Do More Rest Days Increase the Odds of Winning in the NFL?

Nathan Warren

12/10/2019

Summary

The goal of this paper was to determine if the number of rest days was influential on whether an NFL team won or lost a game. In addition, the model is also useful for understanding which National Football League (NFL) game events are statistically significant and to what magnitude they increase or decrease the odds of winning. A logistic regression was used to understand the effects of these game events on the outcome of the game. It was found that fourth down completions, defensive touchdowns, and being the Patriots increase the odds of winning the most. Total number of fourth downs, interceptions, and fumbles increase the odds of losing the most.

Introduction

This paper studies data scraped from ESPN from the 2002 season till week 6 of the 2019 season. Regular season matches and post season games are included in the data set. This consisted of a total of 8666 viable games. There are 32 teams in the NFL which all play matches on a similar schedule. A single match occurs on any given Thursday during the season, a single match on Mondays, and 14 on Saturdays. The season lasts 17 weeks, with each team receiving a single 'bye' week, in which they do not play. Occasionally a team may play on a Monday night and have to play again on a Thursday night. The amount of rest days was separated into below 7 days, 7 days (the average amount of rest days), and above 7 days. A logistic regression was created to examine if the number of days of rest a team had between games influenced their probability of winning. The model is also useful for examining what factors increase and decrease team's odds of winning a match in the National Football League (NFL).

Data Preprocessing

The data was scraped from the ESPN website and includes all NFL game stats from 2002 to 2019 (Week 6). The data was taken from the datasets subreddit on Reddit. All regular season games and playoff games are included, while the Pro Bowl was not. In total there are 4628 games in the data set. There are 3 games that are missing from the data set due to the ESPN website being unable to load these games. These games are: DAL @ WASH 12-30-2007, CAR @ PIT 12-23-2010, and TB @ ATL 01-01-2012. These games were added manually using www.pro-football-reference.com as a resource for the statistics. The only column that could not be filled from this website was redzone, which had already been excluded as ESPN did not record this statistic until 2006. The following columns were in the preprocessed data set:

Variables						
date	home	away	first downs	third downs	fourth downs	passing yards
rushing yards	total yards	comp att	sacks	rushing attempts	fumbles	interceptions
turnovers	penalties	redzone	drives	def st td	possession	score

Table 1: Variables in Initial NFL Dataset

Each variable except for date, home, and away were actually two columns that were followed by home or away. As a result, there are a total of 39 columns in the dataset; 37 with redzone excluded. An outcome column was created using "W" if the home team had a higher score than the away team and a "L" if this was

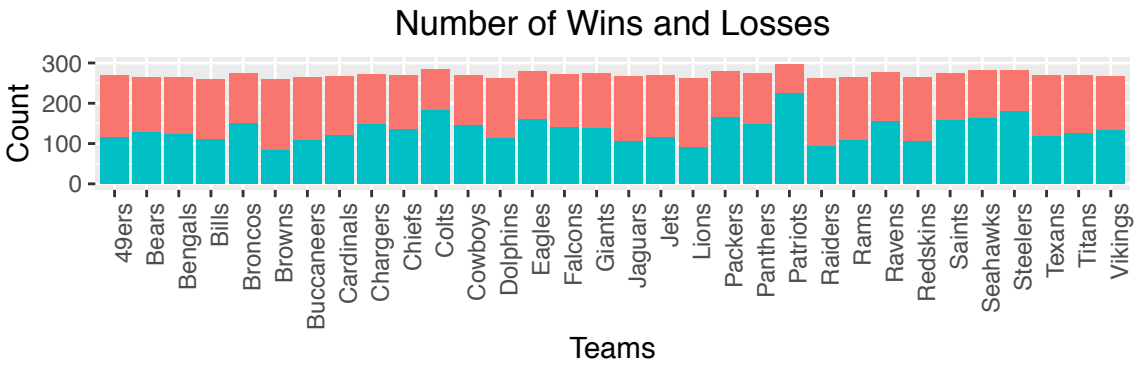
false. An 'outcomedifference' column was also created by subtracting the away team score from the home team score. There was a total of 10 tie matches out of 4631 games which were removed by removing all values which had an outcome difference of 0. The dataframe was split into two dataframes, one for all home team stats and one for all away team stats. Date, outcome and outcome difference, were added in each of these dataframes. Both dataframes also included sacks for home and away as they are relevant to the team in terms of offense and defense. Outcome was reversed in terms of "W" and "L" for the away dataframe in order to properly display the outcome of the away team. Outcome difference was reversed for away teams by multiplying by -1. A location column was added for home teams with a "H", and away teams with an "A". Columns were renamed to match between dataframes and the home and away dataframes were combined using rowbind. A column named "total" was added as the total number of games played. The total number of games was 8666, and the response variable, "outcome" was evenly split between "W" and "L" (4333 each).

Third down completions, third down attempts, fourth down completions, fourth down attempts, and completions attempted, completions made, were all separated out from the columns "third_downs", "fourth_downs", "comp_att". This was done by separating based on the "-" between these statistics formatted successful attempts-attempts. The same was done for number of sacks and yards gained/lost due to the sack. This was also done for penalties and penalty yards. The original columns for all the previously mentioned separated columns were removed since the new two columns contained the same information. Doing this made all the columns atomic, holding only one piece of information. 3 columns for percentage rates were created for completions, fourth down completions, and third down completions by dividing the columns for attempts by the total for the corresponding statistic and multiplying by 100. NANs that resulted from this were converted to zeroes.

In order to find the number of days between games, the data was grouped by team name, arranged by date, and the date of one game was subtracted from the date of the preceding game for each team. Games resulting with a 0 or number larger than 100 in the 'days' column were removed as these were start of season games which were not of interest in this analysis. The 'days' column was converted to a factor for 0 if the number of days between games was under 7, 1 if the number was 7 (the average number of rest days between games), and 2 if the amount of rest days was larger than 7 (the most being 22).

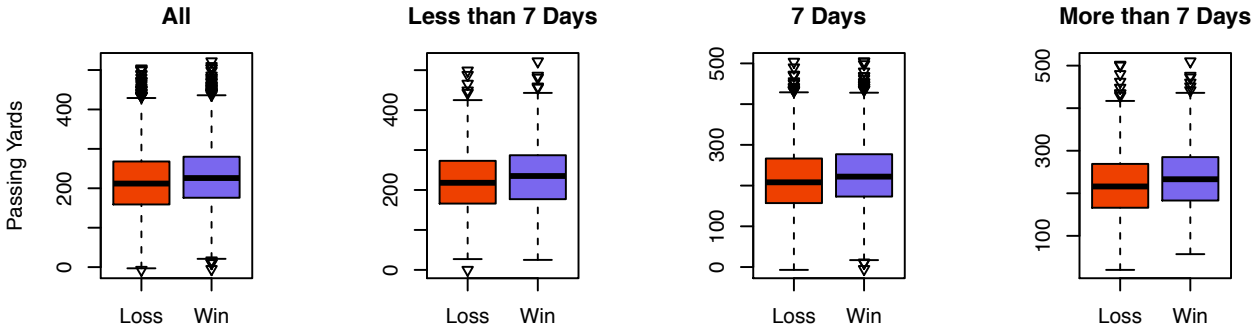
EDA

Stacked column plots were used to observe the distribution of wins and losses per team. Here we notice that in the last 17 years (including this season till week 6), the Patriots have the best record in the NFL. Meanwhile, the Bears unfortunately have the lowest win rate, that is statistically significant. This was done in order to later examine if these teams had intercepts relevant to their ratings during model validation. Team outcome by rest days was also plotted to observe if there was a difference in the win rate for teams by rest days. By comparing the Number of Wins and Losses plot to the Team Outcome by Rest Days plot (Appendix 2), we can identify that there seems to be no difference in outcome by rest days.



Below, boxplots were used to examine the distribution of passing yards by outcome. These were created to identify if any of the continuous variable distributions changed due to a difference in number of days

rested. Passing yards was used below, as it is generally a good indicator of if a team won or loss. Generally, the more yards a team has by the end of a game, the more likely that team is to have scored points. An increase in points scored is heavily correlated with an increase in win probability. No significant difference was observed, indicating that the number of days rested did not have an impact on how many passing yards a team had. Uneven game amount is due to the fact that not all teams make it into playoffs every year.



Binned plots of continuous predictors were also plotted to check for patterns and trends (Appendix 2). Most plots displayed a positive correlation with outcome. Other's displayed a negative correlation compared to outcome. This is to be kept in mind as the model is built. Variables might need to be transformed accordingly. This will be checked in the model validation stage.

Model Building

A full model was made using all the variables except date, score, outcome difference, and total as these columns served no value in the model. Total yards was removed as it was simply the addition of rushing yards and passing yards. Forward, backward, and stepwise AIC and BIC functions were used to help determine which variables were significant. BIC was primarily used due to the number of variables with some additions that were significant in the AIC selections. Variables were further eliminated manually. The first model generated after selection methods contained the following predictors: rushing attempts, rushing yards, total fourth downs, completed fourth downs, interceptions, passing yards, defensive touchdowns, yards gained from sacks, fumbles, location (H or A), completion attempts, completions successful, completion rate, penalty yards, drives, yards lost due to sacks, days between games, team name, number of sacks given, number of sacks received, third down conversion rate, and third down completions. What was not found to be significant was number of penalties, number of turnovers, fourth downs rate, and completion attempts. This model was tested with a Chi-Squared test against the previous full model as would found to be statistically better.

This model had a high VIF for: completion rate, total completions, successful completions, third downs rate, third down completions. To remedy this, total completions, attempted completions and third down completions were removed one by one with VIF being tested along each removal. Running the resulting model, it was found that third down rate was not significant and was subsequently removed in the next iteration of the model. VIF for the resulting model was under 2.8 for all predictors, which was acceptable. Numerous interactions were tested through trial and error, which were evaluated with a Chi-Square test to examine if they improved the model. One interaction was found between completion rate and location (H or A). This final model was shown to be significantly better by the Chi-Square test. The final model is shown below:

$$\begin{aligned} logit(Pr[Outcome = W]) = \beta_0 + \beta_1 RushingAttempts + \beta_2 RushingYards + \beta_3 FourthDownCompletions + \\ \beta_4 FourthDownsTotal + \beta_5 Interceptions + \beta_6 DaysBetweenGames + \beta_7 PassingYards + \\ \beta_8 DefensiveTouchdowns + \beta_9 SacksGivenYards + \beta_{10} Fumbles + \beta_{11} CompletionRate + \\ \beta_{12} Location + \beta_{13} PenaltyYards + \beta_{14} Drives + \beta_{15} SacksTakenYards + \beta_{16} TeamName \\ + \beta_{17} CompletionRate : Location \end{aligned}$$

Model Validation

A confusion matrix was generated and an ROC curve was created (Appendix 2) based off the final model. Model statistics are shown below. In the confusion matrix, 1 refers to a win and 0 refers to a loss.

	0	1
0	3709	554
1	624	3779

Table 2: Confusion Matrix of Model

Accuracy	Sensitivity	Specificity
0.8641	0.8721	0.8560

Table 3: Model Statistics

Based on the ROC curve, the model correctly predicts the outcome 93.9% of the time. The accuracy, sensitivity, and specificity, all above .859, indicate that the model is strong in predicting both wins and losses and does not have a high false positive or false negative rate. Binned residuals were plotted for the model and all continuous predictors (Appendix 2). All plots had few to no outliers, indicating that the model captures these predictors well. In addition no pattern was present in the binned residuals, meaning that the variables did not need to be changed to factors, transformed, or dropped from the model. K-fold validation was also used to ensure that the model was not overfitted and just performing well on the entire dataset. Random partitions equating to 40% of the data were sampled 10 times for this validation. The outcome of an accuracy of 0.8624, sensitivity of 0.8515, and specificity of 0.8734, indicated that the model was not overfitted as these are extremely similar values to those that were previously obtained from the entire final model.

Interpretation

The baseline of the model is less than 7 days of rest, and location away, the 49ers NFL team, with all other variables at 0. The exponentiated coefficients of the model identify that out of all the non-team name based individual predictors, fourth down completions increased the odds of winning the most by an average of 136.7% (95% CI, 103% - 177%) for every fourth down completion made relative to the baseline. This is expected since teams do not usually go for fourth downs unless they are close to obtaining enough yards for a first down or for a touchdown. Teams might also go for fourth downs when they must or else, they will lose the game by turning over possession by punting the ball. The next most influential variable on winning was defensive touchdowns which increased the odds of winning by an average of 109% (90% - 131%) relative to baseline. This is also unsurprising as a defensive touchdown only occurs during a kickoff, or an interception, where the player is able to run the ball all the way back for the touchdown. In the case of an interception, this equates to a turnover in addition to a touchdown which are both known to play big factors in winning games. Since this is the only real scoring metric in the model, it is clear why gaining 6 points on a touchdown (and usually an additional extra point through a kick) increases the odds of winning by such a high amount. Interestingly enough, the variables that contributed the most towards losing, relative to the baseline, were total fourth downs, interceptions and fumbles, which reduced the odds of winning by an average of 70.1% (66.7% - 74%), 54.8% (51.04% - 58.26%), and 51.12% (46.52% - 55.44%), respectively. Making sense of this, we can construe these results as teams with a high amount of fourth downs may not always make them. Teams that are far behind must go for more fourth downs and they may not always succeed. Interceptions and fumbles generally result in a change of possession, which gives the opposing team a chance to score, hence the reduction in the odds of winning. The most interesting results of the model are shown below. The full regression table and exponentiated confidence interval table are shown in Appendix 1.

Looking at the variables involved in the interaction between location and completion rate, it appears that the 95% confidence interval for location home varies almost exactly evenly across 0 indicating that being at home

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-10.0430	0.4984	-20.15	0.0000
fourth_downs_comp	0.8617	0.0796	10.82	0.0000
fourth_downs_tot	-1.2242	0.0629	-19.46	0.0000
int	-0.7940	0.0407	-19.53	0.0000
7_days	0.0533	0.1049	0.51	0.6113
less_than_7_days	0.0975	0.1248	0.78	0.4348
defensive_td	0.7376	0.0502	14.70	0.0000
fumbles	-0.7172	0.0466	-15.40	0.0000
completion_rate	0.0470	0.0056	8.37	0.0000
locationH	-0.3655	0.4590	-0.80	0.4258
Bears	0.6046	0.2704	2.24	0.0254
Patriots	1.6145	0.2775	5.82	0.0000
comp_att_rate:locationH	0.0160	0.0074	2.16	0.0309

Table 4: Coefficents of the Regression Model

does increase the odds of winning or losing, relative to baseline. For each percentage increase in completion rate, the odds of winning increase by an average of 4.8% (3.67% - 6%), relative to baseline. Looking at the interaction between these two variables, for each unit percentage increase in completion rate of quarterbacks, playing at home as opposed to away, increases the odds of winning further by an average of 1.6% (0.14% - 3.09%). This might be what is known as the homefield advantage, where the field is more familiar to the passer and receiver as they practice on this field most regularly. In addition, crowds are known to be quieter for home teams plays and loud when the away team is in possession in hopes of distracting the enemy.

In all of the iterations of the model, each category of the "days" predictor was never found to be significant. This suggests that the number of days between games does not impact the outcome of the game. Even though these days between games were not significant, it is interesting to note that on average, relative to baseline, which contained 'less than 7 days' of rest, '7 days' of rest suggested higher odds of winning. 'More than 7 days' of rest suggested even higher odds of winning than '7 days.' However, since these categorical variables have confidence intervals that range over 0 and are not significant, we cannot make any claims as to whether having more rest days actually improves the odds of winning. Finally, looking at the categorical variable "team_name," we see that the Patriots are have the highest odds of winning relative to the baseline with an average increase of 403% (191.17% - 765.77%). This confirms what was seen in the EDA as the Patriots have a much higher win rate than all the other teams in the last 17 years. Not all the teams in the model are significant but the Patriots team is. Out of all the statistically significant teams, the Bears have the least increase in odds of winning relative to the baseline, only increasing the odds of winning by an average of 83% (7.7% - 210%).

Conclusion

Overall, it appears that the number of days between games is not a significant contributor to the odds of a team winning a game. The most influential factors in this model that predict a team winning are fourth down completions, defensive touchdowns, and if that team is the Patriots. Total number of fourth downs, interceptions and fumbles lead the biggest increase in odds of losing. Limitations of this study include the missing redzone variable as it is quite important in determining whether a team scores or not. In addition many other variables could have been incorporated into the data set to yield a better model. Future work includes analyzing team specifically after 4 days between games and seeing if win streaks are more likely after bye weeks. Additional variables and datasets relating to the NFL and rule changes may also be interesting in determining how likely a team is to win. Rule changes, specifically those involving how players may tackle each other could be analyzed using a pre-post model.

Appendix 1: Tables of Regression Coefficients and Exponentiated Confidence Intervals

	2.5 %	97.5 %
(Intercept)	0.00	0.00
fourth_downs_comp	2.03	2.77
fourth_downs_tot	0.26	0.33
int	0.42	0.49
7_days	0.86	1.30
less_than_7_days	0.86	1.41
defensive_td	1.90	2.31
fumbles	0.45	0.53
completion_rate	1.04	1.06
locationH	0.28	1.71
Bears	1.08	3.11
Patriots	2.92	8.66
comp_att_rate:locationH	1.00	1.03

Table 5: Most Interesting Exponentiated Confidence Intervals of Odds

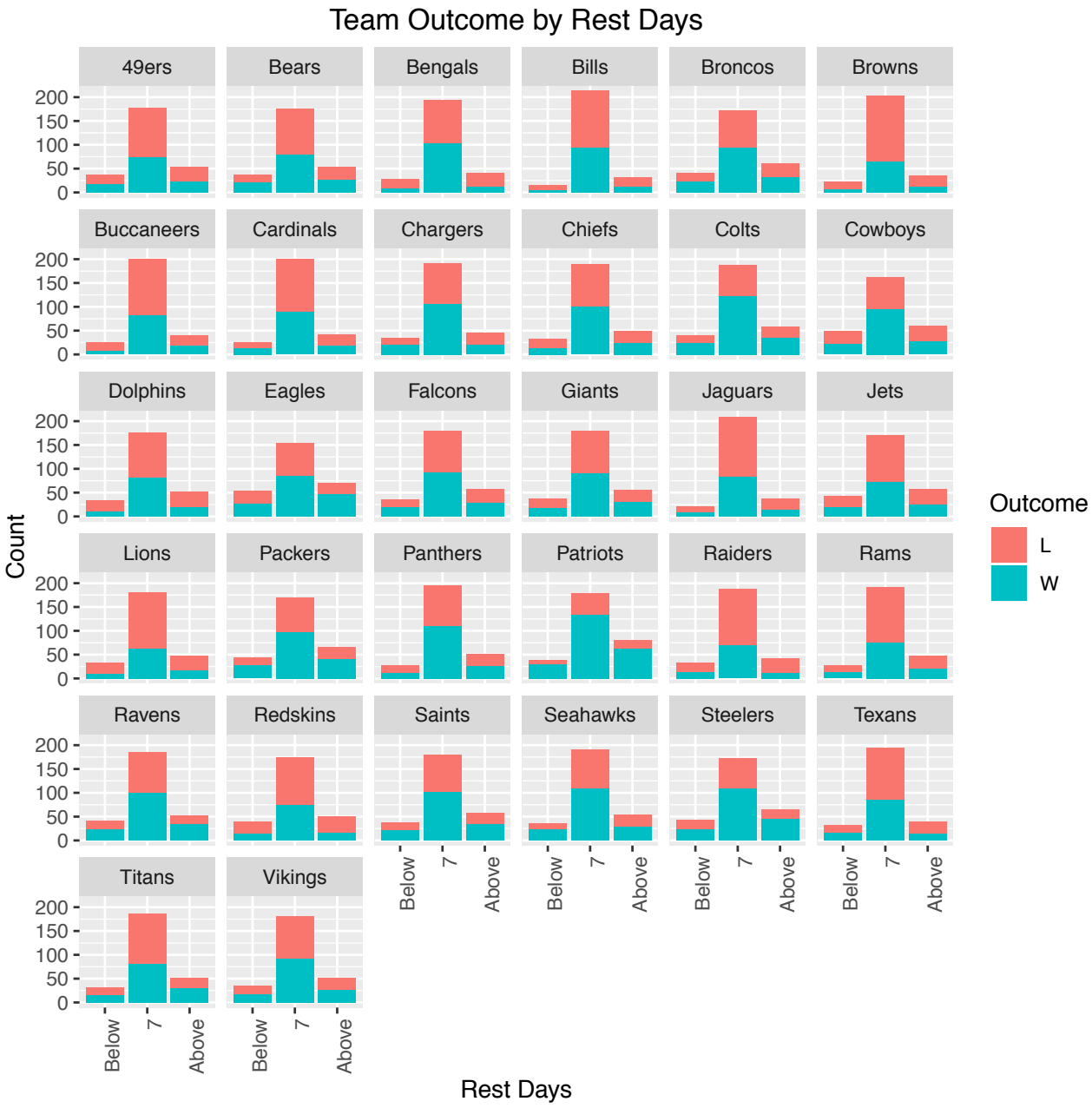
	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-10.0430	0.4984	-20.15	0.0000
rushing_attempts	0.2197	0.0081	27.13	0.0000
rushing_yards	-0.0022	0.0010	-2.10	0.0361
fourth_downs_comp	0.8617	0.0796	10.82	0.0000
fourth_downs_tot	-1.2242	0.0629	-19.46	0.0000
int	-0.7940	0.0407	-19.53	0.0000
days1	0.0533	0.1049	0.51	0.6113
days2	0.0975	0.1248	0.78	0.4348
passing_yards	0.0049	0.0005	8.94	0.0000
def_st_td	0.7376	0.0502	14.70	0.0000
sacks_given_yards	0.0471	0.0031	15.29	0.0000
fumbles	-0.7172	0.0466	-15.40	0.0000
comp_att_rate	0.0470	0.0056	8.37	0.0000
locationH	-0.3655	0.4590	-0.80	0.4258
penalty_yards	-0.0103	0.0013	-7.87	0.0000
drives	0.1117	0.0152	7.32	0.0000
sacks_taken_yards	-0.0269	0.0032	-8.35	0.0000
team_nameBears	0.6046	0.2704	2.24	0.0254
team_nameBengals	0.1108	0.2687	0.41	0.6799
team_nameBills	0.2513	0.2808	0.89	0.3709
team_nameBroncos	0.4921	0.2748	1.79	0.0734
team_nameBrowns	0.0723	0.2784	0.26	0.7952
team_nameBuccaneers	0.0616	0.2713	0.23	0.8202
team_nameCardinals	0.7182	0.2811	2.56	0.0106
team_nameChargers	0.4678	0.2737	1.71	0.0874
team_nameChiefs	0.0097	0.2658	0.04	0.9708
team_nameColts	1.1275	0.2715	4.15	0.0000
team_nameCowboys	0.6023	0.2646	2.28	0.0228
team_nameDolphins	0.4691	0.2723	1.72	0.0850
team_nameEagles	1.0903	0.2640	4.13	0.0000
team_nameFalcons	0.2878	0.2684	1.07	0.2835
team_nameGiants	0.4537	0.2681	1.69	0.0905
team_nameJaguars	-0.0451	0.2703	-0.17	0.8674
team_nameJets	-0.2523	0.2760	-0.91	0.3606
team_nameLions	0.2374	0.2719	0.87	0.3826
team_namePackers	0.8903	0.2664	3.34	0.0008
team_namePanthers	0.3998	0.2745	1.46	0.1453
team_namePatriots	1.6145	0.2775	5.82	0.0000
team_nameRaiders	0.3096	0.2764	1.12	0.2627
team_nameRams	0.4662	0.2740	1.70	0.0889
team_nameRavens	0.5966	0.2788	2.14	0.0324
team_nameRedskins	-0.1641	0.2766	-0.59	0.5529
team_nameSaints	0.7711	0.2687	2.87	0.0041
team_nameSeahawks	0.4917	0.2694	1.83	0.0680
team_nameSteelers	1.0598	0.2708	3.91	0.0001
team_nameTexans	-0.1848	0.2675	-0.69	0.4897
team_nameTitans	0.1905	0.2746	0.69	0.4877
team_nameVikings	0.1226	0.2721	0.45	0.6523
comp_att_rate:locationH	0.0160	0.0074	2.16	0.0309

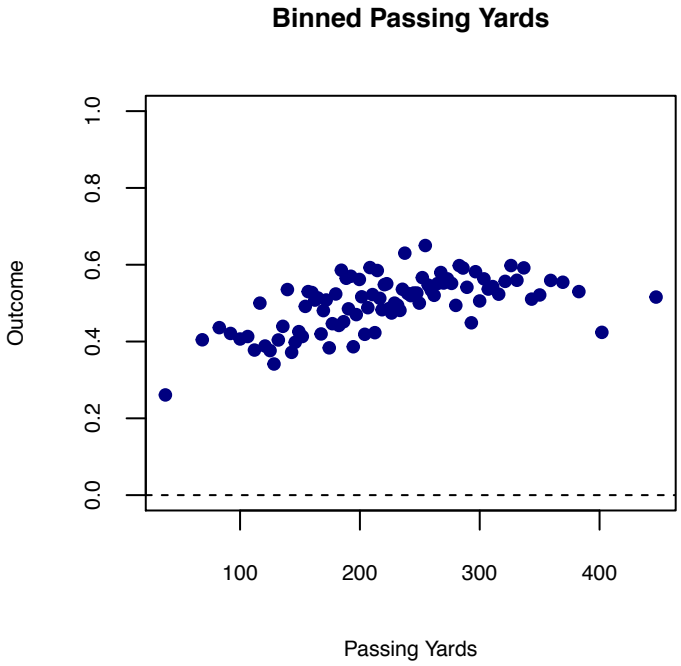
Table 6: Coefficents of the Regression Model

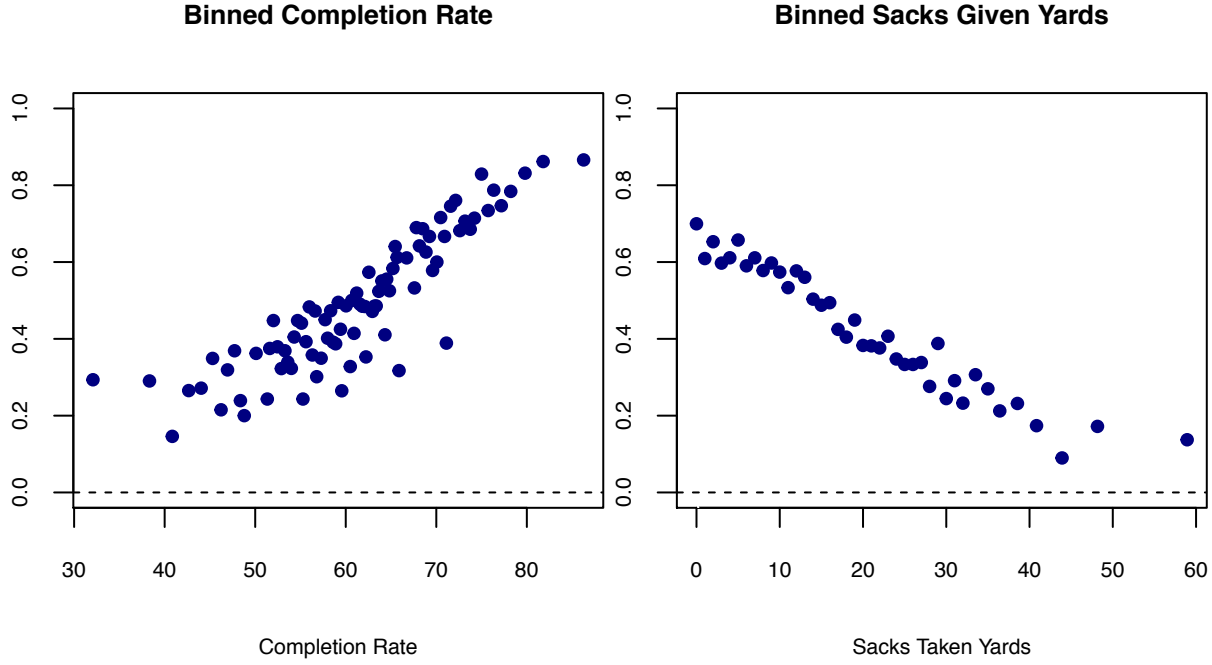
	2.5 %	97.5 %
(Intercept)	0.00	0.00
rushing_attempts	1.23	1.27
rushing_yards	1.00	1.00
fourth_downs_comp	2.03	2.77
fourth_downs_tot	0.26	0.33
int	0.42	0.49
days1	0.86	1.30
days2	0.86	1.41
passing_yards	1.00	1.01
def_st_td	1.90	2.31
sacks_given_yards	1.04	1.05
fumbles	0.45	0.53
comp_att_rate	1.04	1.06
locationH	0.28	1.71
penalty_yards	0.99	0.99
drives	1.09	1.15
sacks_taken_yards	0.97	0.98
team_nameBears	1.08	3.11
team_nameBengals	0.66	1.89
team_nameBills	0.74	2.23
team_nameBroncos	0.95	2.80
team_nameBrowns	0.62	1.86
team_nameBuccaneers	0.62	1.81
team_nameCardinals	1.18	3.56
team_nameChargers	0.93	2.73
team_nameChiefs	0.60	1.70
team_nameColts	1.81	5.26
team_nameCowboys	1.09	3.07
team_nameDolphins	0.94	2.73
team_nameEagles	1.77	4.99
team_nameFalcons	0.79	2.26
team_nameGiants	0.93	2.66
team_nameJaguars	0.56	1.62
team_nameJets	0.45	1.33
team_nameLions	0.74	2.16
team_namePackers	1.44	4.11
team_namePanthers	0.87	2.55
team_namePatriots	2.92	8.66
team_nameRaiders	0.79	2.34
team_nameRams	0.93	2.73
team_nameRavens	1.05	3.14
team_nameRedskins	0.49	1.46
team_nameSaints	1.28	3.66
team_nameSeahawks	0.96	2.77
team_nameSteelers	1.70	4.91
team_nameTexans	0.49	1.40
team_nameTitans	0.71	2.07
team_nameVikings	0.66	1.93
comp_att_rate:locationH	1.00	1.03

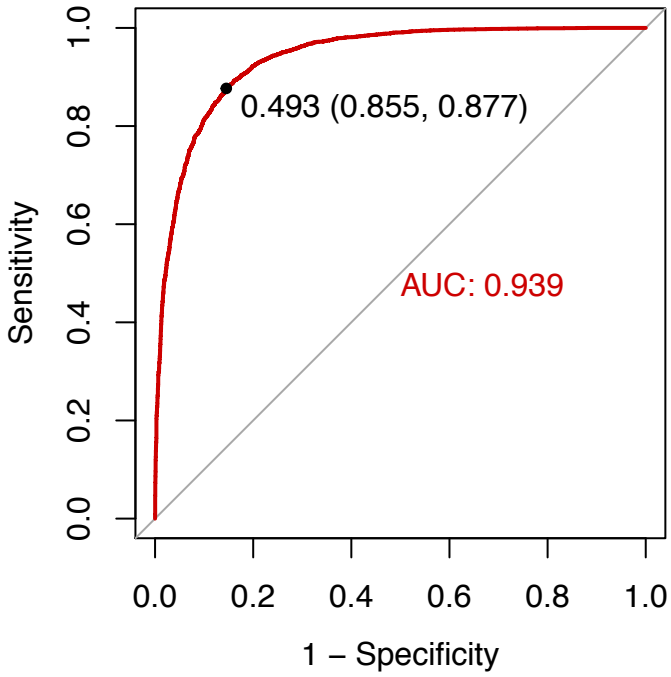
Table 7: Exponentiated Confidence Intervals of Odds

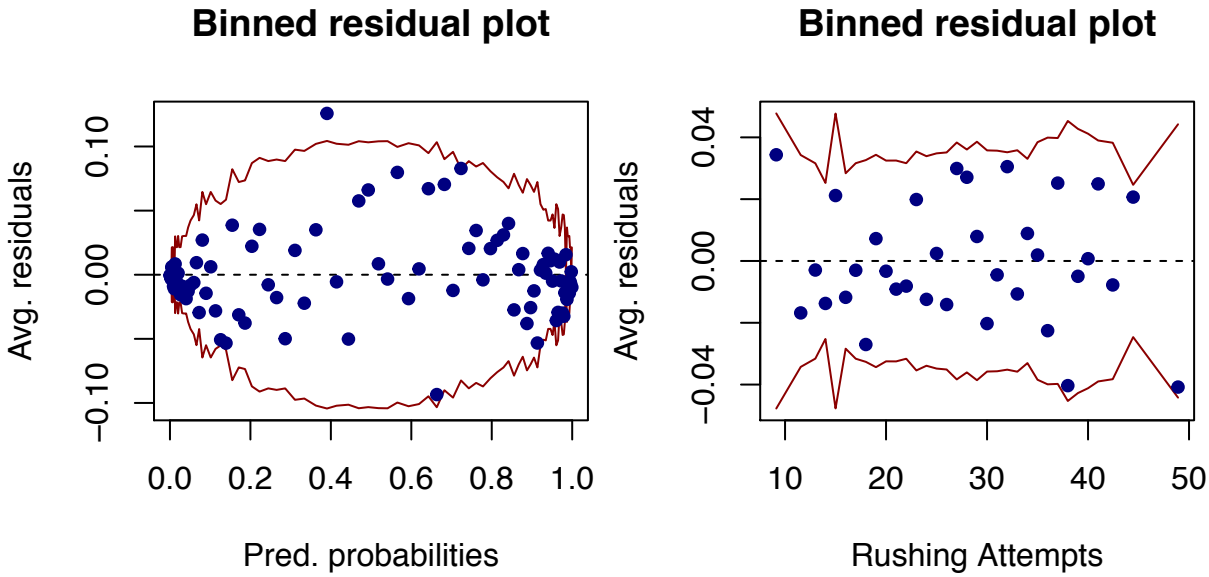
Appendix 2: Plots

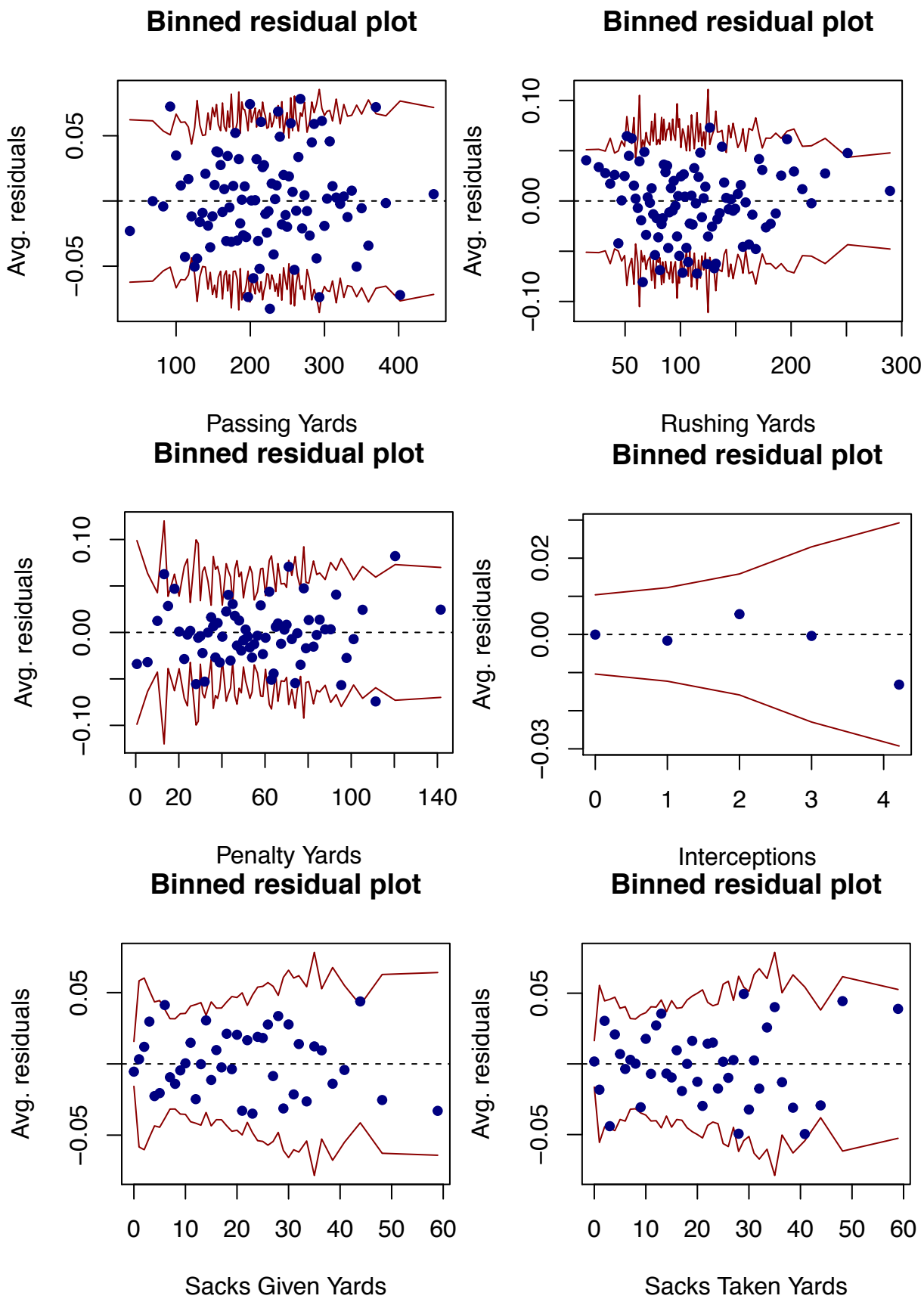


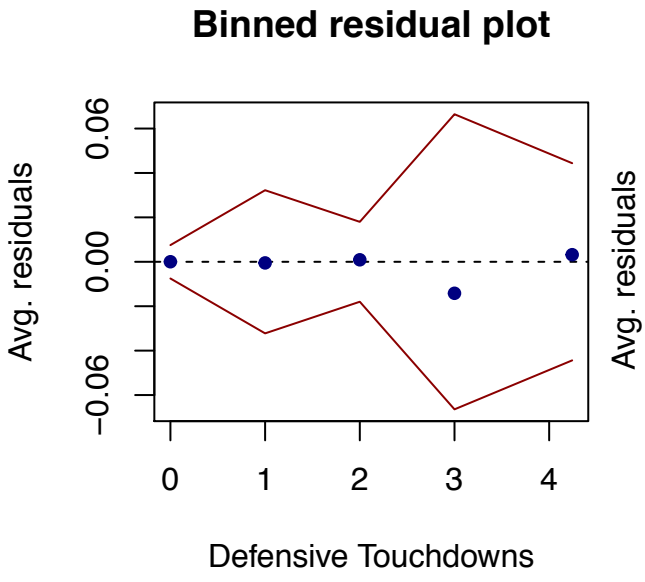

Binned Passing Yards
Outcome
0.0
0.2
0.4
0.6
0.8
1.0
100
200
300
400
Passing Yards

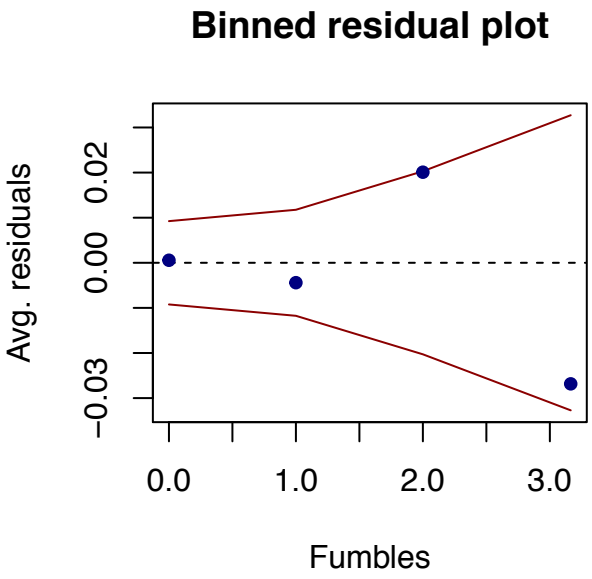

Binned Completion Rate
Binned Sacks Given Yards
0.0
0.2
0.4
0.6
0.8
1.0
30
40
50
60
70
80
Completion Rate
0
10
20
30
40
50
60
Sacks Taken Yards


0.493 (0.855, 0.877)
AUC: 0.939
Sensitivity
1 – Specificity
0.0
0.2
0.4
0.6
0.8
1.0


Binned residual plot
Avg. residuals
Pred. probabilities
−0.10
0.00
0.10
0.0
0.2
0.4
0.6
0.8
1.0
Binned residual plot
Avg. residuals
Rushing Attempts
−0.04
0.00
0.04
10
20
30
40
50


Binned residual plot
Avg. residuals
0.05
−0.05
100
200
300
400
Passing Yards
Binned residual plot
Avg. residuals
0.10
0.00
−0.10
50
100
200
300
Rushing Yards
Binned residual plot
Avg. residuals
0.10
0.00
−0.10
0
20
60
100
140
Penalty Yards
Binned residual plot
Avg. residuals
0.02
0.00
−0.03
0
1
2
3
4
Interceptions
Binned residual plot
Avg. residuals
0.05
−0.05
0
10
20
30
40
50
60
Sacks Given Yards
Binned residual plot
Avg. residuals
0.05
−0.05
0
10
20
30
40
50
60
Sacks Taken Yards


Binned residual plot
Avg. residuals
0.06
0.00
−0.06
0
1
2
3
4
Defensive Touchdowns


Binned residual plot
Avg. residuals
0.02
0.00
−0.03
0.0
1.0
2.0
3.0
Fumbles

Appendix 3: Code

```
library(readr)
library(stringr)
library(lubridate)
library(dplyr)
library(pROC)
library(ggplot2)
library(cowplot)
library(caret)
library(GGally)
library(lme4)
library(scales)
library(car)
library(lmerTest)
library(arm)
library(tidyr)
library(GGally)
library(kableExtra)
library(xtable)
library(gridExtra)
df1 = read.csv("/Users/N1/Desktop/702 - Modeling and Repr Data/Final Project/nfl_dataset_2002-2019week6
str(df1)

# 4268 x 39, 4631 x 39 after adding missing games
dim(df1)

df1 = add_row(df1, date = "2017-12-30", home = 'Redskins', away = 'Cowboys', first_downs_home = 22, firs

df1 = add_row(df1, date = "2010-12-23", home = 'Steelers', away = 'Panthers', first_downs_home = 22, fi

df1 = add_row(df1, date = "2012-01-01", home = 'Falcons', away = 'Buccaneers', first_downs_home = 22, f
# Make factor so can identify teams
df1$away = as.factor(df1$away)
df1$home = as.factor(df1$home)

# Change time to seconds for use
df1$possession_home = period_to_seconds(ms(df1$possession_home))
df1$possession_away = period_to_seconds(ms(df1$possession_away))

#df1$possession_away = as.duration(df1$possession_away)
#df1$possession_home = as.duration(df1$possession_home)
df1$date = as.Date(df1$date)


# Get count of games for each team
table(df1$away)
table(df1$home)
# Account for the missing 2 games ####################

# Add win and loss bar and difference in score
# Outcomedifference is home - away
df1$outcome = ifelse(df1$score_home > df1$score_away, 'W', 'L')
```

```r
table(df1$outcome)
df1$outcomedifference = df1$score_home - df1$score_away
df1 = df1[df1$outcomedifference != 0,]


# Split into two data frames so can see the win loss per team (Redzone was excluded since it is not pro
df_home = dplyr::select(df1, home, date, first_downs_home, third_downs_home, fourth_downs_home, passing


df_away = dplyr::select(df1, away, date, first_downs_away, third_downs_away, fourth_downs_away, passing

# Rename all columns so they match
colnames(df_home) <- c("team_name", "date", "first_downs", "third_downs", "fourth_downs", "passing_yards

colnames(df_away) <- c("team_name", "date", "first_downs", "third_downs", "fourth_downs", "passing_yards

# Marking win and loss outcomes and making outcomedifference the amount lost by
df_away$outcome = ifelse(df_away$outcome == 'W', 'O', 'L')
df_away$outcome = ifelse(df_away$outcome == 'L', 'W', 'L')
df_away$outcomedifference = df_away$outcomedifference * (-1)

# Make sure that they are even
table(df_home$outcome)
table(df_away$outcome)

table(df_home$outcomedifference)
table(df_away$outcomedifference)

# Make a home and away column
df_home$location = "H"
df_away$location = "A"

# Comebine dataframes
df_both = rbind(df_home, df_away)
df = df_both
total = table(df$team_name)
total = as.data.frame(total)
names(total)[1]<-paste("team_name")
names(total)[2]<-paste("total")
total

df = merge(df, total)

# Split into completions and total
df$third_downs_comp = as.numeric(word(df$third_downs,1,sep = "-"))
df$third_downs_tot = as.numeric(word(df$third_downs,2,sep = "-"))

df$fourth_downs_comp = as.numeric(word(df$fourth_downs,1,sep = "-"))
df$fourth_downs_tot = as.numeric(word(df$fourth_downs,2,sep = "-"))

df$comp_att_comp = as.numeric(word(df$comp_att,1,sep = "-"))
df$comp_att_tot = as.numeric(word(df$comp_att,2,sep = "-"))
```

```r
# Split penalties into -yards and number of penalties
df$penalty_yards = as.numeric(word(df$penalties,2,sep = "-"))
df$penalties = as.numeric(word(df$penalties,1,sep = "-"))

# Split sacks_taken into -yards and number of sacks
df$sacks_taken_yards = as.numeric(word(df$sacks_taken,2,sep = "-"))
df$sacks_taken = as.numeric(word(df$sacks_taken,1,sep = "-"))

# Split sacks_given into -yards and number of sacks
df$sacks_given_yards = as.numeric(word(df$sacks_given,2,sep = "-"))
df$sacks_given = as.numeric(word(df$sacks_given,1,sep = "-"))

# Rates
df$comp_att_rate = df$comp_att_comp/df$comp_att_tot * 100
df$fourth_downs_rate = df$fourth_downs_comp/df$fourth_downs_tot
df$third_downs_rate = df$third_downs_comp/df$third_downs_tot

# Make NaN zeroes
df$comp_att_rate[is.nan(df$comp_att_rate)] = 0
df$third_downs_rate[is.nan(df$third_downs_rate)] = 0
df$fourth_downs_rate[is.nan(df$fourth_downs_rate)] = 0

# Check what needs to be changed
df$outcome = ifelse(df$outcome == 'W', 1, 0)
df$outcome = ifelse(df$outcome == 1, 'W', 'L')
df$location = as.factor(df$location)
df$outcome = as.factor(df$outcome)

str(df)
# Calculate the days between last game for each team
df = df %>%
    group_by(team_name) %>%
    arrange(date, .by_group = TRUE) %>%
    mutate(days = date - lag(date, default = first(date)))

# Do not include start of season games
which(df$days == 0)
df[270,]
df = subset(df, days < 100)
df = subset(df, days > 0)
table(df$days)
# The 22 days is a post season game which did indeed take place, there is nothign wrong with the data h
which(df$days == 22)
df[3048,]

# Convert to factor for less than 7 days, 7 days, more than 7 days
df$days[df$days<7] <- 0
df$days[df$days==7] <- 1
df$days[df$days>=8] <- 2
table(df$days)
df$days = as.factor(df$days)
# OUTCOME BY TEAM NAME
ggplot(df, aes(x = team_name, fill = outcome)) +
```

```
  geom_bar(position = 'stack') +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  xlab("Teams") +
  ylim(0, 300) +
  ylab("Count") +
  ggtitle("Number of Wins and Losses") +
  theme(plot.title = element_text(hjust = 0.5)) +
  labs(fill = "Outcome")

# SACKS GIVEN
df$days = as.numeric(df$days)
ggplot(df, aes(x = outcome, y = sacks_given_yards/total, fill = outcome)) +
  geom_col() +
  ggtitle("Average Sacks per ") +
  theme(plot.title = element_text(hjust = 0.5)) +
  facet_wrap(~team_name)

# SACKS TAKEN
ggplot(df, aes(x = outcome, y = sacks_taken_yards/total)) +
  geom_col() +
  facet_wrap(~team_name)

# FUMBLES
ggplot(df, aes(x = outcome, y = fumbles/total)) +
  geom_col() +
  facet_wrap(~team_name)

str(df)
ggplot(df, aes(x = days, fill = outcome)) +
  geom_bar(position = "stack") +
  labs(y = "Count", fill="outcome") +
  scale_fill_discrete(name = "Outcome", labels = c("L", "W")) +
  xlab("Rest Days") +
  scale_x_discrete(labels=c("0" = "Below", "1" = "7", "2" = "Above")) +
  facet_wrap(~team_name) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  ggtitle("Team Outcome by Rest Days") +
  theme(plot.title = element_text(hjust = 0.5))

ggplot(df, aes(x = days, fill = outcome)) +
  geom_bar(position = "stack") +
  labs(y = "Count", fill="outcome") +
  scale_fill_discrete(name = "Outcome", labels = c("L", "W")) +
  xlab("Rest Days") +
  scale_x_discrete(labels=c("0" = "Below", "1" = "7", "2" = "Above")) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  ggtitle("Outcome by Rest Days") +
  theme(plot.title = element_text(hjust = 0.5)) +
  facet_wrap(~team_name)

ggplot(df, aes(x = days, fill = outcome)) +
  geom_bar(position = "stack") +
  labs(y = "Count", fill="outcome") +
```

```
  scale_fill_discrete(name = "Outcome", labels = c("L", "W")) +
  xlab("Days") +
  scale_x_discrete(labels=c("0" = "Below 7", "1" = "7",
                            "2" = "Above 7"))

ggplot(df, aes(x = outcome, fill = days)) +
  geom_bar(position = "stack") +
  scale_y_continuous(labels = scales::percent) +
  labs(y = "Count", fill="Days") +
  facet_wrap(~team_name)
table(df$outcome)

ggplot(df, aes(x = outcome, y = total_yards, fill = days)) +
  geom_boxplot() +
  theme_cowplot(font_size = 12) +
  xlab("Outcome") +
  ylab("Total Yards") +
  scale_fill_discrete(name = "Days", labels = c("Below 7", "7", "Above 7"))
df$outcome = as.factor(df$outcome)
df$days = as.factor(df$days)
#TEAM OUTCOME
ggplot(df, aes(x = team_name, fill = outcome)) +
  geom_bar(position = 'stack') +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  xlab("Teams") +
  ylim(0, 300) +
  ylab("Count") +
  ggtitle("Number of Wins and Losses") +
  theme(plot.title = element_text(hjust = 0.5)) +
  labs(fill = "Outcome") +
  theme(legend.title = element_blank()) +
  theme(legend.position = "none")
ifelse(df$outcome == 'W', 1, 0)
df$outcome = as.numeric(df$outcome)
df$outcome = df$outcome -1
par(mar=c(3, 4.1, 3, 3), mfrow=c(1,4))
boxplot(passing_yards~outcome, data = df, ylab = "Passing Yards", pch=25,xaxt='n',
        xlab="Outcome",col = c("orangered2","slateblue2"), cex = 0.85, main ="All", cex.main = 1.1);axis
boxplot(passing_yards~outcome, data=df, subset = days==0, ylab=NA,
        xlab="Outcome",col=c("orangered2","slateblue2"), xaxt='n',
        pch = 25, cex = 0.85, main ="Less than 7 Days", cex.main = 1.1);axis(1,at = c(1,2), labels = c(


boxplot(passing_yards~outcome, data=df, subset = days==1, ylab=NA,
        xlab="Outcome",col=c("orangered2","slateblue2"), xaxt='n',
        pch = 25, cex = 0.85, main ="7 Days", cex.main = 1.1);axis(1,at = c(1,2), labels = c("Loss", "W


boxplot(passing_yards~outcome, data=df, subset = days==2, ylab=NA,
        xlab="Outcome",col=c("orangered2","slateblue2"), xaxt='n',
        pch = 25, cex = 0.85, main ="More than 7 Days", cex.main = 1.1);axis(1,at = c(1,2), labels = c(
df$outcome = as.factor(df$outcome)
df <- subset(df, select = -c(comp_att, turnovers, penalties, third_downs, fourth_downs))
```

```r
df$days = as.factor(df$days)
glm0 = glm(outcome ~ .-date-score-outcomedifference-total_yards-total, data = df, family = binomial)
summary(glm0)
#df$days[df$days<7] <- 0
#df$days[df$days==7] <- 1
#df$days[df$days>=8] <- 2
#table(df$days)
glm1 = glm(outcome ~ .-date-score-outcomedifference-total-total_yards, data = df, family = binomial)
summary(glm1)
FullModel = glm1
NullModel = glm(outcome~1, df, family = binomial)

#AIC
#Forward with AIC
AIC_forward <- step(NullModel, scope = formula(FullModel),trace = 0, direction="forward")
AIC_forward$call
summary(AIC_forward)

#Backward with AIC
AIC_backward <- step(FullModel,direction="backward",trace=0)
AIC_backward$call
summary(AIC_backward)

#Stepwise AIC
AIC_stepwise <- step(NullModel, scope = formula(FullModel),direction="both",trace=0)
AIC_stepwise$call
summary(AIC_stepwise)


#BIC
#Forward with BIC
n = nrow(df)
BIC_forward <- step(NullModel, scope = formula(FullModel),direction="forward", trace = 0, k = log(n))
BIC_forward$call
summary(BIC_forward)

#Backward with BIC
n = nrow(df)
BIC_backward <- step(FullModel,direction="backward",trace=0, k = log(n))
BIC_backward$call
summary(BIC_backward)

#Stepwise BIC
BIC_stepwise <- step(NullModel, scope = formula(FullModel),direction="both",trace=0, k = log(n))
BIC_stepwise$call
summary(BIC_stepwise)

# Based on model selection above, these variables were excluded

#Model
glm2 = glm(outcome ~ rushing_attempts + rushing_yards + fourth_downs_tot + fourth_downs_comp + int + pas
anova(glm1, glm2, test ='Chisq')
summary(glm2)
```

```r
#VIF for comp_att_comp and comp_att_tot is high
vif(glm2)

# Model with comp_att_rate used instead due to vif
# Sacks_given, sacks_taken removed due to vif

glm2_5 = glm(outcome ~ rushing_attempts + rushing_yards + fourth_downs_comp + fourth_downs_tot + int + p
summary(glm2_5)

vif(glm2_5)

# third_downs_comp removed due to vif
glm2_6 = glm(outcome ~ rushing_attempts + rushing_yards + fourth_downs_comp + fourth_downs_tot + int + p

#Third_down_rate not sig
summary(glm2_6)

# This model is better than previous and the vif problem has been solved
anova(glm2_5, glm2_6, test ='Chisq')
vif(glm2_6)

glm3 = glm(outcome ~ rushing_attempts + rushing_yards + fourth_downs_comp + fourth_downs_tot + int + pas
summary(glm3)
# vif looks good
vif(glm3)

# Adding days:team_name interaction does not help the model
glm4 = glm(outcome ~ rushing_attempts + rushing_yards + fourth_downs_comp + fourth_downs_tot + int + pas
summary(glm4)
anova(glm3, glm4, test = 'Chisq')

# Rushing_attempts:Rushing_yards does not help
glm5 = glm(outcome ~ rushing_attempts*rushing_yards + fourth_downs_comp + fourth_downs_tot + int + passi
summary(glm5)
anova(glm3, glm5, test = 'Chisq')

#Fourth_down interaction does not help
glm6 = glm(outcome ~ rushing_attempts + rushing_yards + fourth_downs_comp*fourth_downs_tot + int + passi
summary(glm6)
anova(glm3, glm6, test = 'Chisq')

#Interaction between days and sacks_taken_yards almost helps
glm6_5 = glm(outcome ~ rushing_attempts + rushing_yards + fourth_downs_comp + fourth_downs_tot + int + p
summary(glm6_5)
anova(glm3, glm6_5, test = 'Chisq')

# Penalty_yards:days doesn't help
# days:drives - no
# days:int - no
# days:passing_yards - no
# days:def_st_td - no
# days:fumbles - no
# days:int - no
```

```r
# location * penalty_yards - no
#comp_att_rate: location - yes

# comp_att_rate:location interaction
glm7 = glm(outcome ~ rushing_attempts + rushing_yards + fourth_downs_comp + fourth_downs_tot + int + da
summary(glm7)
anova(glm3, glm7, test = 'Chisq')

# none for location*penalty yards
glm8 = glm(outcome ~ rushing_attempts + rushing_yards + fourth_downs_comp + fourth_downs_tot + int + da
summary(glm8)
anova(glm7, glm8, test = 'Chisq')

# VIF is only high for the interaction term/what was used in the interaction term which is fine
vif(glm7)
### FINAL MODEL = glm7

# Model AUC
Conf_mat <- confusionMatrix(as.factor(ifelse(fitted(glm7) >= 0.5, '1', '0')),
                            df$outcome,positive = "1")
invisible(roc(df$outcome,fitted(glm7),plot=T,print.thres="best",legacy.axes=T,
              print.auc =T,col="red3"))
Conf_mat$overall["Accuracy"];Conf_mat$byClass[c("Sensitivity","Specificity")]
xtable(Conf_mat$table)
q = as.data.frame(Conf_mat$overall["Accuracy"])
b = as.data.frame(Conf_mat$byClass[c("Sensitivity","Specificity")])
#tables = cbind(q, b)
#xtable(tables)
# Model Accuracy is 0.8758
set.seed(234)
Train <- createDataPartition(df$outcome, p=0.4, list=FALSE)
training <- df[ Train, ]
testing <- df[ -Train, ]

ctrl <- trainControl(method = "repeatedcv", number = 10, savePredictions = TRUE)

mod_fit <- train(outcome ~ rushing_attempts + rushing_yards + fourth_downs_comp + fourth_downs_tot + in

pred = predict(mod_fit, newdata=testing)
confusionMatrix(data=pred, testing$outcome)
summary(glm7)
confint.default(glm7)
z = exp(confint.default(glm7))
exp(coef(glm7))
z = as.data.frame(z)
xtable(z)
summary(glm7)
#TEAM OUTCOME BY REST DAYS
ggplot(df, aes(x = days, fill = outcome)) +
  geom_bar(position = "stack") +
  labs(y = "Count", fill="outcome") +
  scale_fill_discrete(name = "Outcome", labels = c("L", "W")) +
  xlab("Rest Days") +
```

```
  scale_x_discrete(labels=c("0" = "Below", "1" = "7", "2" = "Above")) +
  facet_wrap(~team_name) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  ggtitle("Team Outcome by Rest Days") +
  theme(plot.title = element_text(hjust = 0.5))
ifelse(df$outcome == 'W', 1,0)
df$outcome = as.numeric(df$outcome) -1

binnedplot(y=df$outcome, df$passing_yards,xlab="Passing Yards",ylim=c(0,1),col.pts="navy", ylab = "Outc
binnedplot(y=df$outcome, df$comp_att_rate,xlab="Completion Rate",ylim=c(0,1),col.pts="navy", ylab = NA,
binnedplot(y=df$outcome, df$sacks_taken_yards,xlab="Sacks Taken Yards",ylim=c(0,1),col.pts="navy", ylab
invisible(roc(df$outcome,fitted(glm7),plot=T,print.thres="best",legacy.axes=T,
              print.auc =T,col="red3"))
glm7_residuals <- residuals(glm7, "resp")
binnedplot(x=fitted(glm7), y=glm7_residuals ,xlab="Pred. probabilities", col.int="red4", ylab="Avg. resi
binnedplot(df$rushing_attempts,residuals(glm7,"resp"),xlab="Rushing Attempts", col.int="red4",ylab="Avg
binnedplot(df$passing_yards,residuals(glm7,"resp"),xlab="Passing Yards", col.int="red4",ylab="Avg. resi
binnedplot(df$rushing_yards,residuals(glm7,"resp"),xlab="Rushing Yards", col.int="red4",ylab="Avg. resi
binnedplot(df$penalty_yards,residuals(glm7,"resp"),xlab="Penalty Yards", col.int="red4",ylab="Avg. resi
binnedplot(df$int,residuals(glm7,"resp"),xlab="Interceptions", col.int="red4",ylab="Avg. residuals",mai
binnedplot(df$sacks_given_yards,residuals(glm7,"resp"),xlab="Sacks Given Yards", col.int="red4",ylab="A
binnedplot(df$sacks_taken_yards,residuals(glm7,"resp"),xlab="Sacks Taken Yards", col.int="red4",ylab="A
binnedplot(df$def_st_td,residuals(glm7,"resp"),xlab="Defensive Touchdowns", col.int="red4",ylab="Avg. r
binnedplot(df$fumbles,residuals(glm7,"resp"),xlab="Fumbles", col.int="red4",ylab="Avg. residuals",main=
```