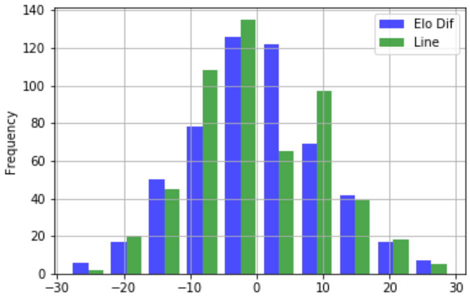
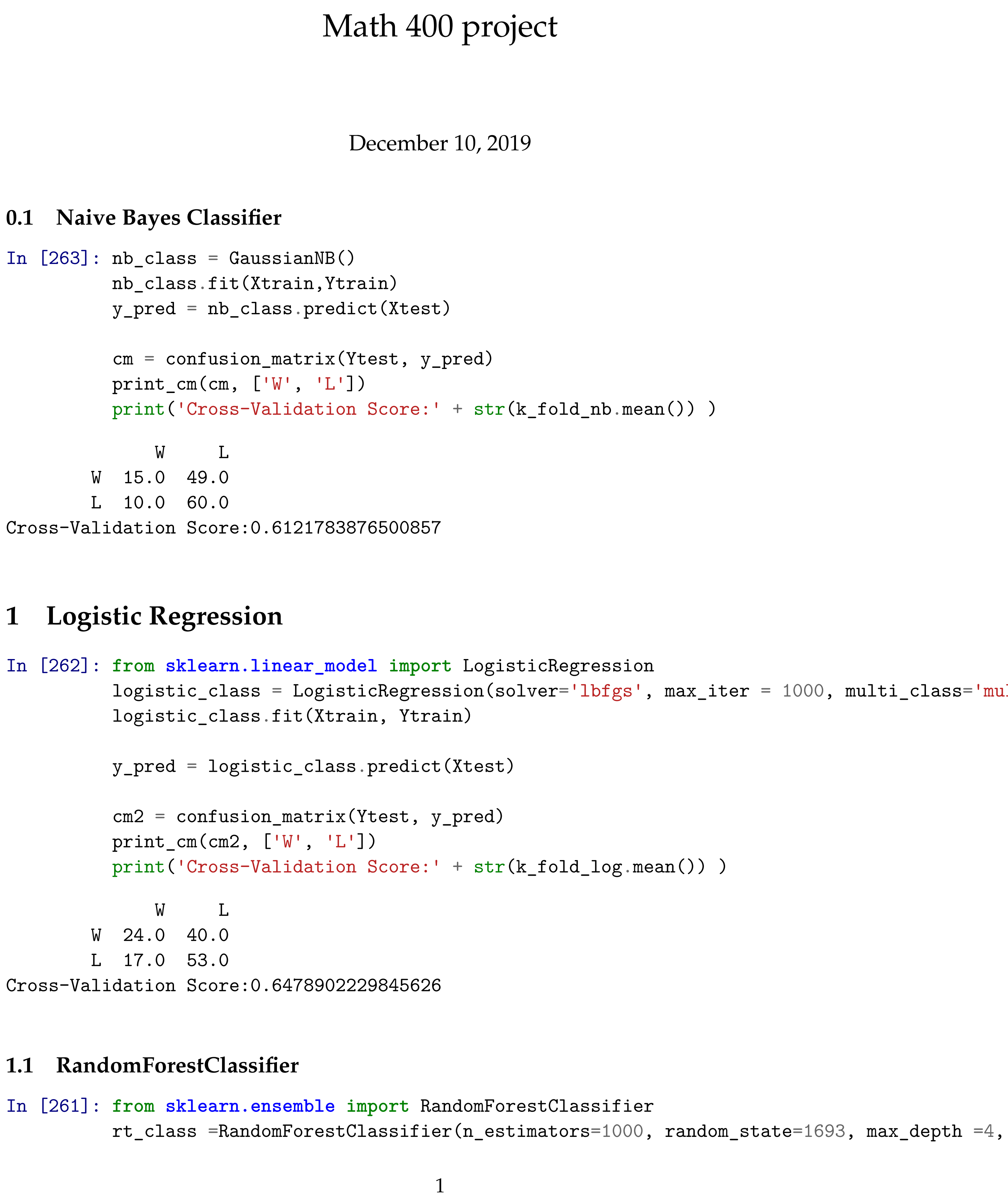


Figure 7: Two Frequency of Point line and Elo Difference

Overall the model the perform the best was the Random Forest Classifier. In comparison with other football prediction models the Fivethrityeights elo model alone is 51 percent accurate. This model contains some of the same variables that my model does but does not have a strong elo difference home quantifier. The flaws in this model is that it doesn't account for the weather, draft picks from the next year, trades, and resting players. These are all factors that were missed in the Elo model but covered in mine. The Vegas data I decided to use was very useful. In Vegas point line the include factors such as who is going to play and weather. These aspects make my model a more complete model than most other prediction models. Espn's Football power index was shown to perform at 63 percent on this same data set. This however can also be misleading because as stated before the cross validation score would be a better way of testing these statement. For instance there is no real way to tell if the Espn model was better than my best model because there is no data over time that their model was tested against and averaged. My model is still inherently better than the professional gamblers who study football. I believe that there is an a thresh hold that a model can not surpass on accuracy. This is because there are so many factors that go into a football game. Football is considered to be the most lucky sport because there isn't a big sample size of games to weed through to figure out who is the best. The NFL season last 17 weeks with each team playing 16 games. The sample size to determine which team is better than the other is drastically challenging. This is why a model that predicts in the 60-70 percent range should be considered a good one.

Code Using Python



Math 400 project

December 10, 2019

0.1 Naive Bayes Classifier

```
In [263]: nb_class = GaussianNB()
          nb_class.fit(Xtrain,Ytrain)
          y_pred = nb_class.predict(Xtest)

          cm = confusion_matrix(Ytest, y_pred)
          print_cm(cm, ['W', 'L'])
          print('Cross-Validation Score:' + str(k_fold_nb.mean()) )
```

```
            W     L
        W  15.0  49.0
        L  10.0  60.0
Cross-Validation Score:0.6121783876500857
```

1 Logistic Regression

```
In [262]: from sklearn.linear_model import LogisticRegression
          logistic_class = LogisticRegression(solver='lbfgs', max_iter = 1000, multi_class='mu
          logistic_class.fit(Xtrain, Ytrain)

          y_pred = logistic_class.predict(Xtest)

          cm2 = confusion_matrix(Ytest, y_pred)
          print_cm(cm2, ['W', 'L'])
          print('Cross-Validation Score:' + str(k_fold_log.mean()) )
```

```
            W     L
        W  24.0  40.0
        L  17.0  53.0
Cross-Validation Score:0.6478902229845626
```

1.1 RandomForestClassifier

```
In [261]: from sklearn.ensemble import RandomForestClassifier
          rt_class =RandomForestClassifier(n_estimators=1000, random_state=1693, max_depth =4,
          rt_class.fit(Xtrain, Ytrain)
          rt_pred = rt_class.predict(Xtest)
          rt_conflat = confusion_matrix(Ytest, rt_pred)
          print_cm(rt_conflat, ['W', 'L'])
          print('Cross-Validation Score:' + str(k_fold_rt.mean()) )
```

```
            W     L
        W  28.0  36.0
        L  17.0  53.0
Cross-Validation Score:0.6536192109777016
```

2 Voting Classifier

```
In [260]: from mlxtend.classifier import EnsembleVoteClassifier
          from sklearn.ensemble import RandomForestClassifier, VotingClassifier
          clf1 = LogisticRegression(solver='saga', multi_class='multinomial',random_state=1693
          clf2 = RandomForestClassifier(n_estimators=1000, random_state=1693, max_depth =4, mi
          clf3 = GaussianNB()
          clf4 = MLPClassifier(hidden_layer_sizes = (50,1,30,7,30), activation='relu', solver=
          clf5 = KNeighborsClassifier(n_neighbors=14)


          eclf1 = VotingClassifier(estimators=[('lr', clf1), ('rf', clf2), ('gnb', clf3),('mlp
          model = eclf1.fit(Xtrain, Ytrain)
          yypred= model.predict(Xtest)
          vt_conflat = confusion_matrix(Ytest, yypred)
          print_cm(vt_conflat, ['W', 'L'])

          print('Cross-Validation Score:' + str(k_fold.mean()) )
```

```
            W     L
        W  45.0  19.0
        L  21.0  49.0
Cross-Validation Score:0.6142024013722127
```

2.1 K-Neighbors Classifier

```
In [259]: Knn = KNeighborsClassifier(n_neighbors=14)
          Knn.fit(Xtrain,Ytrain)
          y_pred3 = Knn.predict(Xtest)
          cm_Knn = confusion_matrix(Ytest,y_pred3)
          print_cm(cm_Knn, ['W', 'L'])
          print('Cross-Validation Score:' + str(k_fold_Knn.mean()) )
```

```
        W     L
W    34.0  30.0
L    19.0  51.0
Cross-Validation Score:0.6030874785591767
```

2.2 Neural Network Classifier

```
In [258]: from sklearn.neural_network import MLPClassifier
          mlp = MLPClassifier(hidden_layer_sizes = (50,1,30,7,30), activation='relu', solver='
          mlp.fit(Xtrain,Ytrain)
          y_pred2 = mlp.predict(Xtest)
          cm_nn = confusion_matrix(Ytest,y_pred2)
          print_cm(cm_nn, ['W', 'L'])
          print('Cross-Validation Score:' + str(k_fold_mlp.mean()) )
```

```
        W     L
W    44.0  20.0
L    19.0  51.0
Cross-Validation Score:0.5655574614065181
```

2.3 Cross Validation Scores

```
In [ ]: from sklearn.model_selection import cross_val_score

        k_fold_rt = cross_val_score(estimator = rt_class, X = X, y= Y, cv=10,scoring='accuracy
        k_fold_mlp = cross_val_score(estimator = mlp, X = X, y= Y, cv=10,scoring='accuracy')
        k_fold_Knn = cross_val_score(estimator = Knn, X = X, y= Y, cv=10,scoring='accuracy')
        k_fold = cross_val_score(estimator = model, X = X, y= Y, cv=10,scoring='accuracy')
        k_fold_log = cross_val_score(estimator = logistic_class, X = X, y= Y, cv=10,scoring='a

        k_fold_nb = cross_val_score(estimator = nb_class, X = X, y= Y, cv=10,scoring='accuracy
```

16 Bibliography

Dare and Holland, 2004 W.H. Dare, A.S. Holland Efficiency in the NFL betting market: modifying and consolidating research methods Applied Economics, 36 (2004), pp. 9-15 Boulier and Stekler, 2003 B.L.

Boulier, H.O. Stekler Predicting the outcomes of National Football League games International Journal of Forecasting, 19 (2003), pp. 257-270

Grant and Johnstone, 2010 A. Grant, D. Johnstone Finding profitable forecast combinations using probability scoring rules International Journal of Forecasting, 26 (2010), pp. 498-510

Harville, 1980 D. Harville Predictions for National Football League games via linear-model methodology Journal of the American Statistical Association, 75 (371) (1980), pp. 516-524

oice, "How To Play Our NFL Predictions Game". Fivethrityeight. 2018 https://fivethirtyeight.com/features/how-to-play-our-nfl-predictions-game/

ESPN (2016). A guide to nfl fpi. http://www.espn.com/blog/statsinfo/post/$_{/}id/123048/a-guide-to-nfl-fpi.Accessed$: 2016 − 10 − 30.

$AgrestiA$(2007)$Anintroductiontocategoricaldataanalysis.Wileyseriesinprobabilityandstatistics.Hoboken(N.J.)$ $Wiley - Interscience.https$: $//doi.org/$10.1002/047011475