

226 Project Part II

Peter Karnchanapimonkul (phatk), Tyler Coleman (SUNET: colemant)

November 2019

0 Summary of Data Set and Results from Part I of Project

The data was aggregated on the Kaggle Wesbite for the [NFL Big Data Bowl](#) and contains 49 covariates including the continuous response variable of the total yard gain by a player after receiving a handoff. We transform the data set by creating new binary response variable called FirstDown to observe if the run play results in a new set of downs. This means, that if the yards gained on the play is more than or equal to the distance required to get a first down then they convert the first down and the binary variable is a 1 or 0 otherwise. For regression tasks, our 'best' model for regression task is stepwise backward regression and our 'best' model for classification task is logistic regression instead of AdaBoost and SVM because it has a very small increase in accuracy at the cost of interpretability.

One significant problem is missing data and cases for categorical variable that contains levels that only occur 1 or 2 times. For instance, 'special' offensive and defensive formations, GameWeather and Wind that were either missing, or contained misspellings such as "Mostly Sunny", "Mostly sunny", and "Mostly Sunny Skies" that would provide skewed estimates or caused errors in cross validation are removed since we felt there were enough other observations to be represent an accurate representation of the population model. In addition, there are many collinear variables and we further elaborate this point in 2.e. Overall, the final train set has 16,761 observation and 24 covariates and 2 response variables.

1 Prediction on Test Set

The test error or RMSE of the regression for the 'best' model, stepwise backward regression, is 6.7075. This is higher than the estimate of the RMSE in part 1 on the validation set of 6.3614. This increase is understandable because as the backward stepwise regression is downward biasing the RMSE even using cross-validation to estimate error. This is because inherently, we are overfitting to the data by choosing backward regression. Each covariate left out to reduce complexity, is done so on the training data - the only available data. From a frequentist perspective, it could have also been possible to get a different realization of data that after running backwards regression, chose a covariate that is not included in our best model instead of a covariate that is. Because of this, when running on the test data with the 'overfitted' coefficients, the RMSE is higher.

The test error of the 'best' classification model, logistic regression, is 0.18197. This is calculated by adding the false positive and false negative rate. Likewise to the continuous model, this is higher than the estimate of the test error in part 1 on the validation set of 0.1806.

2 Inference

2.1 Part a

Our model includes 14 covariates - 9 are significant (4 of the 9 are binary variables of the positions of the player). The coefficients that are statistically significant using a size of $\alpha = 0.05$ for our chosen model include: 1) categorical variable if the play was recorded in season 2017 or 2018, 2) variable of the yards needed for a first down, 3) ordered factor variable of the defenders in the box of the opposing team, 4) 4 certain positions of the running player for HB, FB, QB, and RB, 5) a transformed continuous variable HomeScoreAdvantage that describes the amount home team is winning or losing by, and 6) player's age. (See figure 1 of the appendix.)

In terms of our analysis, the results appear believable and understandable. Age, difference in score, and which position is actually carrying the ball all influence how many yards the play will ultimately

gain, and are significant at the 5% level. As mentioned in the data exploration of part 1 of the project, this interpretation was reasonable to us. An example for score difference is that the more a team is winning by, the more conservative they may play, which decreases yards - which was shown in our scatter-plot in part 1 of the project of score difference vs yard gained.

Furthermore, distance to first down and defenders in the box are both very significant per the r output at the .1% level. This high level of significance is congruent with our understanding of the objective of the problem as we have stated previously. The more defenders in the box, the more people there are to tackle the running back meaning less yards per play. Similarly, the distance to first down affect the type of run being called, and who is on the field. Although these coefficients are significant at the .1% level, it is possible the coefficient our model has is not the same as the 'true' model. By this we mean, even if there are more players in the box, it is possible that there are still many yards gained. This interpretation will be continued in 2f and inference.

2.2 Part b

When we fit our backward stepwise regression to our test data, not the same coefficients are significant. Compared to the same model on training data, Gameclock, possession team (team controlling the ball), distance, and down matter. Position of the player no longer matters. (See figure 2 of the appendix).

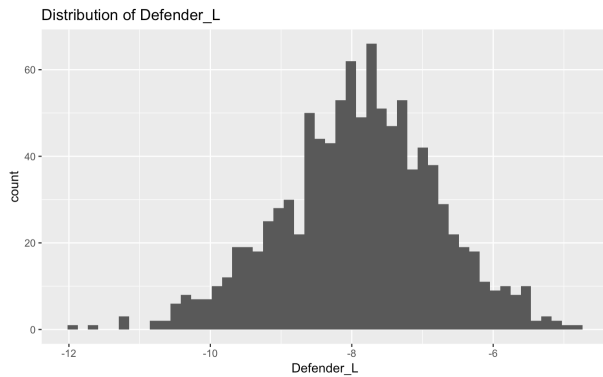
Explanations for why this may be the case are very similar to our analysis under Prediction on Test Set. In relation to the process to generate the coefficients and their significance level, given the data it is understandable that these variables are now significant at a certain threshold as we have a new realization of data. When fitting using backward step, with different data, we are biasing ourselves to the new data in order to best fit relative to the objective function. Distance is significant at the 5% level again and is again in line with our intuition. Although Gameclock, Possesion Team, and Down were not in our final model given from the train data, in our perception, these are still reasonable predictors of yards gained. An example is some teams have better offensive lines or and therefore may gain more yards. Therefore, the differences in what covariates were selected for the test model are also sensible to us.

2.3 Part c

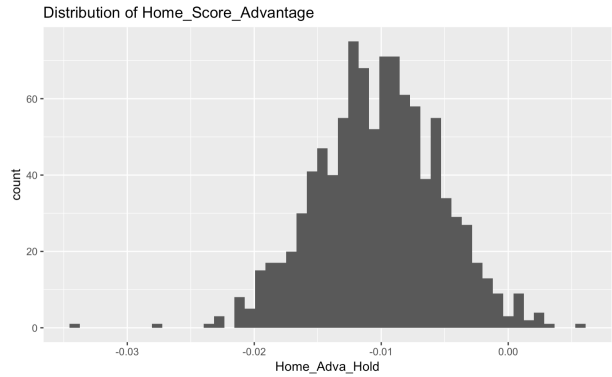
We now use the bootstrap method with 1000 iterations to estimate confidence intervals for each of the estimates in our chosen model - the backward stepwise model - in order to compare with standard R output. We used both Standard Error bootstrap and quantile bootstrap, keeping the n the same size as the train set and sampling each row. An abbreviation of the results are shown in the table below as well as two sample distributions for DefendersInTheBox (linear) and HomeScoreAdvantage. The full Confidence interval table can be found in the appendix.

95% Confidence Interval						
Coef	RMin	RMax	SEmin	SEmax	Q.min	Q.max
Season	0.10395	0.49446	0.29321	0.305762	0.099028	0.494932
Distance	0.05096	0.10401	0.076466	0.078129	0.051988	0.104668
DFense.L	-9.8250	-5.9844	-7.9654	-7.8282	-10.2462	-5.6964
DFense.4	-0.0281	0.49446	1.24975	1.34235	-0.14311	2.76499
DFense.5	-1.0993	1.03828	-0.053814	0.013056	-1.035426	1.024804
DFense.6	0.016685	1.6368	0.788092	0.838929	0.020102	1.622750
DFense.7	-0.8154	0.2694	-0.29082	-0.25420	-0.87713	0.31012
DFense.8	0.05387	0.62296	0.33013	0.35004	0.02921	0.64845
Pos.FB	-10.3878	-0.68641	-5.66184	-5.26025	-11.72697	0.86444
Pos.HB	-9.8507	-0.35647	-5.2157	-4.8165	-11.4572	1.1676
Pos.QB	-11.4106	-1.3744	-6.50034	-6.08900	-12.84262	0.18486
Pos.RB	-9.8083	-0.38414	-5.2206	-4.8214	-11.3616	1.2942
Temp	-0.01085	0.0001978	-0.00559	-0.00524	-0.01092	- .00012
Time.Diff	-0.0566	0.35592	0.136737	0.150228	-0.066454	0.349
Home.Adv	-0.0189	-0.001098	-0.01052	-0.00992	-.019586	-.000949

Interestingly, as shown above and further in the appendix, the 95 percent confidence interval shown by R and calculated by the Quantile approach are consistently similar in range. This contrasts with the Standard Error approach to the confidence interval which often gives what we believe are too tight of an interval. An example of this is in the histogram of the sample distribution of DefendersInBox.L which has a wide range in coefficient depending on the sample; however as shown in the SE columns for



(a) Defenders in Box Sample Distribution



(b) Home Score Advantage Distribution

the 95% confidence interval, the range is small (-7.9654, - 7.8282). As discussed in section, this small range may be a result that bootstrapping can only do so much with the data given; so the standard error bootstrap may not accurately reflect an accurate confidence interval. Because of this, we as data scientists interpreted the Quantile approach to be more applicable.

Also interestingly, the Quantile approach consistently is slightly wider than the confidence interval given through the R output. This could be due to similar ideology of cross-validation, as through bootstrapping we are able to test more variations of randomness, which gives an idea of what the confidence interval may be. Furthermore, some coefficients have a much larger range of values than other coefficients. For instance, a comparison between the confidence intervals DefendersInBox.Linear, and HomeScoreAdvantage distributions shown above. DefendersInBox.Linear is significant at the .1% level and is impactful but hard to measure exactly the effect while HomeScoreAdvantage is significant at the 5% level it may not serve as extremely important for prediction and has a tight interval near 0. Further comments on variables and interpreting the relationship of their effects will be discussed in 2F.

2.4 Part d

In our model that includes all covariates, the coefficients that are statistically significant using a size of $\alpha = 0.05$ include: 1) categorical variable if the play was recorded in season 2017 or 2018, 2) the team that the running player is from, 3) variable of the yards needed for a first down, 4) categorical variable of the offensive team positional grouping, 5) ordered factor variable of the defenders in the box of the opposing team, 6) the categorical variable of the visiting team as HOU of that game, 7) temperature, 8) humidity, and 9) a transformed continuous variable HomeScoreAdvantage that describes the amount home team is winning or losing by.

Therefore, the coefficients that are statistically significant in the full model change. We should note that many of the covariates' coefficients that are significant in the full model do not always make sense. For example, the values of the coefficients for temperature and humidity are -0.011 and -0.006. Hence, these two coefficients affect the values of yard very minimally and our backward stepwise regression correctly did not include these two covariates since it is also true that the temperature and humidity should not materially have a direct impact on the run plays.

2.5 Part e

A significant problem with the original data set is that there are too many collinear covariates about the information of the player, such as DisplayName, JerseyNumber, NflId, PlayerHeight, PlayerWeight, PlayerCollegeName, and Position, and information about the location of the stadium such as HomeTeamAbbr, VisitorTeamAbbr, Team, StadiumName, Location, StadiumType (stadium environment), Turf (field surface), GameWeather (categorical variable such as "Sunny and warm" and "Mostly Cloudy", Temperature, Humidity, and WindSpeed). We decide to only keep information about the player's height, weight, and position. We use this information instead of the player's name as we made the decision that the player characteristics were important for analysis instead of each player's name or 'superstar status'. We also kept temperature, humidity to describe the weather on that day. We also kept three categorical variables - HomeTeamAbbr, VisitorTeamAbbr, Team - to conclude which team this player is from, whether he is playing home (encapsulates field and field type) or away, and what defense he is facing.

Multiple hypothesis testing can generate false positives because we expect 5% of our coefficients to be significant incorrectly, as discussed in lecture . Because of this, we use bonferroni correction knowing

that $p = 13$, we expect the new cut-off p-value to be $\frac{0.05}{13} = 0.00384$ thereby cutting down on the false positive rate. With the bonferroni correction note that two covariates - distance and defenders in the box - are still significant from figure 1. This is indeed the case from figure 6. If we use Benjamini Hochberg correction, we can see that the same two covariates are also significant from our printout in the same figure.

As stated previously, by using backwards stepwise regression it is possible that we favorably biased the coefficients and their significance. However, since we have fitted these covariates on new data, validating their importance beyond our train data, we are more confident in their significance when discussing post selection inference, especially when considering our original intuition and data exploration of Distance and defenders in the box.

2.6 Part f

Even if all the coefficients make sense, we still cannot infer causal relationships since there are many other confounding variables to consider to use this model to accurately predict the yards gained. Having a better Tight End or another positions to block and create a path for the running back will result in an increase in yards.

Who is the running back? What is the development of the play. What coach is calling the play (head coach, assistant coach). Effort. etc. Some things that may be correlated with what we have but are still confounding.

An example with this is the covariate defenders in the box. Although it is highly significant given our data and model - including bonferroni correction - and is in line with our original intuition and data, there are too many confounding variables for us to interpret it as causal. Although as shown in our analysis and explanation, more defenders in the box is correlated with less yards, it is possible for a play the opposing team places all of there defenders in the box - leaving no other defenders to stop the player outside- and the player breaking free for a high number of positive yards. Because of events such as these, and the many realizations of events and data, it is hard for us to conclude a causal relationship. This uncertainty of the covariate mentioned previously is encapsulated by the large confidence interval range of around $(-10.2, -5.6)$.

3 Discussion

3.1 Part a

Our model can be used for basic prediction to simulate how many yards we expect a player of certain description from a specific team against another team. Our model is best used for inference because of the presence of too many confounding variables to consider in an actual football game.

Our model needs to include more covariates if we want to use it to calculate the odds of most rushing plays by a player in a season for betting websites or to use it as a point of reference whether the coach should go for a run play or passing play. Even though the prospect of real-time prediction is very alluring, our model does not have enough data for a specific player or data for a home team against another specific opposing team. In addition, in an actual game, we cannot keep track of many variables in real-time. For example, although the defensive coach can predict which offensive formation the opposing team might use, the quarterback always is the one who can call to change the scheme immediately.

However, in terms of inference we believe our model performs well and in line with our intuitive understanding. This includes analysis of significant coefficients. Coefficients that were processed as significant 'make sense' from our prior experience, even if the exact coefficient calculated is not representative of the population model coefficient, which is in line with our discussion of the linear relationship of defenders in the box and yards (from part 2f).

3.2 Part b

We do not expect our model to hold over time from season to season. There are always new players joining the league and retiring from injuries or old age. Additionally, teams are using new strategies supported by data analytics. Players are also outperforming and breaking old records with the usage of sports science, technology, and sports nutrition. For instance, teams with quarterbacks that have excellent speed such as Lamar Jackson from the Baltimore Raven are more willing to go for 2-points conversion than before or let a play 'break down' so that he can rush for more yards.

To summarize the previous examples, we believe the regime that we are trying to model is constantly changing - and it is a large draw to watching football. The population that we are trying to model may change game to game, or even throughout the same game as coaches may have several different schemes for one opponent. Because of this, it would be very difficult for us to make accurate predictions without constantly training the model. We would have to train the data in real time, and even then we may not be able to fit the model correctly, because as mentioned in part 3a, the quarterback can change the play at the line of scrimmage. Beyond that difficulty, some blocks and rush directions are determined by the players as the play develops and this information is not available beforehand.

3.3 Part c

We make several decisions throughout the data cleaning and analysis process that would be important for someone who is going to use the model to understand - the most likely users would be an NFL coach, sports analyst, or odds maker. Specifically that, in building our model we tried to make it as intuitive and applicable as possible. This means that we sacrificed some specificity for understanding and interpretation. Again, the example of our decision to use general running player characteristics of height and weight vs. using their name comes to mind. With this decision, we are better able to understand a runner with height 6 feet and weight 215, but lose out on the information of if they are a star or a rookie.

In line with the previous thought process is our difficulty to capture specificity, which would be important for a coach. There could be two rows with the same values for observation, with two different yards gained. This could be because in one instance the offense's best receiver lines up again the defense's worse corner back. A head coach or quarterback would recognize this, but our model interprets them the same. The same methodology could be said for the lack of interactive variables in the model and how interaction between variables may be very important during the real game. Examples such as these are why it is important for us to let the user know that their intuition and play calling is valuable beyond our model.

3.4 Part d

Expanding the point from part c, we would want to change the data collection process by collecting more data points for a specific team or player instead of collecting data points with many covariates that describe the field's environment or the appearance of a running player such as which college he is from. Collecting data this way will be more useful for a coach to predict if he should run when it is a third down and there is 4 yards to go. The obvious way to change the data collection process is also to collect data on confounding variables such as data on tight ends to see if the presence of a famous tight end results in more running plays, or on quarter backs to see when mobile quarterbacks use running plays whether it is on the first, second, or third down.

3.5 Part e

If we were to work with this dataset again, there are two main areas that we would attack differently. The first is to understand how significantly any one player - or set of players - can affect the number of yards gained. In our current analysis as we stated, we made the decision that the information on the height and weight characteristics to understand a type of player was more important for our understanding than was the specific player. However, if given another chance it would be very interesting to see if a best model including NFLID has better prediction than our current best model with NFL rusher characteristics. The second area would be how we train the data and when we are trying to predict. It would be interesting to train and test the data in a time-series fashion. Currently we train from any game in any season randomly sampled, and use that to predict. However, maybe our model would have better predictive power if we trained on the first and third quarter of the game and predicted on the second and fourth quarter. The thought behind this would be the players and coaches have a chance to 'learn' the opposing team and our model would have a chance to train, and then we would be able to predict on the following quarter. We would train on the first quarter of each half as during half-time the team may implement a new scheme.

4 Appendix

4.1 Summary of Coefficient and P-values

Figure 1: For chosen model stepwise backward regression on training set

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	10.21528	2.45999	4.15	0.00003303853959795	***
Season2018	0.29921	0.09962	3.00	0.0027	**
Distance	0.07749	0.01353	5.73	0.00000001040465146	***
DefendersInTheBox.L	-7.90479	0.97969	-8.07	0.00000000000000076	***
DefendersInTheBox.Q	1.73697	0.93874	1.85	0.0643	.
DefendersInTheBox.C	-1.43598	0.81786	-1.76	0.0791	.
DefendersInTheBox^4	1.30967	0.68268	1.92	0.0551	.
DefendersInTheBox^5	-0.03050	0.54527	-0.06	0.9554	
DefendersInTheBox^6	0.82678	0.41329	2.00	0.0455	*
DefendersInTheBox^7	-0.27303	0.27673	-0.99	0.3239	
DefendersInTheBox^8	0.33841	0.14517	2.33	0.0198	*
PositionFB	-5.53713	2.47472	-2.24	0.0253	*
PositionHB	-5.10359	2.42187	-2.11	0.0351	*
PositionQB	-6.39248	2.56011	-2.50	0.0125	*
PositionRB	-5.09622	2.40399	-2.12	0.0340	*
PositionTE	-3.26261	2.73287	-1.19	0.2326	
PositionWR	-3.51122	2.42122	-1.45	0.1470	
Temperature	-0.00532	0.00282	-1.89	0.0588	.
Humidity	-0.00470	0.00247	-1.90	0.0573	.
TimeDifference	0.14964	0.10524	1.42	0.1551	
HomeScoreAdvantage	-0.01002	0.00455	-2.20	0.0277	*
PlayerAge	-0.03798	0.01605	-2.37	0.0180	*

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Figure 2: Running chosen model on test set

```
Stepwise Backward Model on test set
Call:
lm(formula = Yards ~ GameClock + PossessionTeam + Down + Distance +
    DefendersInTheBox + Position + PlayerAge, data = NFL_DATA_HOLDOUT_Final)

Residuals:
    Min       1Q   Median       3Q      Max
-18.58  -3.37  -1.20   1.39   85.84

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   5.908389   1.980837   2.98  0.00287 **
GameClock     0.001019   0.000401   2.54  0.01105 *
PossessionTeamATL  2.504678   1.036391   2.42  0.01570 *
PossessionTeamBLT  1.515216   0.879689   1.72  0.08506 .
PossessionTeamBUF -0.028197   0.918678  -0.03  0.97552
PossessionTeamCAR  1.186497   0.948053   1.25  0.21082
PossessionTeamCHI  1.078039   0.930908   1.16  0.24691
PossessionTeamCIN -2.762882   1.744181  -1.58  0.11326
PossessionTeamCLV  2.877629   0.903983   3.18  0.00147 **
PossessionTeamDAL  1.015599   1.012357   1.00  0.31582
PossessionTeamDEN  0.546171   0.890764   0.61  0.53981
PossessionTeamDET  0.486587   0.931135   0.52  0.60130
PossessionTeamGB   0.944200   0.980930   0.96  0.33583
PossessionTeamHST  0.546959   0.898545   0.61  0.54275
PossessionTeamIND  0.477170   0.914433   0.52  0.60182
PossessionTeamJAX  1.346873   0.876690   1.54  0.12454
PossessionTeamKC   0.713865   0.929916   0.77  0.44273
PossessionTeamLA   1.807064   0.900506   2.01  0.04485 *
PossessionTeamLAC  1.779595   0.888103   2.00  0.04516 *
PossessionTeamMIA  0.235876   0.913372   0.26  0.79623
PossessionTeamMIN  0.606280   1.039262   0.58  0.55967
PossessionTeamNE   1.557747   0.875835   1.78  0.07538 .
PossessionTeamNO   2.353866   0.905271   2.60  0.00935 **
PossessionTeamNYG  2.197424   0.916251   2.40  0.01652 *
PossessionTeamNYJ -0.350460   0.888716  -0.39  0.69335
PossessionTeamOAK  1.662020   0.897781   1.85  0.06421 .
PossessionTeamPHI  1.791589   0.897973   2.00  0.04609 *
PossessionTeamPIT  0.778489   0.916755   0.85  0.39583
PossessionTeamSEA  0.097674   0.896580   0.11  0.91326
PossessionTeamSF   1.675230   0.909962   1.84  0.06570 .
PossessionTeamTB   0.251064   0.945813   0.27  0.79068
PossessionTeamTEN  0.778105   0.884290   0.88  0.37895
PossessionTeamWAS  0.188599   0.912825   0.21  0.83632
Down.L          0.296762   0.874675   0.34  0.73441
Down.Q          -0.710557   0.663333  -1.07  0.28415
Down.C          -0.790897   0.400499  -1.97  0.04836 *
Distance        0.076547   0.034218   2.24  0.02534 *
DefendersInTheBox.L -5.374533   1.453335  -3.70  0.00022 ***
DefendersInTheBox.Q  1.113366   1.286213   0.87  0.38675
DefendersInTheBox.C -1.887662   1.150727  -1.64  0.10100
DefendersInTheBox^4  1.369262   1.013648   1.35  0.17683
DefendersInTheBox^5 -0.655902   0.830582  -0.79  0.42975
DefendersInTheBox^6  0.576733   0.598469   0.96  0.33526
DefendersInTheBox^7 -0.351824   0.339640  -1.04  0.30032
PositionHB       1.777396   2.143329   0.83  0.40700
PositionQB      -3.263589   2.307068  -1.41  0.15726
PositionRB      -1.681884   1.365552  -1.23  0.21815
PositionTE      -4.068320   3.297659  -1.23  0.21739
PositionWR      -0.061202   1.524996  -0.04  0.96799
PlayerAge       -0.076123   0.039021  -1.95  0.05115 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.66 on 4088 degrees of freedom
Multiple R-squared:  0.0318,    Adjusted R-squared:  0.0202
F-statistic: 2.74 on 49 and 4088 DF,  p-value: 0.00000000117
```

Figure 3: Part of the printout of coefficients and p-values for the model that includes all covariates

Call:
lm(formula = Yards ~ ., data = NFL_DATA_TRAIN_Filtered_Final)

Residuals:
Min 1Q Median 3Q Max
-19.55 -3.14 -1.12 1.40 93.84

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.98852361	6.36598908	0.16	0.87660
X	-0.00097971	0.00196252	-0.50	0.61764
Y	-0.00000185	0.01417343	0.00	0.99990
Season2018	0.28706580	0.10597429	2.71	0.00676 **
Quarter	0.02764501	0.04402603	0.63	0.53006
GameClock	0.00020721	0.00018950	1.09	0.27421
PossessionTeamATL	0.33670854	0.70019035	0.48	0.63061
PossessionTeamBLT	1.27791156	0.65156844	1.96	0.04986 *
PossessionTeamBUF	0.73458788	0.62874990	1.17	0.24269
PossessionTeamCAR	1.04581505	0.65782103	1.59	0.11190
PossessionTeamCHI	0.62185325	0.63892902	0.97	0.33043
PossessionTeamCIN	-0.33692473	1.07394767	-0.31	0.75373
PossessionTeamCLV	0.77686762	0.63737541	1.22	0.22292
PossessionTeamDAL	0.84325085	0.70092697	1.20	0.22897
PossessionTeamDEN	1.02166064	0.60760301	1.68	0.09269 .
PossessionTeamDET	0.25235019	0.61078060	0.41	0.67949
PossessionTeamGB	0.89866527	0.63598607	1.41	0.15767
PossessionTeamHST	1.18270497	0.61783176	1.91	0.05560 .
PossessionTeamIND	1.10389627	0.59606918	1.85	0.06405 .
PossessionTeamJAX	0.50532274	0.62419223	0.81	0.41820
PossessionTeamKC	0.84888570	0.61827745	1.37	0.16978
PossessionTeamLA	0.21438364	0.58570181	0.37	0.71435
PossessionTeamLAC	0.56204935	0.63206775	0.89	0.37390

Figure 4: Printout of p-values of covariates that are significant $\alpha = 0.05$ for same model as above

Season2018	PossessionTeamBLT	PossessionTeamNO
0.2870658	1.2779116	1.3993310
PossessionTeamPIT	Down.L	Down.Q
1.3272332	-0.8408872	-0.6154830
Distance	OffensePersonnel1 RB, 0 TE, 3 WR,1 DB	OffensePersonnel1 RB, 0 TE, 4 WR
0.0731876	11.5303253	3.3478444
OffensePersonnel1 RB, 1 TE, 3 WR	OffensePersonnel1 RB, 2 TE, 1 WR,1 DL	OffensePersonnel1 RB, 2 TE, 1 WR,1 LB
2.6753443	3.1041609	6.9952040
OffensePersonnel1 RB, 2 TE, 2 WR	OffensePersonnel1 RB, 3 TE, 1 WR	OffensePersonnel2 QB, 2 RB, 1 TE, 1 WR
2.5773441	2.5511291	6.3600602
OffensePersonnel2 RB, 0 TE, 3 WR	OffensePersonnel2 RB, 1 TE, 2 WR	OffensePersonnel2 RB, 2 TE, 1 WR
2.8526605	2.7747298	3.2782210
OffensePersonnel6 OL, 1 RB, 0 TE, 3 WR	OffensePersonnel6 OL, 1 RB, 2 TE, 1 WR	OffensePersonnel6 OL, 2 RB, 0 TE, 2 WR
2.8215540	2.6364301	5.2470039
OffensePersonnel6 OL, 2 RB, 1 TE, 1 WR	DefendersInTheBox.L	DefendersInTheBox.Q
2.8386969	-7.4227564	3.1383908
VisitorTeamAbbrHOU	Temperature	Humidity
-1.3086588	-0.0108235	-0.0060117
HomeScoreAdvantage		
-0.0099095		

Figure 5: Printout of the p-values after applying Bonferroni and Benjamini-Hochberg corrections

Bonferroni Corection:

(Intercept)	Season2018	Distance	DefendersInTheBox.L	DefendersInTheBox.Q
0.00075988641075296	0.06145068323576464	0.00000023930698364	0.00000000000001745	1.0000000000000000
DefendersInTheBox.C	DefendersInTheBox^4	DefendersInTheBox^5	DefendersInTheBox^6	DefendersInTheBox^7
1.0000000000000000	1.0000000000000000	1.0000000000000000	1.0000000000000000	1.0000000000000000
DefendersInTheBox^8	PositionFB	PositionHB	PositionQB	PositionRB
0.45443048147518167	0.58117756343934435	0.80746110260739479	0.28832544652483055	0.78267116275268955
PositionTE	PositionWR	Temperature	Humidity	TimeDifference
1.0000000000000000	1.0000000000000000	1.0000000000000000	1.0000000000000000	1.0000000000000000
HomeScoreAdvantage	PlayerAge			
0.63775462372313607	0.41320217134274029			

Benjamini-Hochberg:

(Intercept)	Season2018	Distance	DefendersInTheBox.L	DefendersInTheBox.Q
0.00025329547025099	0.01536267080894116	0.00000011965349182	0.000000000000001745	0.09240928949852184
DefendersInTheBox.C	DefendersInTheBox^4	DefendersInTheBox^5	DefendersInTheBox^6	DefendersInTheBox^7
0.10707665907577482	0.09014700413743902	0.99881324941339322	0.08714115389371181	0.35469315705138776
DefendersInTheBox^8	PositionFB	PositionHB	PositionQB	PositionRB
0.06491864021074023	0.07086162485812622	0.07340555478249043	0.05766508930496610	0.07340555478249043
PositionTE	PositionWR	Temperature	Humidity	TimeDifference
0.26744104726469331	0.18770195278149382	0.09014700413743902	0.09014700413743902	0.18770195278149382
HomeScoreAdvantage	PlayerAge			
0.07086162485812622	0.06491864021074023			

4.2 Confidence Intervals of Coefficients

Below is the full table of confidence intervals.

95% Confidence Interval						
Coef	RMin	RMax	SEmin	SEmax	Q.min	Q.max
Season	0.10395	0.49446	0.29321	0.305762	0.099028	0.494932
Distance	0.05096	0.10401	0.076466	0.078129	0.051988	0.104668
DFense.L	-9.8250	-5.9844	-7.9654	7.8282	-10.2462	-5.6964
DFense.Q	-0.1031	3.577	1.65512	1.79089	-0.43364	3.92729
DFense.C	-3.039	0.167106	-1.48221	-1.36586	-3.33724	0.38925
DFense.4	-0.0281	0.49446	1.24975	1.34235	-0.14311	2.76499
DFense.5	-1.0993	1.03828	-0.053814	0.013056	-1.035426	1.024804
DFense.6	0.016685	1.6368	0.788092	0.838929	0.020102	1.622750
DFense.7	-0.8154	0.2694	-0.29082	-0.25420	-0.87713	0.31012
DFense.8	0.05387	0.62296	0.33013	0.35004	0.02921	0.64845
Pos.FB	-10.3878	-0.68641	-5.66184	-5.26025	-11.72697	0.86444
Pos.HB	-9.8507	-0.35647	-5.2157	-4.8165	-11.4572	1.1676
Pos.QB	-11.4106	-1.3744	-6.50034	-6.08900	-12.84262	0.18486
Pos.RB	-9.8083	-0.38414	-5.2206	-4.8214	-11.3616	1.2942
Pos.TE	-8.6193	2.0941	-3.4076	-2.9835	-9.6938	3.4088
Pos.WR	-8.257	1.23463	-3.6493	-3.2490	-10.0371	2.8009
Temp	-0.01085	0.0001978	- 0.00558645	- 0.00523635	- 0.01092161	- 0.00011977
Humid	-0.009553	0.000146	- 0.004979322	- 0.004366638	- 0.010160963	0.000011509
Time.Diff	-0.0566	0.35592	0.136737	0.150228	-0.066454	0.349025
Home.Adv	-0.0189	-0.001098	- 0.01052319	- 0.00992362	- 0.01958591	- 0.00094855
Age	-0.0694	-0.00652	- 0.0392192	- 0.0354083	- 0.0668316	- 0.0057647

MS&E 226 Project

Peter Karnchanapimonkul (SUNET: phatk), Tyler Coleman (SUNET: colemant)

IMPORT DATA AND SPLIT INTO TRAIN AND HOLDOUT

```
NFL_DATA <- read.csv(file = "train.csv", header = TRUE, sep=",")
NFL_DATA_Run_Observations <- NFL_DATA[(NFL_DATA$NflIdRusher == NFL_DATA$NflId) , ]

# split 80:20 for training:test
set.seed(121)
training_data = sample(nrow(NFL_DATA_Run_Observations), size = nrow(NFL_DATA_Run_Observations) * 0.8)
NFL_DATA_Train = NFL_DATA_Run_Observations[training_data, ]
NFL_DATA_Holdout = NFL_DATA_Run_Observations[-training_data, ] # holdout is remaining indices
```

DATA MANIPULATION AND CLEANING

Adding the covariates that we want to include, and modifying the dataframe

```
NFL_DATA_TRAIN_Modified <- NFL_DATA_Train

# Function to take the difference in time from the dataframe
timeDifference <- function(time) {
  num <- gsub("[:]", "", str_sub(time, 12, 19), perl=TRUE)
  hr <- ifelse(str_sub(num, 1, 2) == "00", 24, as.numeric(str_sub(num, 1, 2)))
  min <- as.numeric(str_sub(num, 3, 4))
  sec <- as.numeric(str_sub(num, 5, 6))
  newTime <- 3600*hr + 60 * min + sec
  return(newTime)
}

# Add Time_Difference between the snap and the handoff
NFL_DATA_TRAIN_Modified$TimeDifference <-
  timeDifference(NFL_DATA_TRAIN_Modified$TimeHandoff) - timeDifference(NFL_DATA_TRAIN_Modified$TimeSnap)

# Add the Difference in Score by home score - visitor score
# Difference in Score (Pair with which team is winning (HomeScore-AwayScore))
NFL_DATA_TRAIN_Modified$HomeScoreAdvantage <-
  NFL_DATA_TRAIN_Modified$HomeScoreBeforePlay - NFL_DATA_TRAIN_Modified$VisitorScoreBeforePlay

# Add the age of the running player

# Change the birth dates to strings
NFL_DATA_TRAIN_Modified$PlayerBirthDate = as.character(NFL_DATA_TRAIN_Modified$PlayerBirthDate)
# Grab the Year for each of the running player
Birth_Year = str_sub(NFL_DATA_TRAIN_Modified$PlayerBirthDate, 7, 11)
# Grab Month of each running player
Birth_Month = str_sub(NFL_DATA_TRAIN_Modified$PlayerBirthDate, 1, 2)
# If Born in July (07) Have lived 5/12 of a year. ie (12 - (Birth_Month)) / 12
How_Much_Of_Year_Lived = (12 - as.numeric(Birth_Month)) / 12
Years_Lived = NFL_DATA_TRAIN_Modified$Season - as.numeric(Birth_Year)
```

```

Total_Years_Lived = Years_Lived + How_Much_Of_Year_Lived
NFL_DATA_TRAIN_Modified$PlayerAge = Total_Years_Lived

# Change HEIGHT to inches and continuous
Feet = as.numeric(str_sub(NFL_DATA_TRAIN_Modified$PlayerHeight, 1, 1))
Inches = as.numeric(str_sub(NFL_DATA_TRAIN_Modified$PlayerHeight, 3, 4))
Heights = (Feet * 12) + Inches
NFL_DATA_TRAIN_Modified$PlayerHeight = Heights

# Changes GAMECLOCK to Seconds.

NFL_DATA_TRAIN_Modified$GameClock = as.numeric(NFL_DATA_TRAIN_Modified$GameClock)

# FACTORING VARIABLES INTO CATEGORICAL

# Factor OFFENSE FORMATION
NFL_DATA_TRAIN_Modified$OffenseFormation = factor(NFL_DATA_TRAIN_Modified$OffenseFormation)

# DEFENDERS IN BOX (Need Categorical and Ordinal)
NFL_DATA_TRAIN_Modified$DefendersInTheBox = factor(NFL_DATA_TRAIN_Modified$DefendersInTheBox)

# REFACTOR DEFENDER IN BOX TO INCLUDE ORDINALITY
# Leaving them unordered was for graphs above, to look good
NFL_DATA_TRAIN_Modified$DefendersInTheBox = factor(NFL_DATA_TRAIN_Modified$DefendersInTheBox, order = T)

```

SELECTION OF COVARIATES FOR ANALYSIS

```

# Drop columns that are collinear, or we think are not critical to our model
NFL_DATA_TRAIN_Filtered = select(NFL_DATA_TRAIN_Modified,
                                -GameId, -PlayId, -Team, -S, -A, -Dis,
                                -Orientation, -Dir, -DisplayName, -JerseyNumber,
                                -YardLine, -FieldPosition, -HomeScoreBeforePlay,
                                -VisitorScoreBeforePlay, -NflId, -TimeHandoff,
                                -TimeSnap, -PlayerBirthDate, -PlayerCollegeName,
                                -Location, -WindSpeed, -WindDirection, -StadiumType,
                                -Turf, -GameWeather, -NflIdRusher, -Stadium)

# Turf and stadium type all captured in stadium
# View(NFL_DATA_TRAIN_Filtered) # drop game weather, captured in Week, and Stadium. Also too many missing

```

NOW THAT HAVE SELECTED COVARIATES. MORE DATA CLEANING, FACTORING, ETC...

```

# Need to count how many NA / Empty cells there are for each column
# summary(NFL_DATA_TRAIN_Filtered) # changed empty to NA when reading in file
# GameWeather, Temperature, Humidity all have missing or NA data
# sum(is.na(NFL_DATA_TRAIN_Filtered$GameWeather))
# sum(NFL_DATA_TRAIN_Filtered$GameWeather == "")

```

```

# Factor the DOWNS, Ordinally
NFL_DATA_TRAIN_Filtered$Down = factor(as.numeric(NFL_DATA_TRAIN_Filtered$Down), order = TRUE, levels = c(1, 2, 3, 4, 5))

# Player WEIGHT
NFL_DATA_TRAIN_Filtered$PlayerWeight = as.numeric(NFL_DATA_TRAIN_Filtered$PlayerWeight)

# Factor POSITION
NFL_DATA_TRAIN_Filtered$Position = factor(NFL_DATA_TRAIN_Filtered$Position)

# factor POSITION
NFL_DATA_TRAIN_Filtered$Position = factor(NFL_DATA_TRAIN_Filtered$Position)

# factor SEASON
NFL_DATA_TRAIN_Filtered$Season = factor(NFL_DATA_TRAIN_Filtered$Season)

# DATA CLEANING (REMOVING NA'S, and observations that happen less than 3 times)
# This was causing issues where one fold has a factor but another fold does not

# Need to delete a row within a column if there is just 1 special case. (Minimum 3 observations)

# DefensePersonnel (reduces observations by 11)
NFL_DATA_TRAIN_Filtered = NFL_DATA_TRAIN_Filtered[unsplit(table(NFL_DATA_TRAIN_Filtered$DefensePersonnel), 3), ]

# Same for OffensePersonnel (reduces by 18 observations)
NFL_DATA_TRAIN_Filtered = NFL_DATA_TRAIN_Filtered[unsplit(table(NFL_DATA_TRAIN_Filtered$OffensePersonnel), 3), ]

# Same for Position
NFL_DATA_TRAIN_Filtered = NFL_DATA_TRAIN_Filtered[unsplit(table(NFL_DATA_TRAIN_Filtered$Position), NFL_DATA_TRAIN_Filtered$Season), ]

# Same for Defenders In the Box
NFL_DATA_TRAIN_Filtered = NFL_DATA_TRAIN_Filtered[unsplit(table(NFL_DATA_TRAIN_Filtered$DefendersInTheBox), 3), ]

# Same for Offense Formation
NFL_DATA_TRAIN_Filtered = NFL_DATA_TRAIN_Filtered[unsplit(table(NFL_DATA_TRAIN_Filtered$OffenseFormation), 3), ]

# Need to remove NA Rows: Still 16,758 observations out of ~18,000
NFL_DATA_TRAIN_Filtered_Final <- na.omit(NFL_DATA_TRAIN_Filtered)
# View(NFL_DATA_TRAIN_Filtered_Final)

```

BEGINNING OF ANALYSIS

Regression with all Covariates

```

NFL_Train_Total_Model = lm(Yards ~ ., data=NFL_DATA_TRAIN_Filtered_Final)
NFL_Train_Total_Model.cv = cvFit(NFL_Train_Total_Model, data=NFL_DATA_TRAIN_Filtered_Final, y=NFL_DATA_TRAIN_Filtered_Final$Yards)
NFL_Train_Total_Model.cv # RMSE= 6.388 # May have collinearity

```

```

## 10-fold CV results:
##      CV
## 6.3971

```

NFL_Train_Total_Model\$coefficients

```
##          (Intercept)
##          2.52108549
##              X
##          -0.00119220
##              Y
##          0.01045713
##          Season2018
##          0.31226786
##          Quarter
##          -0.02536574
##          GameClock
##          0.00056159
## PossessionTeamATL
##          0.96758290
## PossessionTeamBLT
##          1.48581587
## PossessionTeamBUF
##          0.31194614
## PossessionTeamCAR
##          1.01634540
## PossessionTeamCHI
##          0.86344590
## PossessionTeamCIN
##          -0.94385118
## PossessionTeamCLV
##          0.73746082
## PossessionTeamDAL
##          0.90799918
## PossessionTeamDEN
##          0.58150661
## PossessionTeamDET
##          0.53371977
## PossessionTeamGB
##          1.00583198
## PossessionTeamHST
##          1.00245926
## PossessionTeamIND
##          0.90109676
## PossessionTeamJAX
##          0.40757723
## PossessionTeamKC
##          0.78847864
## PossessionTeamLA
##          0.76652796
## PossessionTeamLAC
##          0.95209005
## PossessionTeamMIA
##          0.05518131
## PossessionTeamMIN
##          -0.05919540
## PossessionTeamNE
##          0.53816994
```

```

##          PossessionTeamNO
##          1.69049649
##          PossessionTeamNYG
##          0.81186538
##          PossessionTeamNYJ
##          0.12309674
##          PossessionTeamOAK
##          0.63107048
##          PossessionTeamPHI
##          0.92693806
##          PossessionTeamPIT
##          1.01782074
##          PossessionTeamSEA
##          0.21956451
##          PossessionTeamSF
##          0.52377034
##          PossessionTeamTB
##          0.04561318
##          PossessionTeamTEN
##          0.94384586
##          PossessionTeamWAS
##          -0.05971857
##          Down.L
##          -0.63313387
##          Down.Q
##          -0.62457582
##          Down.C
##          -0.36004241
##          Distance
##          0.08398595
##          OffenseFormationI_FORM
##          -0.50450769
##          OffenseFormationJUMBO
##          -0.32362647
##          OffenseFormationPISTOL
##          -0.49613609
##          OffenseFormationSHOTGUN
##          -0.68759777
##          OffenseFormationSINGLEBACK
##          -0.56554251
##          OffenseFormationWILDCAT
##          -0.84658640
##          OffensePersonnel0 RB, 2 TE, 3 WR
##          -1.30532263
##          OffensePersonnel1 RB, 0 TE, 4 WR
##          3.43035544
##          OffensePersonnel1 RB, 1 TE, 2 WR,1 DB
##          7.76740241
##          OffensePersonnel1 RB, 1 TE, 2 WR,1 DL
##          1.98621204
##          OffensePersonnel1 RB, 1 TE, 3 WR
##          2.84005797
##          OffensePersonnel1 RB, 2 TE, 1 WR,1 DL
##          3.15175661

```

```

##      OffensePersonnel1 RB, 2 TE, 1 WR,1 LB
##      6.18902002
##      OffensePersonnel1 RB, 2 TE, 2 WR
##      2.67403284
##      OffensePersonnel1 RB, 3 TE, 0 WR,1 DL
##      2.28775612
##      OffensePersonnel1 RB, 3 TE, 0 WR,1 LB
##      4.29108556
##      OffensePersonnel1 RB, 3 TE, 1 WR
##      2.62599267
##      OffensePersonnel1 RB, 4 TE, 0 WR
##      2.11627955
##      OffensePersonnel2 QB, 1 RB, 0 TE, 3 WR
##      2.80048909
##      OffensePersonnel2 QB, 1 RB, 1 TE, 2 WR
##      2.66194505
##      OffensePersonnel2 QB, 1 RB, 2 TE, 1 WR
##      0.89022345
##      OffensePersonnel2 QB, 2 RB, 1 TE, 1 WR
##      6.25593404
##      OffensePersonnel2 RB, 0 TE, 3 WR
##      3.45922831
##      OffensePersonnel2 RB, 1 TE, 2 WR
##      2.73182300
##      OffensePersonnel2 RB, 2 TE, 1 WR
##      3.27488832
##      OffensePersonnel2 RB, 3 TE, 0 WR
##      1.68366195
##      OffensePersonnel3 RB, 0 TE, 2 WR
##      0.37120149
##      OffensePersonnel3 RB, 1 TE, 1 WR
##      2.41667123
##      OffensePersonnel3 RB, 2 TE, 0 WR
##      2.72744073
##      OffensePersonnel6 OL, 1 RB, 0 TE, 3 WR
##      3.08654545
##      OffensePersonnel6 OL, 1 RB, 1 TE, 1 WR,1 DL
##      1.81618010
##      OffensePersonnel6 OL, 1 RB, 1 TE, 2 WR
##      2.08163968
##      OffensePersonnel6 OL, 1 RB, 2 TE, 0 WR,1 DL
##      2.70307245
##      OffensePersonnel6 OL, 1 RB, 2 TE, 0 WR,1 LB
##      3.88172678
##      OffensePersonnel6 OL, 1 RB, 2 TE, 1 WR
##      2.95695541
##      OffensePersonnel6 OL, 1 RB, 3 TE, 0 WR
##      3.14659273
##      OffensePersonnel6 OL, 2 RB, 0 TE, 2 WR
##      3.96960985
##      OffensePersonnel6 OL, 2 RB, 1 TE, 0 WR,1 DL
##      2.17678393
##      OffensePersonnel6 OL, 2 RB, 1 TE, 1 WR
##      3.36070656

```

```

##      OffensePersonnel6 OL, 2 RB, 2 TE, 0 WR
##      3.32902917
##      OffensePersonnel7 OL, 1 RB, 0 TE, 2 WR
##      2.73854889
##      OffensePersonnel7 OL, 1 RB, 2 TE, 0 WR
##      2.67756499
##      OffensePersonnel7 OL, 2 RB, 0 TE, 1 WR
##      3.33216584
##      DefendersInTheBox.L
##      -7.32478206
##      DefendersInTheBox.Q
##      2.85631790
##      DefendersInTheBox.C
##      -1.61679386
##      DefendersInTheBox^4
##      0.93696063
##      DefendersInTheBox^5
##      -0.19978324
##      DefendersInTheBox^6
##      0.40815457
##      DefendersInTheBox^7
##      -0.11597370
##      DefendersInTheBox^8
##      0.25564617
##      DefensePersonnel0 DL, 5 LB, 6 DB
##      0.97626151
##      DefensePersonnel1 DL, 3 LB, 7 DB
##      -0.05725731
##      DefensePersonnel1 DL, 4 LB, 6 DB
##      0.79085292
##      DefensePersonnel1 DL, 5 LB, 5 DB
##      1.32987727
##      DefensePersonnel2 DL, 2 LB, 7 DB
##      3.22851736
##      DefensePersonnel2 DL, 3 LB, 6 DB
##      2.22597056
##      DefensePersonnel2 DL, 4 LB, 5 DB
##      1.72382273
##      DefensePersonnel2 DL, 5 LB, 4 DB
##      1.25490849
##      DefensePersonnel3 DL, 1 LB, 7 DB
##      1.01058640
##      DefensePersonnel3 DL, 2 LB, 6 DB
##      2.00367702
##      DefensePersonnel3 DL, 3 LB, 5 DB
##      1.68305586
##      DefensePersonnel3 DL, 4 LB, 4 DB
##      1.87769291
##      DefensePersonnel3 DL, 5 LB, 3 DB
##      2.49005849
##      DefensePersonnel4 DL, 0 LB, 7 DB
##      -1.05487035
##      DefensePersonnel4 DL, 1 LB, 6 DB
##      2.47192100

```



```

##      DefensePersonnel4 DL, 2 LB, 5 DB
##      1.74946935
##      DefensePersonnel4 DL, 3 LB, 4 DB
##      1.58488048
##      DefensePersonnel4 DL, 4 LB, 3 DB
##      0.50694608
##      DefensePersonnel4 DL, 5 LB, 2 DB
##      0.98642474
##      DefensePersonnel5 DL, 1 LB, 5 DB
##      0.84721849
##      DefensePersonnel5 DL, 2 LB, 4 DB
##      2.35149482
##      DefensePersonnel5 DL, 3 LB, 2 DB, 1 OL
##      0.19962839
##      DefensePersonnel5 DL, 3 LB, 3 DB
##      0.47388302
##      DefensePersonnel5 DL, 4 LB, 1 DB, 1 OL
##      -0.18884772
##      DefensePersonnel5 DL, 4 LB, 2 DB
##      0.68686377
##      DefensePersonnel5 DL, 5 LB, 1 DB
##      -0.24882664
##      DefensePersonnel6 DL, 2 LB, 3 DB
##      -0.13150646
##      DefensePersonnel6 DL, 3 LB, 2 DB
##      0.18396843
##      DefensePersonnel6 DL, 4 LB, 1 DB
##      -0.03062638
##      PlayDirectionright
##      -0.03651157
##      PlayerHeight
##      -0.02456371
##      PlayerWeight
##      -0.00188710
##      PositionHB
##      1.76069692
##      PositionQB
##      -1.13566553
##      PositionRB
##      0.15926141
##      PositionTE
##      1.69615801
##      PositionWR
##      1.93246856
##      HomeTeamAbbrATL
##      0.08198871
##      HomeTeamAbbrBAL
##      -0.31650457
##      HomeTeamAbbrBUF
##      -0.07351799
##      HomeTeamAbbrCAR
##      0.15384437
##      HomeTeamAbbrCHI
##      -0.88531074

```

##	HomeTeamAbbrCIN
##	0.26737373
##	HomeTeamAbbrCLE
##	-0.28456253
##	HomeTeamAbbrDAL
##	0.05802567
##	HomeTeamAbbrDEN
##	-0.27034530
##	HomeTeamAbbrDET
##	-0.51724057
##	HomeTeamAbbrGB
##	-0.20405597
##	HomeTeamAbbrHOU
##	-0.28264632
##	HomeTeamAbbrIND
##	-0.34272825
##	HomeTeamAbbrJAX
##	-0.12364225
##	HomeTeamAbbrKC
##	-0.29151672
##	HomeTeamAbbrLA
##	0.46937017
##	HomeTeamAbbrLAC
##	0.01047147
##	HomeTeamAbbrMIA
##	0.27611058
##	HomeTeamAbbrMIN
##	-0.50006295
##	HomeTeamAbbrNE
##	0.42103917
##	HomeTeamAbbrNO
##	-0.44120522
##	HomeTeamAbbrNYG
##	0.06701709
##	HomeTeamAbbrNYJ
##	0.43354830
##	HomeTeamAbbrOAK
##	0.01076283
##	HomeTeamAbbrPHI
##	-0.15252694
##	HomeTeamAbbrPIT
##	-0.24285636
##	HomeTeamAbbrSEA
##	0.19984037
##	HomeTeamAbbrSF
##	-0.23649103
##	HomeTeamAbbrTB
##	0.05777012
##	HomeTeamAbbrTEN
##	0.02166246
##	HomeTeamAbbrWAS
##	0.11388127
##	VisitorTeamAbbrATL
##	0.03318052

##	VisitorTeamAbbrBAL
##	-0.86663662
##	VisitorTeamAbbrBUF
##	0.00772628
##	VisitorTeamAbbrCAR
##	0.14640238
##	VisitorTeamAbbrCHI
##	-0.16795927
##	VisitorTeamAbbrCIN
##	-0.08809032
##	VisitorTeamAbbrCLE
##	0.17526987
##	VisitorTeamAbbrDAL
##	-0.16115269
##	VisitorTeamAbbrDEN
##	0.45728488
##	VisitorTeamAbbrDET
##	-0.01267271
##	VisitorTeamAbbrGB
##	-0.03680102
##	VisitorTeamAbbrHOU
##	-1.09977799
##	VisitorTeamAbbrIND
##	-0.23560234
##	VisitorTeamAbbrJAX
##	0.49311369
##	VisitorTeamAbbrKC
##	0.64711081
##	VisitorTeamAbbrLA
##	0.76207938
##	VisitorTeamAbbrLAC
##	-0.00839654
##	VisitorTeamAbbrMIA
##	0.43375656
##	VisitorTeamAbbrMIN
##	0.18078730
##	VisitorTeamAbbrNE
##	0.54313845
##	VisitorTeamAbbrNO
##	-0.29318421
##	VisitorTeamAbbrNYG
##	0.06948105
##	VisitorTeamAbbrNYJ
##	0.09937728
##	VisitorTeamAbbrOAK
##	0.21839019
##	VisitorTeamAbbrPHI
##	0.04239268
##	VisitorTeamAbbrPIT
##	-0.21032051
##	VisitorTeamAbbrSEA
##	0.06408241
##	VisitorTeamAbbrSF
##	0.40801456

```
##           VisitorTeamAbbrTB
##           0.31857477
##           VisitorTeamAbbrTEN
##           -0.12675904
##           VisitorTeamAbbrWAS
##           0.54252203
##           Week
##           -0.01535966
##           Temperature
##           -0.01052696
##           Humidity
##           -0.00431358
##           TimeDifference
##           0.09306539
##           HomeScoreAdvantage
##           -0.00600195
##           PlayerAge
##           -0.02960855
```

```
#NFL_Train_Total_Model1$coef)
#summary(NFL_Train_Total_Model1)$coef[summary(NFL_Train_Total_Model1)$coef[,4] <= .05, 1]
```

What if we always predicted the mean of yards? (Just Intercept Term)

```
NFL_Train_Total_Model1 = lm(Yards ~ 1, data=NFL_DATA_TRAIN_Filtered_Final)
NFL_Train_Total_Model1.cv = cvFit(NFL_Train_Total_Model1, data=NFL_DATA_TRAIN_Filtered_Final, y=NFL_DATA_TRAIN_Filtered_Final$Yards)
NFL_Train_Total_Model1.cv # RMSE= 6.4191
```

```
## 10-fold CV results:
##      CV
## 6.4304
```

Forward Stepwise Regression

```
min_model = NFL_Train_Total_Model1
max_model = NFL_Train_Total_Model
stepwise_model = stepAIC(min_model, direction='forward', scope=max_model)
```

```
## Start: AIC=62211
## Yards ~ 1
```

```
summary(stepwise_model)
```

```
##
## Call:
## lm(formula = Yards ~ 1, data = NFL_DATA_TRAIN_Filtered_Final)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.2    -3.2    -1.2     1.8    94.8
```

```
##
## Coefficients:
##           Estimate Std. Error t value      Pr(>|t|)
## (Intercept)   4.1974     0.0497   84.4 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.43 on 16713 degrees of freedom
# Ultimately, this is saying that the extra info we gain is not worth the complexity
```

Backward Stepwise Regerssion

```
backward_step = step(max_model, direction='backward')
backward_step
```

```
summary(backward_step)
```

```
##
## Call:
## lm(formula = Yards ~ Season + GameClock + Distance + DefendersInTheBox +
##      PlayerHeight + Position + Temperature + HomeScoreAdvantage +
##      PlayerAge, data = NFL_DATA_TRAIN_Filtered_Final)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.54   -3.15   -1.18    1.32   94.57
##
## Coefficients:
##              Estimate Std. Error t value      Pr(>|t|)
## (Intercept)    8.110000   2.203312   3.68      0.00023 ***
## Season2018     0.306328   0.099802   3.07      0.00215 **
## GameClock      0.000526   0.000188   2.79      0.00521 **
## Distance       0.078724   0.013740   5.73 0.00000001024767 ***
## DefendersInTheBox.L -7.732176   1.070072  -7.23 0.000000000000052 ***
## DefendersInTheBox.Q  1.745432   1.036363   1.68      0.09216 .
## DefendersInTheBox.C -1.411945   0.896918  -1.57      0.11546
## DefendersInTheBox^4  0.706433   0.736744   0.96      0.33764
## DefendersInTheBox^5  0.279722   0.577483   0.48      0.62812
## DefendersInTheBox^6  0.556270   0.427169   1.30      0.19286
## DefendersInTheBox^7  0.043472   0.280617   0.15      0.87689
## DefendersInTheBox^8  0.311971   0.145645   2.14      0.03221 *
## PlayerHeight    -0.041499   0.027849  -1.49      0.13620
## PositionHB       0.102257   0.660893   0.15      0.87704
## PositionQB      -0.974017   1.042221  -0.93      0.35003
## PositionRB       0.147750   0.585328   0.25      0.80072
## PositionTE       1.694155   1.481742   1.14      0.25291
## PositionWR       1.746851   0.657730   2.66      0.00792 **
## Temperature     -0.007202   0.002829  -2.55      0.01092 *
## HomeScoreAdvantage -0.007435   0.004602  -1.62      0.10618
## PlayerAge       -0.050160   0.016400  -3.06      0.00223 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 6.37 on 16693 degrees of freedom
## Multiple R-squared: 0.0205, Adjusted R-squared: 0.0193
## F-statistic: 17.4 on 20 and 16693 DF, p-value: <0.0000000000000002
```

correction

```
pvals <- summary(backward_step)$coefficients[, 'Pr(>|t|)']
n_significant = 13
n_covariates = 24
```

```
cat("Bonferroni Corection: \n")
```

```
## Bonferroni Corection:
```

```
p.adjust(pvals, method = 'bonferroni', n = n_significant + n_covariates + 1)
```

```
##      (Intercept)      Season2018      GameClock
## 0.008862146043535 0.081649268785298 0.198035839299143
##      Distance DefendersInTheBox.L DefendersInTheBox.Q
## 0.000000389411565 0.000000000019742 1.000000000000000
## DefendersInTheBox.C DefendersInTheBox^4 DefendersInTheBox^5
## 1.000000000000000 1.000000000000000 1.000000000000000
## DefendersInTheBox^6 DefendersInTheBox^7 DefendersInTheBox^8
## 1.000000000000000 1.000000000000000 1.000000000000000
##      PlayerHeight      PositionHB      PositionQB
## 1.000000000000000 1.000000000000000 1.000000000000000
##      PositionRB      PositionTE      PositionWR
## 1.000000000000000 1.000000000000000 0.300872162062431
##      Temperature HomeScoreAdvantage      PlayerAge
## 0.414900546828430 1.000000000000000 0.084629037841733
```

```
cat("Benjamini-Hochberg: \n")
```

```
## Benjamini-Hochberg:
```

```
bh <- p.adjust(pvals, method = 'BH', n = n_significant + n_covariates + 1)
bh
```

```
##      (Intercept)      Season2018      GameClock
## 0.002954048681178 0.016925807568347 0.033005973216524
##      Distance DefendersInTheBox.L DefendersInTheBox.Q
## 0.000000194705783 0.000000000019742 0.350222460842332
## DefendersInTheBox.C DefendersInTheBox^4 DefendersInTheBox^5
## 0.365610766419858 0.782418248625900 1.000000000000000
## DefendersInTheBox^6 DefendersInTheBox^7 DefendersInTheBox^8
## 0.523471072141957 1.000000000000000 0.135989827319851
##      PlayerHeight      PositionHB      PositionQB
## 0.398117387374248 1.000000000000000 0.782418248625900
##      PositionRB      PositionTE      PositionWR
## 1.000000000000000 0.640700707715220 0.042981737437490
##      Temperature HomeScoreAdvantage      PlayerAge
## 0.051862568353554 0.365610766419858 0.016925807568347
```

```
which(bh < 0.05)
```

```
##          (Intercept)          Season2018          GameClock
##              1              2              3
##          Distance DefendersInTheBox.L          PositionWR
##              4              5              18
##          PlayerAge
##              21
```

```
backward_step.cv = cvFit(backward_step, data=NFL_DATA_TRAIN_Filtered_Final, y=NFL_DATA_TRAIN_Filtered_F
backward_step.cv # RMSE= 6.5386 # Best Model so far
```

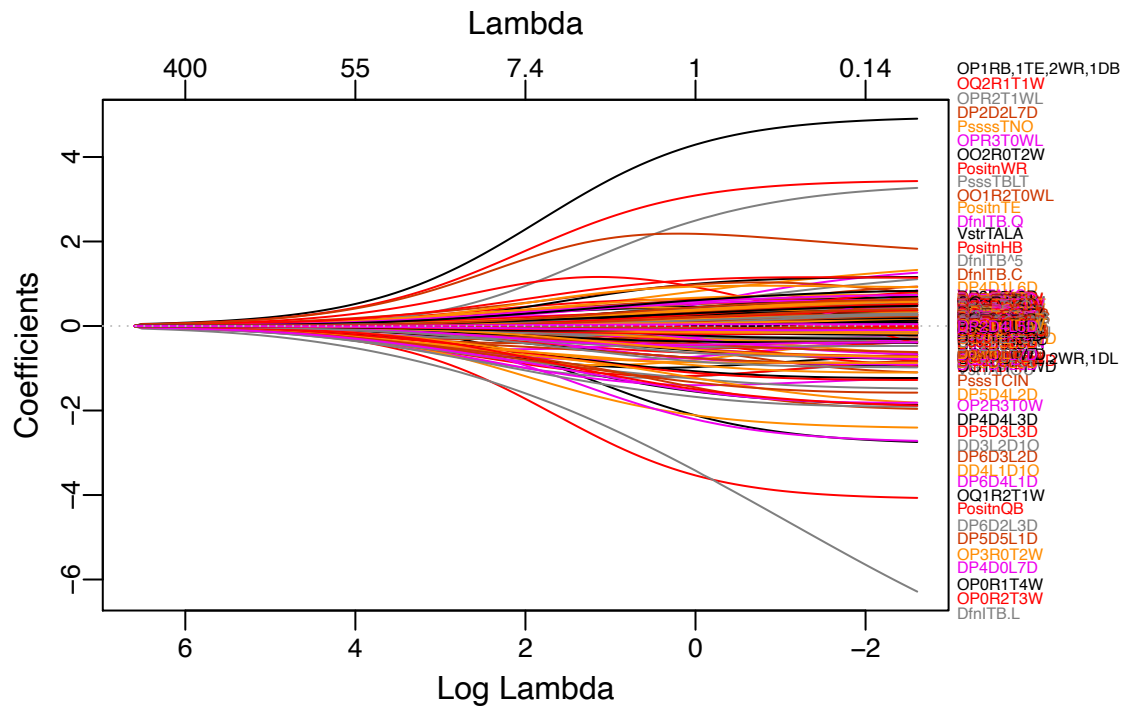
```
## 10-fold CV results:
##      CV
## 6.3744
```

Ridge Regression

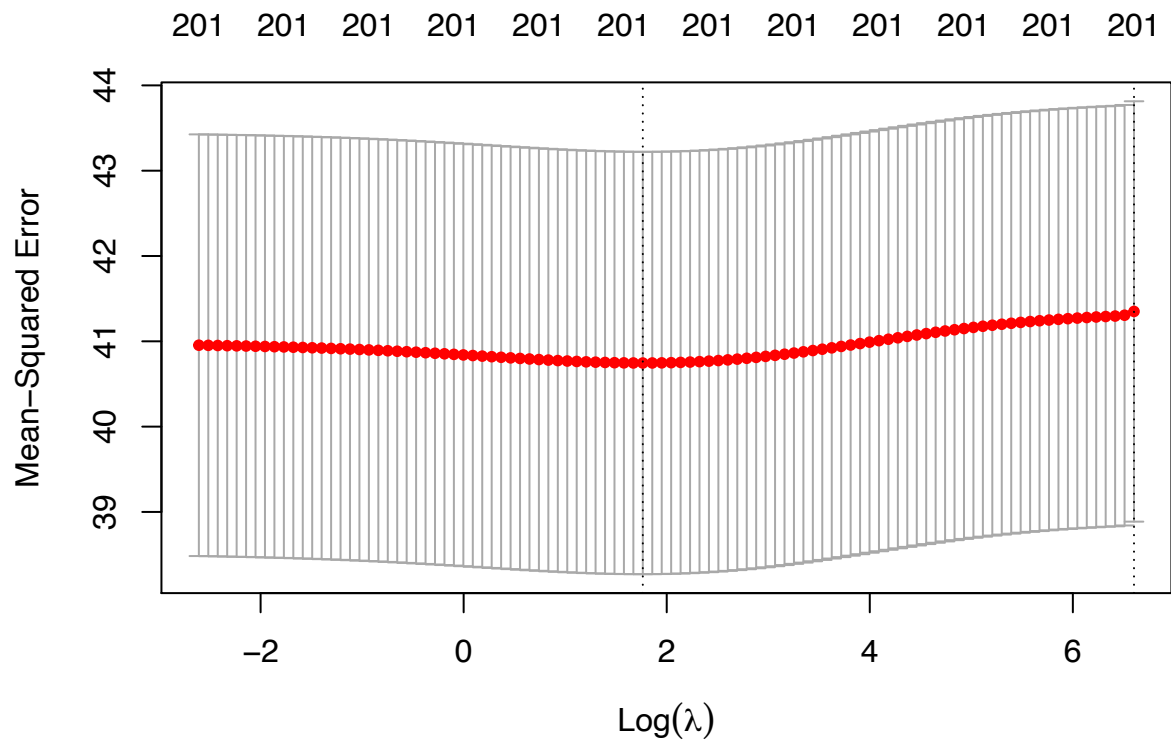
```
library(glmnet)
## NORMALIZE Continuous Covariates
Standardized_NFL_TRAIN = NFL_DATA_TRAIN_Filtered_Final
Standardized_NFL_TRAIN$X = scale(Standardized_NFL_TRAIN$X)
Standardized_NFL_TRAIN$Y = scale(Standardized_NFL_TRAIN$Y)
Standardized_NFL_TRAIN$GameClock = scale(Standardized_NFL_TRAIN$GameClock )
Standardized_NFL_TRAIN$Distance = scale(Standardized_NFL_TRAIN$Distance)
Standardized_NFL_TRAIN$PlayerHeight = scale(Standardized_NFL_TRAIN$PlayerHeight)
Standardized_NFL_TRAIN$PlayerWeight = scale(Standardized_NFL_TRAIN$PlayerWeight)
Standardized_NFL_TRAIN$Week = scale(Standardized_NFL_TRAIN$Week)
Standardized_NFL_TRAIN$Temperature = scale(Standardized_NFL_TRAIN$Temperature)
Standardized_NFL_TRAIN$Humidity = scale(Standardized_NFL_TRAIN$Humidity)
Standardized_NFL_TRAIN$TimeDifference = scale(Standardized_NFL_TRAIN$TimeDifference)
Standardized_NFL_TRAIN$HomeScoreAdvantage = scale(Standardized_NFL_TRAIN$HomeScoreAdvantage )
Standardized_NFL_TRAIN$PlayerAge = scale(Standardized_NFL_TRAIN$PlayerAge)

# Ridge Regression
# Ridge alpha = 0
x = model.matrix(Yards~. , Standardized_NFL_TRAIN)
y = Standardized_NFL_TRAIN$Yards
ridge_mod = glmnet(x, y, alpha = 0)
# install.packages("plotmo")
plot_glmnet(ridge_mod, label = TRUE)
```

```
## Warning in TeachingDemos::spread.labs(beta[iname, ncol(beta)], mindiff =
## 1.2 * : Maximum iterations reached
```



```
cvfit = cv.glmnet(x, y, alpha = 0)
plot(cvfit)
```



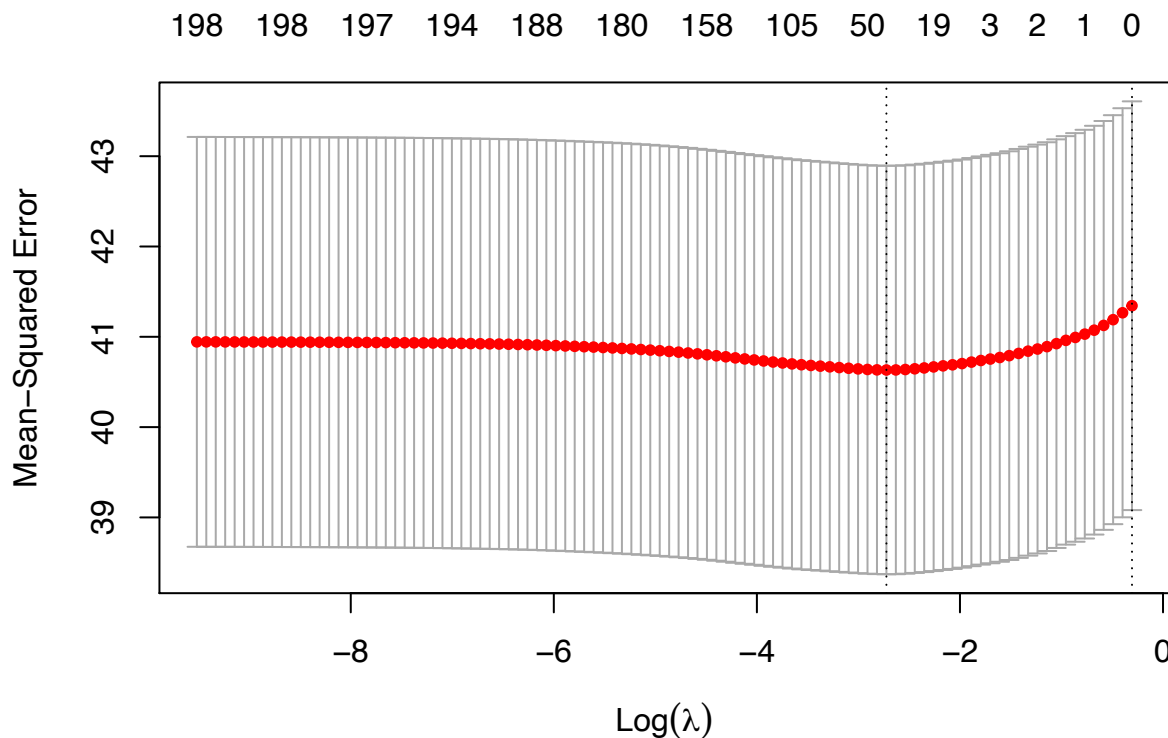
```
bestlam = cvfit$lambda.min # = 4.9384
```



```
y_predicted <- predict(cvfit, s = bestlam, newx = x) # same x, in sample prediction
ridge_RMSE = sqrt(mean((y_predicted - y)^2))
# ridge_RMSE # = 6.3333
#coef(ridge_mod)[,4.9384] # Best is again basically forcing all the betas to 0. Just predict mean
```

```
lasso_mod= glmnet(x, y, alpha = 1)
# coef(lasso_mod)[,50]
plot_glmnet(lasso_mod, label = TRUE)
```

16



```
bestlam_lasso = cvfit_lasso$lambda.min # = 0.046056
```

```
y_predicted_lasso <- predict(cvfit_lasso, s = bestlam_lasso, newx = x) # same x, in sample prediction
lasso_RMSE = sqrt(mean((y_predicted_lasso - y)^2))
lasso_RMSE # = 6.3349
```

```
## [1] 6.3578
```

```
out = glmnet(x, y, alpha = 1) # Fit ridge regression model on full dataset
#predict(out, type = "coefficients", s = bestlam_lasso)[1:80,] # Display coefficients using lambda chosen
```

```
#Applying same transformation to test set
```

```
# Function to take the difference in time from the dataframe
```

```
timeDifference <- function(time) {
  num <- gsub("[:]", "", str_sub(time, 12, 19), perl=TRUE)
  hr <- ifelse(str_sub(num, 1, 2) == "00", 24, as.numeric(str_sub(num, 1, 2)))
  min <- as.numeric(str_sub(num, 3, 4))
  sec <- as.numeric(str_sub(num, 5, 6))
  newTime <- 3600*hr + 60 * min + sec
  return(newTime)
}
```

```
# Add Time_Difference between the snap and the handoff
```

```
NFL_DATA_Holdout$TimeDifference <-
  timeDifference(NFL_DATA_Holdout$TimeHandoff) - timeDifference(NFL_DATA_Holdout$TimeSnap)
```

```
# Add the Difference in Score by home score - visitor score
```

```
# Difference in Score (Pair with which team is winning (HomeScore-AwayScore))
```

```
NFL_DATA_Holdout$HomeScoreAdvantage <-
```

```
  NFL_DATA_Holdout$HomeScoreBeforePlay - NFL_DATA_Holdout$VisitorScoreBeforePlay
```

```
# Add the age of the running player
```

```

# Change the birth dates to strings
NFL_DATA_Holdout$PlayerBirthDate = as.character(NFL_DATA_Holdout$PlayerBirthDate)
# Grab the Year for each of the running player
Birth_Year = str_sub(NFL_DATA_Holdout$PlayerBirthDate, 7, 11)
# Grab Month of each running player
Birth_Month = str_sub(NFL_DATA_Holdout$PlayerBirthDate, 1, 2)
# If Born in July (07) Have lived 5/12 of a year. ie (12 - (Birth_Month)) / 12
How_Much_Of_Year_Lived = (12 - as.numeric(Birth_Month)) / 12
Years_Lived = NFL_DATA_Holdout$Season - as.numeric(Birth_Year)
Total_Years_Lived = Years_Lived + How_Much_Of_Year_Lived
NFL_DATA_Holdout$PlayerAge = Total_Years_Lived

# Change HEIGHT to inches and continuous
Feet = as.numeric(str_sub(NFL_DATA_Holdout$PlayerHeight, 1, 1))
Inches = as.numeric(str_sub(NFL_DATA_Holdout$PlayerHeight, 3, 4))
Heights = (Feet * 12) + Inches
NFL_DATA_Holdout$PlayerHeight = Heights

# Changes GAMECLOCK to Seconds.

NFL_DATA_Holdout$GameClock = as.numeric(NFL_DATA_Holdout$GameClock)

# FACTORING VARIABLES INTO CATEGORICAL

# Factor OFFENSE FORMATION
NFL_DATA_Holdout$OffenseFormation = factor(NFL_DATA_Holdout$OffenseFormation)

# DEFENDERS IN BOX (Need Categorical and Ordinal)
NFL_DATA_Holdout$DefendersInTheBox = factor(NFL_DATA_Holdout$DefendersInTheBox)

# REFACTOR DEFENDER IN BOX TO INCLUDE ORDINALITY
# Leaving them unordered was for graphs above, to look good
NFL_DATA_Holdout$DefendersInTheBox = factor(NFL_DATA_Holdout$DefendersInTheBox, order = TRUE, levels = c(
# Need to count how many NA / Empty cells there are for each column
# summary(NFL_DATA_TRAIN_Filtered) # changed empty to NA when reading in file
# GameWeather, Temperature, Humidity all have missing or NA data
# sum(is.na(NFL_DATA_TRAIN_Filtered$GameWeather))
# sum(NFL_DATA_TRAIN_Filtered$GameWeather == ""))

# Factor the DOWNS, Ordinally
NFL_DATA_Holdout$Down = factor(as.numeric(NFL_DATA_Holdout$Down), order = TRUE, levels = c(1,2,3,4,5))

# Player WEIGHT
NFL_DATA_Holdout$PlayerWeight = as.numeric(NFL_DATA_Holdout$PlayerWeight)

# Factor POoldout
NFL_DATA_Holdout$Position = factor(NFL_DATA_Holdout$Positoldout# factor POSITION
NFL_DATA_Holdout$Position = factor(NFL_DATA_Holdout$Position)

# factor SEASON
NFL_DATA_Holdout$Season = factor(NFL_DATA_Holdout$Season)

```

```

# DATA CLEANING (REMOVING NA'S, and observations that happen less than 3 times)
# This was causing issues where one fold has a factor but another fold does not

# Need to delete a row within a column if there is just 1 special case. (Minimun 3 observations)

# DefensePersonnel (reduces observations by 11)
NFL_DATA_Holdout = NFL_DATA_Holdout[unsplit(table(NFL_DATA_Holdout$DefensePersonnel), NFL_DATA_Holdout$DefensePersonnel)]

# Same for OffensePersonnel (reduces by 18 observations)
NFL_DATA_Holdout = NFL_DATA_Holdout[unsplit(table(NFL_DATA_Holdout$OffensePersonnel), NFL_DATA_Holdout$OffensePersonnel)]

# Same for Pooldout
NFL_DATA_Holdout = NFL_DATA_Holdout[unsplit(table(NFL_DATA_Holdout$Position), NFL_DATA_Holdout$Position)]

# Same for Defenders In the Box
NFL_DATA_Holdout = NFL_DATA_Holdout[unsplit(table(NFL_DATA_Holdout$DefendersInTheBox), NFL_DATA_Holdout$DefendersInTheBox)]

# Same for Offense Formation
NFL_DATA_Holdout = NFL_DATA_Holdout[unsplit(table(NFL_DATA_Holdout$OffenseFormation), NFL_DATA_Holdout$OffenseFormation)]

# Drop columns that are collinear, or we think are not critical to our model
NFL_DATA_Holdout = select(NFL_DATA_Holdout,
                           -GameId, -PlayId, -Team, -S, -A, -Dis,
                           -Orientation, -Dir, -DisplayName, -JerseyNumber,
                           -YardLine, -FieldPosition, -HomeScoreBeforePlay,
                           -VisitorScoreBeforePlay, -NflId, -TimeHandoff,
                           -TimeSnap, -PlayerBirthDate, -PlayerCollegeName,
                           -Location, -WindSpeed, -WindDirection, -StadiumType,
                           -Turf, -GameWeather, -NflIdRusher, -Stadium)

# Turf and stadium type all captured in stadium
# drop game weather, captured in Week, and Stadium. Also too many missing values

NFL_DATA_HOLDOUT_Final <- na.omit(NFL_DATA_Holdout)

View(NFL_DATA_HOLDOUT_Final)

## Warning in system2("/usr/bin/otool", c("-L", shQuote(DSO)), stdout = TRUE):
## running command ''/usr/bin/otool' -L '/Library/Frameworks/R.framework/
## Resources/modules/R_de.so'' had status 1

```

Using model with all covariates to predict on test set

```

NFL_TEST_Model.cv = cvFit(NFL_Train_Total_Model, data=NFL_DATA_HOLDOUT_Final, y=NFL_DATA_HOLDOUT_Final$Y)

## Warning in predict.lm(...): prediction from a rank-deficient fit may be
## misleading
NFL_TEST_Model.cv # RMSE= 6.4191

```

```
## 10-fold CV results:
##      CV
## 6.7821
```

```
#pred = predict(NFL_Train_Total_Model, NFL_DATA_HOLDOUT_Final)
#rmse.test = sqrt(mean((pred - NFL_DATA_HOLDOUT_Final$Yards)^2))
#rmse.test
```

Using model with only intercept term

```
NFL_TEST_Model.cv = cvFit(NFL_Train_Total_Model1, data=NFL_DATA_HOLDOUT_Final, y=NFL_DATA_HOLDOUT_Final$Yards, K=10)
NFL_TEST_Model.cv # RMSE= 6.4191
```

```
## 10-fold CV results:
##      CV
## 6.6685
```

```
#Using 'best' model (backward stepwise regression)
```

```
NFL_TEST_Model.cv = cvFit(backward_step, data=NFL_DATA_HOLDOUT_Final, y=NFL_DATA_HOLDOUT_Final$Yards, K=10)
NFL_TEST_Model.cv # RMSE= 6.4191
```

```
## 10-fold CV results:
##      CV
## 6.6491
```

```
summary(backward_step)
```

```
##
## Call:
## lm(formula = Yards ~ Season + GameClock + Distance + DefendersInTheBox +
##      PlayerHeight + Position + Temperature + HomeScoreAdvantage +
##      PlayerAge, data = NFL_DATA_TRAIN_Filtered_Final)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.54   -3.15   -1.18    1.32   94.57
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    8.110000   2.203312   3.68    0.00023 ***
## Season2018     0.306328   0.099802   3.07    0.00215 **
## GameClock      0.000526   0.000188   2.79    0.00521 **
## Distance       0.078724   0.013740   5.73 0.00000001024767 ***
## DefendersInTheBox.L -7.732176  1.070072  -7.23 0.000000000000052 ***
## DefendersInTheBox.Q  1.745432  1.036363   1.68    0.09216 .
## DefendersInTheBox.C -1.411945  0.896918  -1.57    0.11546
## DefendersInTheBox^4  0.706433  0.736744   0.96    0.33764
## DefendersInTheBox^5  0.279722  0.577483   0.48    0.62812
## DefendersInTheBox^6  0.556270  0.427169   1.30    0.19286
## DefendersInTheBox^7  0.043472  0.280617   0.15    0.87689
## DefendersInTheBox^8  0.311971  0.145645   2.14    0.03221 *
## PlayerHeight    -0.041499  0.027849  -1.49    0.13620
## PositionHB       0.102257  0.660893   0.15    0.87704
```

```
## PositionQB      -0.974017   1.042221   -0.93       0.35003
## PositionRB      0.147750   0.585328    0.25       0.80072
## PositionTE      1.694155   1.481742    1.14       0.25291
## PositionWR      1.746851   0.657730    2.66       0.00792 **
## Temperature     -0.007202   0.002829   -2.55       0.01092 *
## HomeScoreAdvantage -0.007435   0.004602   -1.62       0.10618
## PlayerAge       -0.050160   0.016400   -3.06       0.00223 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.37 on 16693 degrees of freedom
## Multiple R-squared:  0.0205, Adjusted R-squared:  0.0193
## F-statistic: 17.4 on 20 and 16693 DF,  p-value: <0.0000000000000002
```

Fitting Backward Stepwise Regression to test set

```
#NFL_TEST_min_Model = lm(Yards ~ 1, data=NFL_DATA_HOLDOUT_Final)
NFL_TEST_max_Model = lm(Yards ~ ., data=NFL_DATA_HOLDOUT_Final)
test_backward_step = step(NFL_TEST_max_Model, direction='backward')

## Start:  AIC=16034
## Yards ~ X + Y + Season + Quarter + GameClock + PossessionTeam +
##      Down + Distance + OffenseFormation + OffensePersonnel + DefendersInTheBox +
##      DefensePersonnel + PlayDirection + PlayerHeight + PlayerWeight +
##      Position + HomeTeamAbbr + VisitorTeamAbbr + Week + Temperature +
##      Humidity + TimeDifference + HomeScoreAdvantage + PlayerAge
##
##           Df Sum of Sq    RSS   AIC
## - HomeTeamAbbr      31      1216 177364 16001
## - DefensePersonnel   22       571 176718 16003
## - VisitorTeamAbbr    31      1483 177630 16007
## - OffensePersonnel   25      1181 177328 16012
## - PossessionTeam     31      2162 178309 16023
## - OffenseFormation    6       131 176278 16025
## - Down                3       110 176257 16030
## - PlayerHeight        1          0 176147 16032
## - Y                   1          1 176148 16032
## - Season              1          1 176148 16032
## - Temperature         1         17 176164 16032
## - Week                1         19 176166 16032
## - PlayDirection       1         19 176166 16032
## - X                   1         33 176180 16033
## - PlayerWeight        1         44 176191 16033
## - GameClock           1         48 176195 16033
## - HomeScoreAdvantage  1         61 176208 16033
## - Distance            1         68 176215 16033
## - Humidity            1         70 176217 16033
## <none>                                176147 16034
## - TimeDifference      1        175 176322 16036
## - DefendersInTheBox   8        781 176928 16036
## - Quarter             1        241 176388 16038
## - PlayerAge           1        243 176390 16038
```

```

## - Position          6          695 176842 16038
##
## Step:  AIC=16001
## Yards ~ X + Y + Season + Quarter + GameClock + PossessionTeam +
##      Down + Distance + OffenseFormation + OffensePersonnel + DefendersInTheBox +
##      DefensePersonnel + PlayDirection + PlayerHeight + PlayerWeight +
##      Position + VisitorTeamAbbr + Week + Temperature + Humidity +
##      TimeDifference + HomeScoreAdvantage + PlayerAge
##
##
##      Df Sum of Sq    RSS    AIC
## - DefensePersonnel 22      765 178129 15975
## - VisitorTeamAbbr  31     1649 179012 15977
## - OffensePersonnel 25     1165 178529 15978
## - OffenseFormation  6      132 177495 15992
## - PossessionTeam   31     2281 179645 15992
## - Down              3      127 177491 15998
## - Y                  1         0 177364 15999
## - PlayerHeight       1         0 177364 15999
## - Temperature        1         1 177364 15999
## - Season              1         2 177366 15999
## - PlayDirection      1        16 177379 15999
## - Humidity            1        17 177381 15999
## - HomeScoreAdvantage 1        31 177395 15999
## - X                   1        33 177396 15999
## - GameClock           1        45 177408 16000
## - Week                1        46 177410 16000
## - PlayerWeight        1        49 177412 16000
## - Distance            1        84 177447 16001
## <none>                  177364 16001
## - TimeDifference      1       185 177548 16003
## - DefendersInTheBox  8       793 178157 16003
## - Quarter             1       201 177565 16003
## - PlayerAge           1       251 177615 16005
## - Position            6       706 178070 16005
##
## Step:  AIC=15975
## Yards ~ X + Y + Season + Quarter + GameClock + PossessionTeam +
##      Down + Distance + OffenseFormation + OffensePersonnel + DefendersInTheBox +
##      PlayDirection + PlayerHeight + PlayerWeight + Position +
##      VisitorTeamAbbr + Week + Temperature + Humidity + TimeDifference +
##      HomeScoreAdvantage + PlayerAge
##
##
##      Df Sum of Sq    RSS    AIC
## - VisitorTeamAbbr  31     1549 179678 15949
## - OffensePersonnel 25     1109 179238 15951
## - PossessionTeam   31     2211 180340 15964
## - OffenseFormation  6      131 178260 15966
## - Down              3      116 178245 15971
## - Y                  1         0 178129 15973
## - PlayerHeight       1         1 178129 15973
## - Season              1         3 178132 15973
## - Temperature        1         4 178133 15973
## - Humidity            1         8 178137 15973
## - PlayDirection      1        19 178148 15973

```

```

## - X                1          32 178161 15973
## - Week              1          36 178165 15974
## - HomeScoreAdvantage 1          39 178168 15974
## - GameClock         1          51 178180 15974
## - PlayerWeight      1          51 178180 15974
## <none>              178129 15975
## - Distance          1          86 178215 15975
## - TimeDifference     1         193 178322 15977
## - Quarter           1         203 178332 15977
## - PlayerAge          1         266 178395 15979
## - Position           6         745 178874 15980
## - DefendersInTheBox 8        1084 179213 15984
##
## Step:  AIC=15949
## Yards ~ X + Y + Season + Quarter + GameClock + PossessionTeam +
##   Down + Distance + OffenseFormation + OffensePersonnel + DefendersInTheBox +
##   PlayDirection + PlayerHeight + PlayerWeight + Position +
##   Week + Temperature + Humidity + TimeDifference + HomeScoreAdvantage +
##   PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - OffensePersonnel 25      1088 180766 15924
## - PossessionTeam   31      1635 181313 15925
## - OffenseFormation  6       135 179813 15940
## - Down              3       137 179815 15946
## - PlayerHeight      1         0 179678 15947
## - Y                  1         1 179679 15947
## - Temperature       1         2 179680 15947
## - Season             1         4 179682 15947
## - Humidity           1         7 179686 15947
## - PlayDirection     1        16 179695 15947
## - X                  1        29 179707 15948
## - Week               1        32 179710 15948
## - GameClock          1        55 179734 15948
## - PlayerWeight       1        58 179736 15948
## - HomeScoreAdvantage 1        72 179750 15949
## - Distance           1        85 179764 15949
## <none>              179678 15949
## - Quarter           1       195 179873 15952
## - TimeDifference     1       207 179885 15952
## - PlayerAge          1       271 179949 15953
## - Position           6       745 180423 15954
## - DefendersInTheBox 8      1084 180762 15958
##
## Step:  AIC=15924
## Yards ~ X + Y + Season + Quarter + GameClock + PossessionTeam +
##   Down + Distance + OffenseFormation + DefendersInTheBox +
##   PlayDirection + PlayerHeight + PlayerWeight + Position +
##   Week + Temperature + Humidity + TimeDifference + HomeScoreAdvantage +
##   PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - PossessionTeam   31      1935 182701 15907
## - OffenseFormation  6         39 180805 15913

```



```

## - Down          3      114 180880 15921
## - Y             1        0 180766 15922
## - PlayerHeight  1        1 180767 15922
## - Temperature   1        2 180768 15922
## - Humidity       1        4 180770 15922
## - Season         1        8 180774 15923
## - PlayDirection 1       15 180781 15923
## - Week           1       28 180794 15923
## - X              1       31 180797 15923
## - GameClock      1       45 180811 15923
## - PlayerWeight   1       51 180817 15923
## - HomeScoreAdvantage 1     57 180823 15924
## - Distance       1       85 180851 15924
## <none>           180766 15924
## - Position       6     558 181324 15925
## - Quarter        1     170 180936 15926
## - TimeDifference  1     209 180975 15927
## - PlayerAge      1     265 181031 15928
## - DefendersInTheBox 8    1204 181970 15936
##
## Step: AIC=15907
## Yards ~ X + Y + Season + Quarter + GameClock + Down + Distance +
##      OffenseFormation + DefendersInTheBox + PlayDirection + PlayerHeight +
##      PlayerWeight + Position + Week + Temperature + Humidity +
##      TimeDifference + HomeScoreAdvantage + PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - OffenseFormation    6      56 182757 15896
## - Down                 3      98 182799 15903
## - Position             6     362 183063 15903
## - Y                    1        0 182701 15905
## - PlayerHeight         1        0 182701 15905
## - Temperature          1     15 182716 15905
## - Week                  1     18 182719 15905
## - Season                1     25 182726 15906
## - PlayDirection        1     26 182727 15906
## - PlayerWeight          1     27 182728 15906
## - Humidity              1     32 182733 15906
## - GameClock             1     37 182738 15906
## - X                     1     39 182740 15906
## - HomeScoreAdvantage    1     63 182764 15906
## <none>                 182701 15907
## - Distance              1     93 182794 15907
## - TimeDifference         1    181 182882 15909
## - Quarter               1    185 182886 15909
## - PlayerAge             1    237 182938 15910
## - DefendersInTheBox     8   1138 183839 15917
##
## Step: AIC=15896
## Yards ~ X + Y + Season + Quarter + GameClock + Down + Distance +
##      DefendersInTheBox + PlayDirection + PlayerHeight + PlayerWeight +
##      Position + Week + Temperature + Humidity + TimeDifference +
##      HomeScoreAdvantage + PlayerAge
##

```

	Df	Sum of Sq	RSS	AIC
## - Position	6	355	183111	15892
## - Down	3	99	182856	15893
## - Y	1	0	182757	15894
## - PlayerHeight	1	1	182758	15894
## - Temperature	1	15	182771	15895
## - Week	1	19	182776	15895
## - PlayerWeight	1	22	182779	15895
## - Season	1	24	182780	15895
## - PlayDirection	1	25	182782	15895
## - Humidity	1	33	182790	15895
## - GameClock	1	36	182793	15895
## - X	1	37	182794	15895
## - HomeScoreAdvantage	1	65	182821	15896
## <none>			182757	15896
## - Distance	1	98	182855	15897
## - Quarter	1	180	182936	15898
## - TimeDifference	1	199	182956	15899
## - PlayerAge	1	232	182988	15900
## - DefendersInTheBox	8	1534	184291	15915

Step: AIC=15892

Yards ~ X + Y + Season + Quarter + GameClock + Down + Distance +
DefendersInTheBox + PlayDirection + PlayerHeight + PlayerWeight +
Week + Temperature + Humidity + TimeDifference + HomeScoreAdvantage +
PlayerAge

	Df	Sum of Sq	RSS	AIC
## - Down	3	108	183219	15889
## - Y	1	1	183113	15890
## - PlayerWeight	1	4	183116	15890
## - Temperature	1	13	183125	15891
## - PlayerHeight	1	20	183131	15891
## - PlayDirection	1	23	183134	15891
## - Season	1	24	183136	15891
## - Week	1	26	183137	15891
## - Humidity	1	31	183142	15891
## - GameClock	1	31	183142	15891
## - X	1	32	183143	15891
## - HomeScoreAdvantage	1	67	183179	15892
## <none>			183111	15892
## - Distance	1	100	183211	15893
## - Quarter	1	162	183273	15894
## - TimeDifference	1	209	183320	15895
## - PlayerAge	1	211	183323	15895
## - DefendersInTheBox	8	1532	184644	15911

Step: AIC=15889

Yards ~ X + Y + Season + Quarter + GameClock + Distance + DefendersInTheBox +
PlayDirection + PlayerHeight + PlayerWeight + Week + Temperature +
Humidity + TimeDifference + HomeScoreAdvantage + PlayerAge

	Df	Sum of Sq	RSS	AIC
## - Y	1	2	183221	15887

```

## - PlayerWeight      1      4 183224 15887
## - Temperature       1     16 183235 15887
## - PlayerHeight      1     21 183240 15887
## - Season            1     22 183241 15887
## - PlayDirection     1     23 183243 15887
## - Week              1     26 183245 15887
## - Humidity          1     30 183250 15888
## - X                 1     31 183251 15888
## - GameClock         1     32 183251 15888
## - HomeScoreAdvantage 1     65 183284 15888
## <none>              183219 15889
## - Distance          1     88 183308 15889
## - Quarter           1    165 183384 15891
## - PlayerAge         1    204 183423 15892
## - TimeDifference     1    214 183434 15892
## - DefendersInTheBox 8    1679 184898 15911
##
## Step: AIC=15887
## Yards ~ X + Season + Quarter + GameClock + Distance + DefendersInTheBox +
##      PlayDirection + PlayerHeight + PlayerWeight + Week + Temperature +
##      Humidity + TimeDifference + HomeScoreAdvantage + PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - PlayerWeight      1      4 183225 15885
## - Temperature       1     16 183237 15885
## - PlayerHeight      1     21 183242 15885
## - Season            1     21 183243 15885
## - PlayDirection     1     24 183245 15885
## - Week              1     26 183247 15885
## - Humidity          1     30 183251 15886
## - GameClock         1     31 183253 15886
## - X                 1     32 183253 15886
## - HomeScoreAdvantage 1     65 183286 15886
## <none>              183221 15887
## - Distance          1     88 183309 15887
## - Quarter           1    166 183387 15889
## - PlayerAge         1    204 183425 15890
## - TimeDifference     1    215 183436 15890
## - DefendersInTheBox 8    1680 184902 15909
##
## Step: AIC=15885
## Yards ~ X + Season + Quarter + GameClock + Distance + DefendersInTheBox +
##      PlayDirection + PlayerHeight + Week + Temperature + Humidity +
##      TimeDifference + HomeScoreAdvantage + PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - Temperature       1     16 183241 15883
## - PlayerHeight      1     17 183242 15883
## - Season            1     22 183248 15884
## - PlayDirection     1     24 183250 15884
## - Week              1     25 183251 15884
## - Humidity          1     31 183256 15884
## - GameClock         1     31 183256 15884
## - X                 1     32 183257 15884

```

```

## - HomeScoreAdvantage 1      65 183290 15884
## <none>                  183225 15885
## - Distance             1      89 183314 15885
## - Quarter              1     167 183392 15887
## - PlayerAge            1     211 183437 15888
## - TimeDifference        1     215 183440 15888
## - DefendersInTheBox    8    1699 184924 15908
##
## Step: AIC=15883
## Yards ~ X + Season + Quarter + GameClock + Distance + DefendersInTheBox +
##   PlayDirection + PlayerHeight + Week + Humidity + TimeDifference +
##   HomeScoreAdvantage + PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - PlayerHeight 1         17 183259 15882
## - Season        1         24 183266 15882
## - PlayDirection 1         24 183266 15882
## - GameClock     1         31 183272 15882
## - X             1         31 183273 15882
## - Humidity      1         41 183283 15882
## - HomeScoreAdvantage 1      63 183304 15883
## <none>          183241 15883
## - Distance     1         89 183331 15883
## - Week         1         89 183331 15883
## - Quarter      1        167 183408 15885
## - PlayerAge    1        203 183444 15886
## - TimeDifference 1        211 183453 15886
## - DefendersInTheBox 8    1698 184939 15906
##
## Step: AIC=15882
## Yards ~ X + Season + Quarter + GameClock + Distance + DefendersInTheBox +
##   PlayDirection + Week + Humidity + TimeDifference + HomeScoreAdvantage +
##   PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - PlayDirection 1         25 183283 15880
## - Season        1         26 183285 15880
## - GameClock     1         32 183291 15880
## - X             1         32 183291 15881
## - Humidity      1         40 183299 15881
## - HomeScoreAdvantage 1      67 183326 15881
## <none>          183259 15882
## - Distance     1         88 183347 15882
## - Week         1         90 183349 15882
## - Quarter      1        170 183429 15884
## - TimeDifference 1        211 183470 15885
## - PlayerAge    1        230 183489 15885
## - DefendersInTheBox 8    1688 184946 15904
##
## Step: AIC=15880
## Yards ~ X + Season + Quarter + GameClock + Distance + DefendersInTheBox +
##   Week + Humidity + TimeDifference + HomeScoreAdvantage + PlayerAge
##
##              Df Sum of Sq    RSS    AIC

```

```

## - X                1          20 183303 15879
## - Season           1          25 183309 15879
## - GameClock        1          31 183314 15879
## - Humidity          1          40 183323 15879
## - HomeScoreAdvantage 1          67 183351 15880
## <none>              183283 15880
## - Distance          1          88 183371 15880
## - Week              1          92 183375 15880
## - Quarter           1         172 183455 15882
## - TimeDifference     1         208 183491 15883
## - PlayerAge          1         230 183513 15884
## - DefendersInTheBox 8        1684 184967 15903
##
## Step:  AIC=15879
## Yards ~ Season + Quarter + GameClock + Distance + DefendersInTheBox +
##      Week + Humidity + TimeDifference + HomeScoreAdvantage + PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - Season          1         26 183329 15877
## - GameClock        1         31 183335 15877
## - Humidity          1         41 183345 15878
## - HomeScoreAdvantage 1         67 183370 15878
## <none>              183303 15879
## - Distance          1         90 183393 15879
## - Week              1         90 183394 15879
## - Quarter           1        172 183476 15881
## - TimeDifference     1        207 183510 15882
## - PlayerAge          1        229 183532 15882
## - DefendersInTheBox 8        1683 184987 15901
##
## Step:  AIC=15877
## Yards ~ Quarter + GameClock + Distance + DefendersInTheBox +
##      Week + Humidity + TimeDifference + HomeScoreAdvantage + PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - GameClock          1         32 183361 15876
## - Humidity            1         36 183366 15876
## - HomeScoreAdvantage 1         64 183393 15877
## <none>              183329 15877
## - Distance            1         90 183419 15877
## - Week                 1         90 183419 15877
## - Quarter              1        172 183501 15879
## - TimeDifference        1        211 183540 15880
## - PlayerAge             1        239 183568 15881
## - DefendersInTheBox    8        1717 185046 15900
##
## Step:  AIC=15876
## Yards ~ Quarter + Distance + DefendersInTheBox + Week + Humidity +
##      TimeDifference + HomeScoreAdvantage + PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - Humidity            1         37 183398 15875
## - HomeScoreAdvantage 1         64 183425 15876
## <none>              183361 15876

```

```

## - Week          1          89 183450 15876
## - Distance      1          90 183451 15876
## - Quarter       1         175 183536 15878
## - TimeDifference 1         212 183573 15879
## - PlayerAge     1         241 183602 15880
## - DefendersInTheBox 8       1706 185068 15899
##
## Step: AIC=15875
## Yards ~ Quarter + Distance + DefendersInTheBox + Week + TimeDifference +
##   HomeScoreAdvantage + PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## - HomeScoreAdvantage 1         61 183459 15874
## - Week                1         86 183484 15875
## <none>                  183398 15875
## - Distance            1         92 183490 15875
## - Quarter             1        175 183573 15877
## - TimeDifference       1        212 183610 15878
## - PlayerAge           1        235 183633 15878
## - DefendersInTheBox   8       1700 185098 15898
##
## Step: AIC=15874
## Yards ~ Quarter + Distance + DefendersInTheBox + Week + TimeDifference +
##   PlayerAge
##
##              Df Sum of Sq    RSS    AIC
## <none>                  183459 15874
## - Week                1         88 183547 15874
## - Distance            1         91 183550 15874
## - Quarter             1        166 183625 15876
## - TimeDifference       1        210 183669 15877
## - PlayerAge           1        232 183692 15878
## - DefendersInTheBox   8       1695 185154 15897

```

```
cat("Stepwise Backward Model on test set")
```

```

## Stepwise Backward Model on test set

```

```
summary(test_backward_step)
```

```

##
## Call:
## lm(formula = Yards ~ Quarter + Distance + DefendersInTheBox +
##   Week + TimeDifference + PlayerAge, data = NFL_DATA_HOLDOUT_Final)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.34  -3.30  -1.26   1.34   87.21
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    4.9791     1.0588   4.70 0.0000027 ***
## Quarter         0.1742     0.0897   1.94   0.052 .
## Distance        0.0390     0.0271   1.44   0.150
## DefendersInTheBox.L -7.7622     1.8948  -4.10 0.0000427 ***

```

```
## DefendersInTheBox.Q    0.7879    1.7613    0.45    0.655
## DefendersInTheBox.C   -1.5955    1.5613   -1.02    0.307
## DefendersInTheBox^4    1.8728    1.3789    1.36    0.174
## DefendersInTheBox^5    0.1048    1.1624    0.09    0.928
## DefendersInTheBox^6    0.2640    0.9069    0.29    0.771
## DefendersInTheBox^7   -0.3332    0.6072   -0.55    0.583
## DefendersInTheBox^8   -0.3392    0.3134   -1.08    0.279
## Week                   -0.0292    0.0206   -1.41    0.158
## TimeDifference          0.4770    0.2182    2.19    0.029 *
## PlayerAge              -0.0746    0.0324   -2.30    0.021 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.62 on 4180 degrees of freedom
## Multiple R-squared:  0.015, Adjusted R-squared:  0.012
## F-statistic: 4.91 on 13 and 4180 DF, p-value: 0.0000000128
```

```
# Bootstrap sample distribution holders
```

```
Season2018_Hold = rep(0, 1000) # 2
Distance_Hold = rep(0, 1000) # 3
Defender_L = rep(0, 1000)
Defender_Q = rep(0, 1000)
Defender_C = rep(0, 1000)
Df_4 = rep(0, 1000)
Df_5 = rep(0, 1000)
Df_6 = rep(0, 1000)
Df_7 = rep(0, 1000)
Df_8 = rep(0, 1000)
FB = rep(0, 1000)
HB = rep(0, 1000)
QB = rep(0, 1000)
RB = rep(0, 1000)
TE = rep(0, 1000)
WR = rep(0, 1000)
Temp_Hold = rep(0, 1000)
Humidity_Hold = rep(0, 1000)
Time_Diff_Hold = rep(0, 1000)
Home_Adva_Hold = rep(0, 1000)
Age_Hold = rep(0, 1000)
```

Bootstrap

```
# backward_step # This is the BackwardStep Model
# NFL_DATA_TRAIN_Filtered_Final # this is where the data is kept
# NFL_Train_Total_Model = lm(Yards ~ ., data=NFL_DATA_TRAIN_Filtered_Final)
# summary(model_boot)$coefficients[2]

set.seed(121)
for(i in 1:1000){
  Sample_Train_Data = sample_n(NFL_DATA_TRAIN_Filtered_Final, 16761, replace=TRUE)
  model_boot = lm(Yards ~ Season + Distance + DefendersInTheBox +
```

```

        Position + Temperature + Humidity + TimeDifference +
        HomeScoreAdvantage + PlayerAge, data=Sample_Train_Data)
# Grab the coefficient
Season2018_Hold[i] = summary(model_boot)$coefficients[2]
Distance_Hold[i] = summary(model_boot)$coefficients[3] #3
Defender_L[i] = summary(model_boot)$coefficients[4]
Defender_Q[i] = summary(model_boot)$coefficients[5]
Defender_C[i] = summary(model_boot)$coefficients[6]
Df_4[i] = summary(model_boot)$coefficients[7]
Df_5[i] = summary(model_boot)$coefficients[8]
Df_6[i] = summary(model_boot)$coefficients[9]
Df_7[i] = summary(model_boot)$coefficients[10]
Df_8[i] = summary(model_boot)$coefficients[11]
FB[i] = summary(model_boot)$coefficients[12]
HB[i] = summary(model_boot)$coefficients[13]
QB[i] = summary(model_boot)$coefficients[14]
RB[i] = summary(model_boot)$coefficients[15]
TE[i] = summary(model_boot)$coefficients[16]
WR[i] = summary(model_boot)$coefficients[17]
Temp_Hold[i] = summary(model_boot)$coefficients[18]
Humidity_Hold[i] = summary(model_boot)$coefficients[19]
Time_Diff_Hold[i] = summary(model_boot)$coefficients[20]
Home_Adva_Hold[i] = summary(model_boot)$coefficients[21]
Age_Hold[i] = summary(model_boot)$coefficients[22]
}

```

```

calculate_CI <- function(dist){
  se_bootstrap = sd(dist) / sqrt(length(dist))
  min_conf = mean(dist) - (1.96 * se_bootstrap)
  max_conf = mean(dist) + (1.96 * se_bootstrap)
  # Quantile Bootstrap
  sorted_dist = sort(dist)
  min = sorted_dist[.025 * length(sorted_dist)]
  max = sorted_dist[.975 * length(sorted_dist)]
  return(c(min_conf,max_conf, min,max))
}

```

```
calculate_CI(Season2018_Hold)
```

```
## [1] 0.29684 0.30920 0.10857 0.49517
```

```
calculate_CI(Distance_Hold)
```

```
## [1] 0.078872 0.080400 0.054514 0.104388
```

```
calculate_CI(Defender_L)
```

```
## [1] -7.7990 -7.6771 -9.6550 -5.8412
```

```
calculate_CI(Defender_Q)
```

```
## [1] 1.65068 1.77067 -0.23781 3.66113
```

```
calculate_CI(Defender_C)
```

```
## [1] -1.4051 -1.2995 -3.0751 0.2877
```



```

calculate_CI(Df_4)

## [1] 0.72537 0.81084 -0.61647 2.09134
calculate_CI(Df_5)

## [1] 0.21223 0.27931 -0.79968 1.37916
calculate_CI(Df_6)

## [1] 0.53384 0.58808 -0.33052 1.37683
calculate_CI(Df_7)

## [1] -0.0061014 0.0317655 -0.5721019 0.6239428
calculate_CI(Df_8)

## [1] 0.2925951 0.3125357 0.0015328 0.6175797
calculate_CI(FB)

## [1] 0.022648 0.084107 -0.925562 1.071911
calculate_CI(HB)

## [1] -1.23120 -1.11945 -2.87400 0.73868
calculate_CI(QB)

## [1] 0.12078 0.16695 -0.62256 0.83467
calculate_CI(RB)

## [1] 1.3832 1.5736 -1.5631 4.3932
calculate_CI(TE)

## [1] 1.68869 1.75826 0.63754 2.79246
calculate_CI(WR)

## [1] -0.0076818 -0.0073097 -0.0129717 -0.0014519
calculate_CI(Temp_Hold)

## [1] -0.0031879 -0.0028899 -0.0077618 0.0016044
calculate_CI(Humidty_Hold)

## [1] 0.082352 0.096248 -0.127267 0.300684
calculate_CI(Time_Diff_Hold)

## [1] -0.0076761 -0.0071092 -0.0166968 0.0015104
calculate_CI(Home_Adva_Hold)

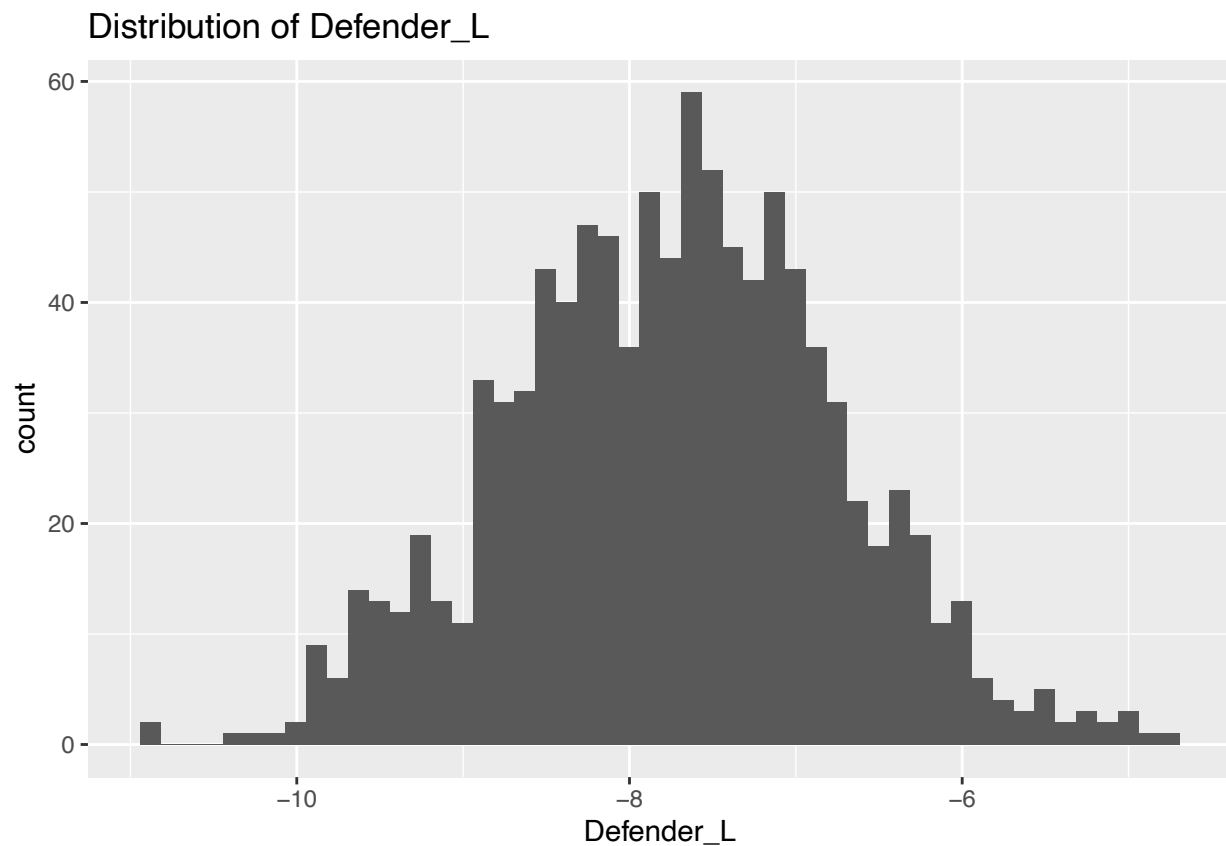
## [1] -0.047159 -0.045177 -0.077090 -0.014881
calculate_CI(Age_Hold)

## [1] 0.82705 0.83071 0.77654 0.89219
confint(backward_step)

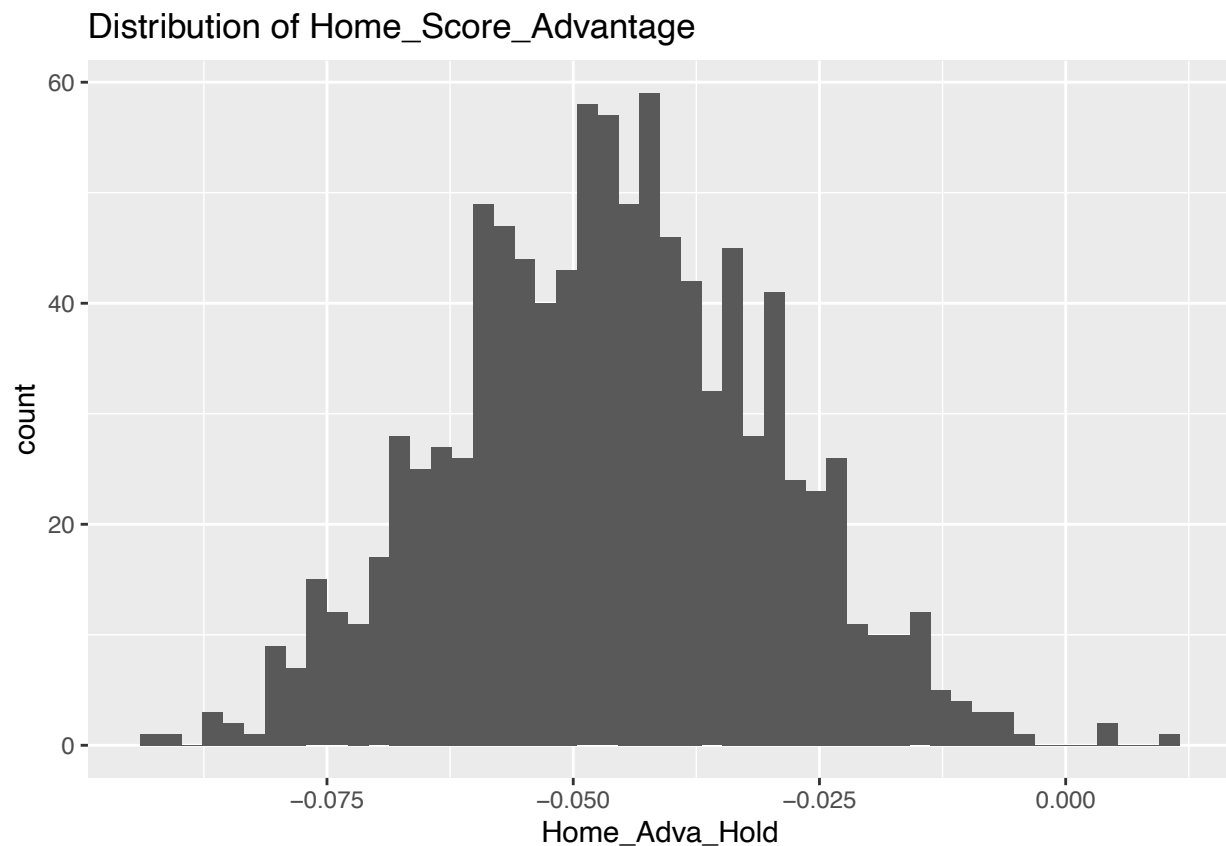
```

##		2.5 %	97.5 %
##	(Intercept)	3.79127562	12.42872468
##	Season2018	0.11070534	0.50195106
##	GameClock	0.00015695	0.00089477
##	Distance	0.05179192	0.10565643
##	DefendersInTheBox.L	-9.82963123	-5.63471996
##	DefendersInTheBox.Q	-0.28595010	3.77681310
##	DefendersInTheBox.C	-3.17000012	0.34611021
##	DefendersInTheBox^4	-0.73766275	2.15052841
##	DefendersInTheBox^5	-0.85220605	1.41165027
##	DefendersInTheBox^6	-0.28102709	1.39356774
##	DefendersInTheBox^7	-0.50656802	0.59351201
##	DefendersInTheBox^8	0.02649158	0.59745126
##	PlayerHeight	-0.09608546	0.01308703
##	PositionHB	-1.19316377	1.39767875
##	PositionQB	-3.01688181	1.06884733
##	PositionRB	-0.99955461	1.29505478
##	PositionTE	-1.21021647	4.59852705
##	PositionWR	0.45763126	3.03607147
##	Temperature	-0.01274723	-0.00165640
##	HomeScoreAdvantage	-0.01645468	0.00158494
##	PlayerAge	-0.08230533	-0.01801529

```
ggplot() +
  geom_histogram(mapping = aes(x = Defender_L), bins = 50) +
  ggtitle("Distribution of Defender_L")
```



```
ggplot() +
  geom_histogram(mapping = aes(x = Home_Adva_Hold), bins = 50) +
  ggtitle("Distribution of Home_Score_Advantage")
```



Classification. Outcome Variable: NFL Yards \geq distance.
(Whether or not they get a first down)

With all Covariates for training set

```
NFL_Train_Total_Model_c <- NFL_DATA_TRAIN_Filtered_Final
NFL_Train_Total_Model_c$FirstDown <- ifelse(
  NFL_Train_Total_Model_c$Yards >= NFL_Train_Total_Model_c$Distance, 1, 0)
NFL_Train_Total_Model_c$FirstDown <-
  factor(NFL_Train_Total_Model_c$FirstDown)
NFL_Train_Total_Model_c <- select(NFL_Train_Total_Model_c, -Yards) #remove colinear response variable f
```

For test set

```
NFL_HOLDOUT_C <- NFL_DATA_HOLDOUT_Final
NFL_HOLDOUT_C$FirstDown <- ifelse(
  NFL_HOLDOUT_C$Yards >= NFL_HOLDOUT_C$Distance, 1, 0)
```

```
NFL_HOLDOUT_C$FirstDown <-
  factor(NFL_HOLDOUT_C$FirstDown)
NFL_HOLDOUT_C <- select(NFL_HOLDOUT_C, -Yards) #remove colinear response variable from regression
```

Implementing Cross-Fold Validation for Classification to calculate validation error

and test error

```
set.seed(122)
k = 10

f <- createFolds(y=NFL_Train_Total_Model_c$FirstDown, k)
train_fold <- function (i) {
  NFL_Train_Total_Model_c[-unlist(f[i]),]
}

test_fold <- function (i) {
  NFL_Train_Total_Model_c[unlist(f[i]),]
}

accuracyR <- c()

for (i in 1:k) {
  glm_Model = glm(FirstDown ~ ., data=train_fold(i),
                  family = binomial)
  predict_result <- predict(glm_Model, newdata=test_fold(i), type="response")
  predict_logit <- ifelse(predict_result > 0.5, 1, 0)
  t <- table(predict_logit, test_fold(i)$FirstDown)
  accuracyR[i] = (t[1,1]+t[2,2])/dim(test_fold(i))[1]
}
cat("Error of 10-fold validation is: ", 1 - mean(accuracyR), "\n")

## Error of 10-fold validation is: 0.18093
```