For my final project I chose to analyze betting data for the NFL. I was interested in this topic as I am interested in a career in sports analytics and I also enjoy sports gambling. More specifically I find the statistics in the gambling industry very interesting and I thought I could potentially use the statistical models we learned about in class in order to be able to predict which bet to place on NFL games to make money.

There are two main types of bets for NFL games. The first of these are bets on the spread. Each NFL game has a 'favorite' and an 'underdog'. The spread is determined by how much the sportsbook think the favorite team is better than the underdog team and by how much they will win the game by. Some examples of the spread line for the favorite would be -3, -3.5, -7, -7.5, etc. If the line was set to -3.5 and you bet on the favorite you would then need the favorite to win the game by 4 or more points in order to win your bet, this is called covering the spread. If the favorite lost by 3 or less points you would subsequently lose your bet, this is called not covering the spread. On the reverse some examples of the spread line for the underdog would be +3, +3.5, +7, +7.5, etc. If the line was then set at +3.5 and you bet on the underdog if the underdog lost the game by 3 or less points or if they won the game you would win your bet. But if the underdog lost the game by 4 or more points you would then lose your bet. The bet can also tie if, for example, the line was set at -7 and the favorite won by exactly 7 points or if the line was set at +7 and the underdog lost by exactly 7 points, this is called a push. The other type of bet is a bet on the over/under line. Every game has an over/under line set based on the two teams that are playing. If the game is between two more offensive teams then the line is set higher and if its two more defensive teams the line is usually set lower. Some examples of over/under lines would be 24, 30.5, 35, 42.5. If the line was set at 30.5 and you bet on the over, both teams would need to combine score 31 or more points for you to win your bet. So, if the home team scored 20 and the away team scored 15 that would be 35 points and you would win your bet. On the other hand, if you had bet the under you would have lost your bet. But if the home team had scored 14 and the away team had scored 13 and you bet the under you would have won your bet as the teams had only scored 27 combined points which is under the 30.5 point line.

And if you had bet the over you would have lost your bet. Just like the spread you can also push on this bet if say for example the line was set at 42 and the home team scored 20 and the away team scored 22.

The datasets I found for the NFL games and betting data was found on Kaggle. The data set consisted of every NFL game from 1966 to 2021. However, I shrunk this down to every games from 1979-2018 because those are the years that had all the betting data. This got rid of any missing values. It also consisted of a main dataset which had 17 variables to start. Some of the most important variables were the home team, their score, the away team, their score, the team which was favorited in the spread, the spread line, the over/under line, the weather humidity, temperature and the weather conditions. I also combined this dataset with two other datasets that gave us the team's divisions and also the stadium data to tell us where the games were played. This data allowed me to create the variable divisional game which gave a 1 if it was a divisional game or 0 if it was not. I also created the spread result variable and over/under result variable to allow me to know the outcome of the bets for the game through using the scores of the home and away teams.

I started with some basic analysis into which team won the most bets if you bet on them.

team_favorite_id <chr>	count <int>
PIT	217
SF	216
NE	215
DAL	198
GB	197
DEN	193
TB	103
ARI	99
CLE	96
CAR	91
JAX	75
HOU	55

Above is the amount of times if you were to bet on the team as a favorite that you then won your bet. The first 6 teams are the teams with the most wins while the bottom six teams are the teams with the least number of wins.

team_underdog_id <chr>	count <int>
DET	205
TB	204
ARI	201
CIN	185
CLE	185
NYJ	184

SF	120
DEN	119
CAR	118
JAX	111
BAL	90
HOU	76

Above is the amount of times if you were to bet on the team as an underdog that you then won your bet. The first 6 teams are the teams with the most wins while the bottom six teams are the teams with the least number of wins.

team_id <chr>	count <int>
GB	349
DEN	338
NE	335
TEN	335
NO	330
MIN	328

TB	284
CLE	275
JAX	198
CAR	189
BAL	176
HOU	136

Above is the amount of times if you were to bet on the over and that team was in the game that the over hit. The first 6 teams are the teams with the most wins while the bottom six teams are the teams with the least number of wins.

team_id <chr>	count <int>
TB	349
MIA	346
KC	345
NYG	343
PIT	340
PHI	339

ARI	304
TEN	304
CAR	205
BAL	201
JAX	196
HOU	139

Above is the amount of times if you were to bet on the under and that team was in the game that the under hit. The first 6 teams are the teams with the most wins while the bottom six teams are the teams with the least number of wins. As you can notice the teams in the bottom six had four very consistent teams there.

These teams were the Carolina Panthers, Jacksonville Jaguars, Baltimore Ravens and the Houston Texans. I looked into this and found that these teams were all added to the NFL later than 1979. The Panthers were added in 1995, the Jaguars in 1995, the Ravens in 1996 and the Texans in 2002. This meant that these teams just had fewer games then the other teams. One particular bit of information that came from this was that the Ravens actually were not present in the bottom 6 of the teams that if you bet on them and they were favorited that you won your bet. This means that they actually had more wins as the favorites in less games than other teams that were in the dataset for the whole time like the Tampa bay Buccaneers. Because of this I decided to look into the percentage of games that each team covered when they were favorites.

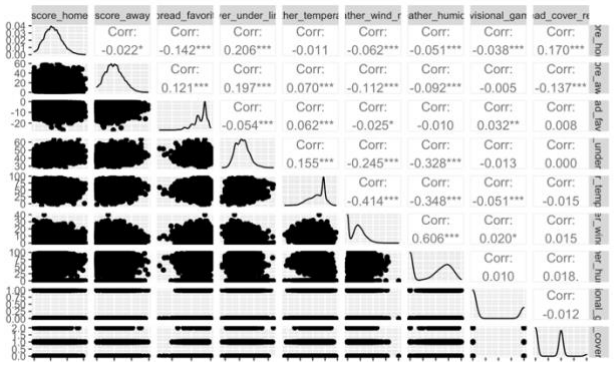
team_id <chr>	count <int>	spread_count <int>	cover_percentage <dbl>
GB	252	197	0.7817460
IND	226	143	0.6327434
BUF	242	150	0.6198347
DEN	319	193	0.6050157
ATL	231	133	0.5757576
NE	381	215	0.5643045
CAR	237	91	0.3839662
NYJ	307	117	0.3811075
DET	276	104	0.3768116
JAX	234	75	0.3205128
CLE	303	96	0.3168317
HOU	200	55	0.2750000

Above is the top 6 teams and bottom 6 teams in terms of cover percentage. As you can see the teams did change a bit from just the sheer number of times that they covered. For example, the Pittsburgh Steelers which was the team with the most wins from the previous analysis is not even in the top 6 for percentage of times that if they were the favorite and you bet on them that you then won your bet. And for the bottom

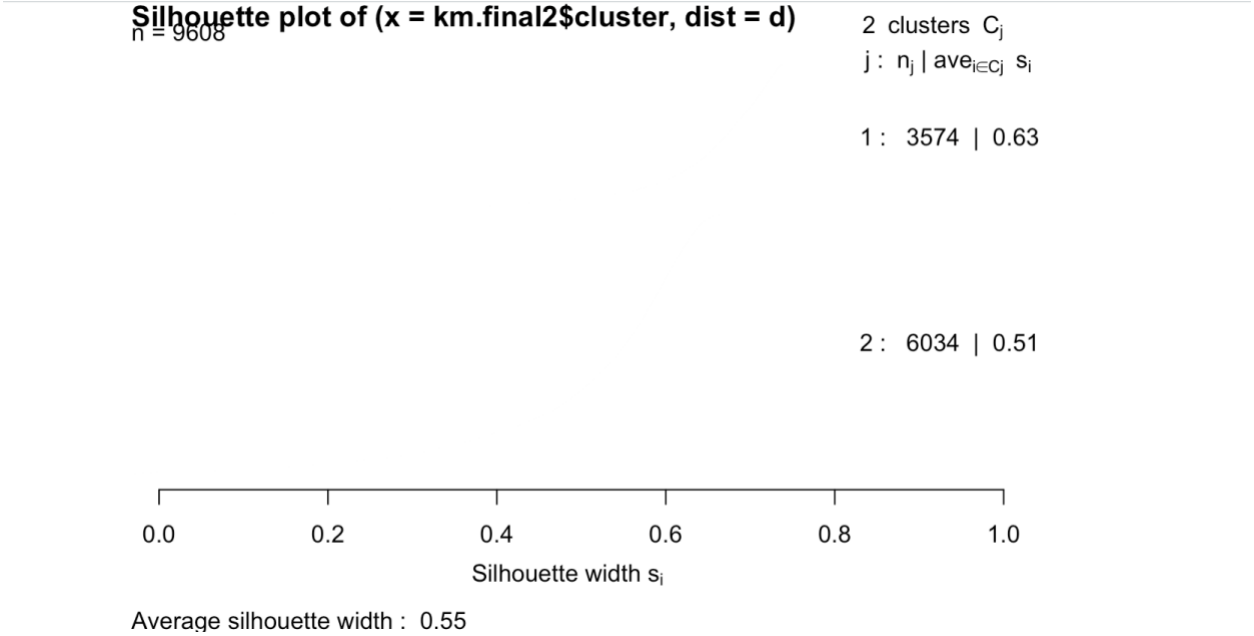
6 the Detroit lines which were not previously in the bottom 6 in terms of the number of times that they covered, were now seen as the fourth worst team in terms of the percentage of times that they covered.

After this analysis I moved onto unsupervised learning.

```
$ score_home         : int  31 7 6 10 14 17 28 34 22 23 ...
$ score_away         : int  16 9 3 0 0 24 22 40 25 17 ...
$ spread_favorite    : num  -3 -5 -3 -3 -1 -4 -7 -5 -2 -7 ...
$ over_under_line    : num  30 39 31 31.5 37 36.5 32 32 41 31.5 ...
$ weather_temperature: int  79 74 78 69 76 70 70 72 73 76 ...
$ weather_wind_mph   : int  9 15 11 6 8 10 11 0 10 9 ...
$ weather_humidity   : num  87 74 68 38 71 77 67 0 76 84 ...
$ divisional_game    : num  0 1 1 0 0 0 0 1 0 1 ...
$ spread_cover_result: num  1 0 2 1 1 0 0 0 0 0 ...
```



I first ran the ggpairs function to look at the correlation between the variables listed above. Correlation was relatively low for all variables with the highest correlation being .606 and the next highest being .414. I next ran PCA and did not find it to be a useful analysis tool, so I will not include it for this project. I believe this was the case because the correlation between variables was low and I also had two categorical variables that I included in the PCA. I did however look into the optimal number of clusters for this dataset using the default variables distances.



I analyzed silhouette plots for 2-7 clusters and as shown above 2 clusters had the highest silhouette width with a value of 0.55.

The final analysis I did was 2 supervised learning models. For my supervised models I chose my response variable to be the over/under result variable which produced an O if the game hit the over, a U if the game hit the under and a P if it was a push. Below are the explanatory variables used in order to predict the over/under result variable.

```
$ spread_favorite    : num  -3 -5 -3 -3 -1 -4 -7 -5 -2 -7 ...
$ over_under_line    : num  30 39 31 31.5 37 36.5 32 32 41 31.5 ...
$ weather_temperature: int  79 74 78 69 76 70 70 72 73 76 ...
$ weather_wind_mph   : int  9 15 11 6 8 10 11 0 10 9 ...
$ weather_humidity   : num  87 74 68 38 71 77 67 0 76 84 ...
$ weather_detail     : chr  "C" "C" "C" "C" ...
$ over_under_result  : chr  "O" "U" "U" "U" ...
$ divisional_game    : num  0 1 1 0 0 0 0 1 0 1 ...
```

My goal for these models was to find an equation that would allow me to predict the over under result before the game took place in order to place bets on the games and win money more often than I lost money. Because of this, variables such as the home and away teams scores and the spread result were taken out of the explanatory variables because they were variables you would only be able to know after the game had taken place. The first model I chose to run was linear discriminant analysis or LDA. I chose to run this model as it is a classification model that can deal with 3 classes as opposed to 2 with KNN.

```
Linear Discriminant Analysis

9608 samples
   7 predictor
   3 classes: 'O', 'P', 'U'

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 7687, 7688, 7685, 7686, 7686
Resampling results:

  Accuracy  Kappa
  0.51207   0.04153287

Call:
lda(x, grouping = y)

Prior probabilities of groups:
        O         P         U
0.48428393 0.01894255 0.49677352
```

```
Coefficients of linear discriminants:
                              LD1          LD2
spread_favorite      0.020191141 -0.030864050
over_under_line      0.111744169  0.004099433
weather_temperature  0.018219217  0.002398865
weather_wind_mph     0.194210679  0.001032284
weather_humidity     0.002309624 -0.003744072
weather_detailD      1.209140773  0.776495310
weather_detailF     -2.075165082  8.489461098
weather_detailR      1.063278448  5.945592999
weather_detailS     -1.994662302  8.407300109
divisional_game      0.933480202  0.381053869

Proportion of trace:
 LD1  LD2
0.92 0.08
```

Above are the results of the LDA model. This model had an accuracy of 51.20% which was actually fairly successful. The next model I ran was random forest because again it can handle categorical variables with 3 classes.

```
Random Forest

9608 samples
   7 predictor
   3 classes: 'O', 'P', 'U'

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 7687, 7688, 7685, 7686, 7686
Resampling results across tuning parameters:

  mtry  Accuracy   Kappa
  1     0.5074933  0.03148695
  2     0.5093680  0.03741764
  3     0.5039550  0.02605329

Accuracy was used to select the optimal model using the largest value.
The final value used for the model was mtry = 2.
rf variable importance
```

	Overall <dbl>
weather_temperature	100.0000000
over_under_line	91.7725107
weather_humidity	85.3023209
spread_favorite	80.7292935
weather_wind_mph	74.0531380
divisional_game	9.8635848
weather_detailR	2.8375427
weather_detailD	2.6402099
weather_detailF	0.4863659
weather_detailS	0.0000000

Above are the results of the random forest model. As you can see the accuracy was 50.94% which was actually slightly worse than the LDA model. Random forest analysis did however allow me to see which variables were the most important in predicting our over/under result variable. I found it interesting that weather temperature was the most important variable as I thought the over/under line may have been the most important. I also found it interesting how unimportant the divisional game variable was as divisional games are very important in the NFL and so I thought they would have a large effect on the over/under.

In conclusion my goal of using NFL game data in order to predict which bet to place on NFL games to make money was fairly successful. On average to make money you need about a 52%-win rate and my LDA model was close at 51.20%. I am sure that with some tweaking and more information I could build an even better model. If I had access to more information such as individual player and coach data I think the model could be greatly improved. It is my guess that certain quarterbacks have high rates of hitting the over or going under than others and same with coaches because there are many coaches that are better either offensively or defensively which would affect how many points their team scores and

how many points the opposition team scores. If I had more time as well in this upcoming football season I may test out my model. I could go through and use the predict function for every NFL game using a controlled amount of money for each game ($1) and see if it does end up being profitable or close to it. In the future I could also look at the spread/cover result variable and see how successful/accurate a model would be for predicting that variable and see if that would be a profitable way of betting on NFL games.