NBA 2021 Predictions:

Spread, Total, and Offensive Rebounds

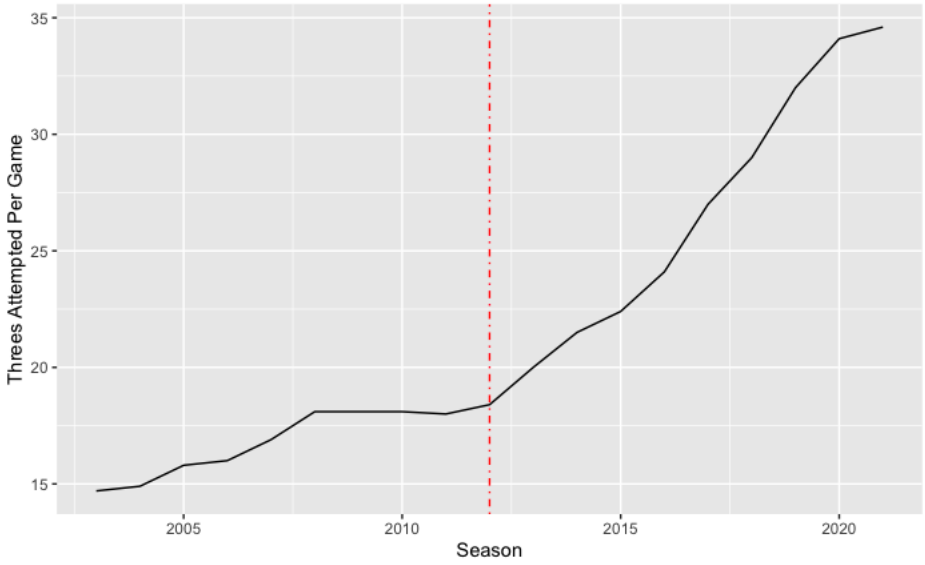
Team 4: Sayak Basu, Alexa Edwards, Omar Hilal Shaban, and Ryan Shock

I. Data Information

A. Cleaning and Joining

In order to begin working with the data sets, we first loaded them into R. The three data sets were Games, Games details, and Teams. Games.csv contained data at the team level for many different seasons ranging from 2003 to 2021; it contains information from the game box score about both the home and away teams. It is worth noting that this data is at the team level. Games_details.csv contains data at the player level for each game. Similarly to games.csv, this data set contains data from the 2003 season to the 2021 season. Lastly, teams.csv is composed of general team data such as the year it was founded and its coaches.

Once the datasets were downloaded into R, we updated the games dataset to include the box score information from the most recent games (up until March). These games were found on Kaggle. We then assessed the data in this dataset and decided to only include games from 2012 to present. In 2012 (red line in graph), there was a drastic jump in the average number of 3 point shots attempted and made by each team. This trend has persisted to the present day with the average number of three point attempts continuing to rise each season, thus we felt it was best to only include the years that reflect the current trends. The increase in 3 point shots affects other aspects of the game as well, such as offensive rebounds. 3 pointers have longer rebounds than 2-point field goals closer to the basket.



To ensure the data sets were uniform, we filtered games_details to also only include games from 2012 to present. In addition, we adjusted the minute variable. This variable contains the playing time of each player. Initially, the time was in minute:second format; we changed this so that the time was strictly in decimal minutes. In other words, 2 minutes 30 seconds is now represented by 2.5 minutes. By doing this, we have changed the variable to be numeric and more useful for our predictions.

In the teams dataset, we created a variable called team name by combining the city and the team's nickname. For example, the city Boston was combined with team nickname Celtics to

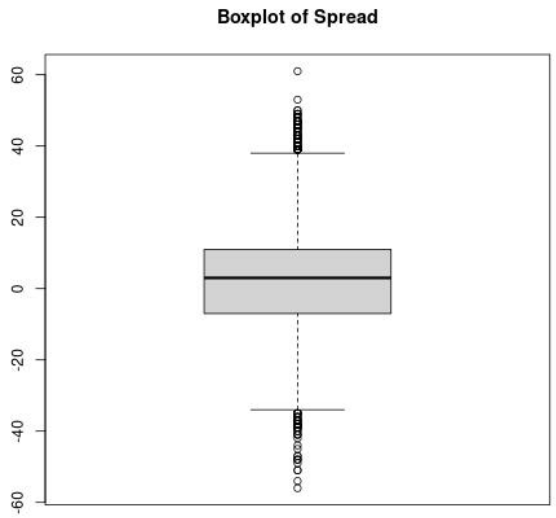
read as Boston Celtics. This change enables us to more easily merge the data for our final predictions.csv file.

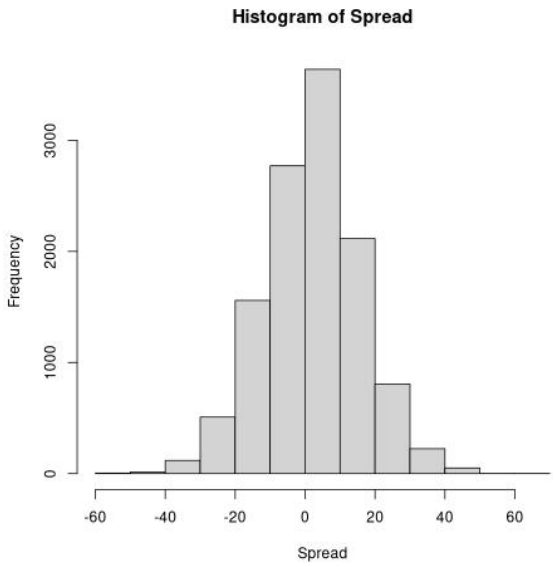
Lastly, we created a dataset that we titled games_final. This is the data set we will use as our primary dataset when making our predictions. Games_final was formed by merging games with the aggregated data from games_details. Here, we split every observation into two. Thus, creating a separate observation for the home and away team of each game. More specifically, instead of having a home and away team we now have a game with a team and an opponent. Though this change may not have been entirely necessary, it is a simplification that streamlines our prediction approach. It was made to allow us to predict the points by individual teams in each game.

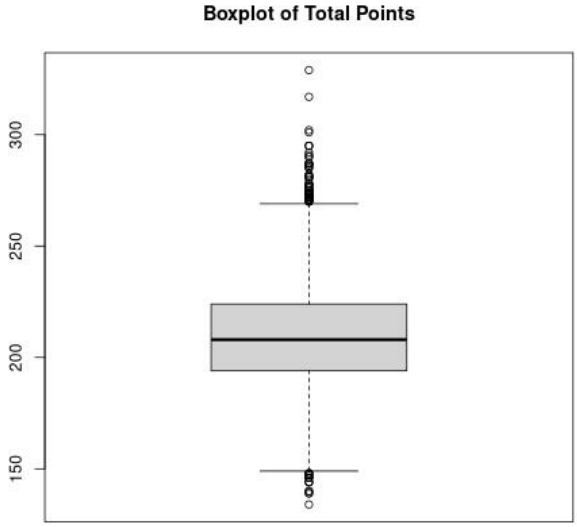
We did not notice a large amount of missing data. However, we did see many issues with missing games from the 2003 season in games_details. This issue was avoided since we chose to drop all seasons before 2012. In many instances, the minutes variable is missing, but this means that the player did not play. In other situations, players' minutes were missing, so we divided total minutes played by the number of games played to have an estimate of the minutes the player played. The total minutes played and the number of games played were found on the Basketball Reference shooting stats. Other than minutes, NA means the player or that observation was not used for the prediction. In other words, the row was removed. Additionally, we dropped the +/- variable because we are predicting game outcomes at the team level rather than the player level. We did not see it as meaningful to aggregate +/- like the other variables.

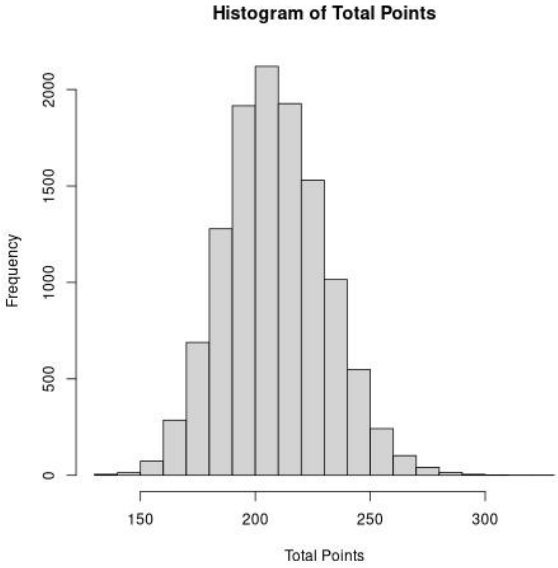
To address outliers in our data, we first viewed box plots of spread, total points and total offensive rebounds. Any points outside of the two lines extending from the first and third quartiles would be labeled as extreme data points and could potentially be outliers. We calculated the interquartile range from subtracting the third and first quartiles and removed data that was not within the range of the first quartile minus 1.5 times the IQR and the third quartile plus 1.5 times the IQR. This process was done to remove 129 extreme games for spread, 85 extreme games for total points, and 24 extreme games for total offensive rebounds. Our sample has 11,811 games from seasons 2012-2021, so we feel confident that our sample size is large enough to remove 235 games (3 games had extreme values in at least two of the variables) from this sample as we still have over 11,500 games to generate models with.

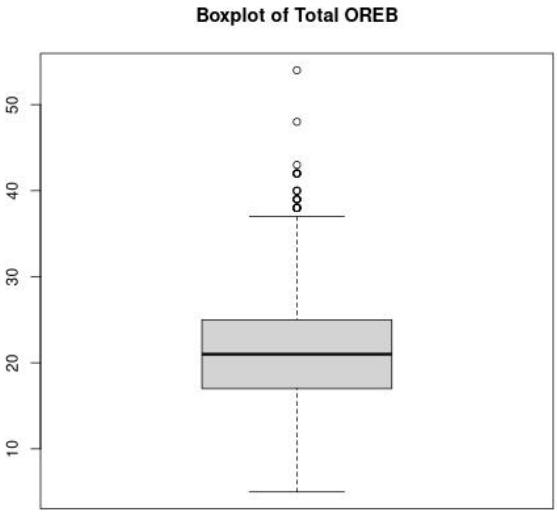
Below are the box plots before the outliers were removed, and beside each plot is a histogram to show the data once the outliers were removed. The adjusted data more closely follows a normal distribution.

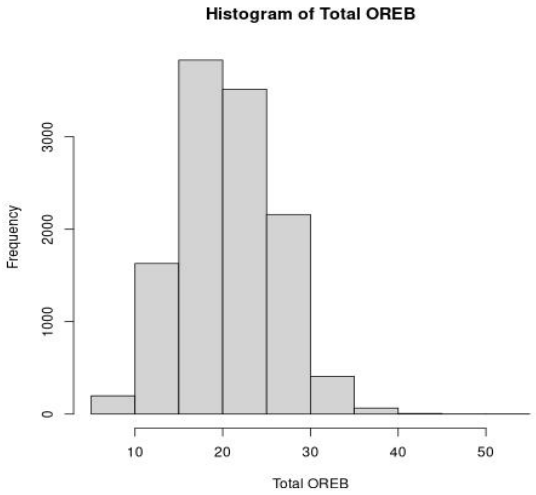

Boxplot of Spread
60
40
20
0
-20
-40
-60


Histogram of Spread
Frequency
3000
2000
1000
0
-60
-40
-20
0
20
40
60
Spread


Boxplot of Total Points
300
250
200
150


Histogram of Total Points
Frequency
2000
1500
1000
500
0
150
200
250
300
Total Points


Boxplot of Total OREB
50
40
30
20
10


Histogram of Total OREB
Frequency
3000
2000
1000
0
10
20
30
40
50
Total OREB

B. Engineered Variables

One variable that we engineered is the offensive rebound percentage. This percentage was found for each team by game. The season average for each team was then computed. Team FGA-Team FGM is the number of missed shots by the team. Each of these missed shots is an opportunity for the shooting team to gain an offensive rebound and have an additional possession. If the defending team gets the rebound, the rebound is recorded as a defensive rebound for the defending team. The maximum number of offensive rebounds a team can get is determined by the number of missed field goals. So when predicting offensive rebounds it is important to factor in the number of opportunities each team has to secure offensive rebounds. We can attempt to use the teams field goal percentage to predict the number of opportunities the team has for offensive rebounds. By calculating the team offensive rebounding percentage we also found the opponents defensive rebounding percentage. Defensive rebounding is very important because it allows your team to gain possession and transition from defense to offense.

$$Team\ Offensive\ Rebound\ Percentage\ = \frac{Team\ Offensive\ Rebounds}{Team\ Field\ Goals\ Attempted - Team\ Field\ Goals\ Made}$$

$$Opponet's\ Defensive\ Rebound\ Percentage\ = 1 - Team\ Offensive\ Rebound\ Percentage$$

C. Outside Data

For our outside data, we web scrapped from Basketball Reference. For the team level, this data included wins percentage, average game statistics, offensive ratings, defensive ratings, net ratings, and margin of victory. The margin of victory is how many points the team wins by on average. We expected this variable to be particularly beneficial for predicting the *Spread*. This data needed to be cleaned as well. We renamed the variables to the column names, deleted the first two rows to move row 3 to be the first row, and made a vector of different hyperlinks to each webpage corresponding to each season. Then, by calling that element of the vector, we were able to combine all seasons, 2012-present, into one data set. We added a season variable to distinguish the teams' different seasons.

At the player level, we used Basketball Reference to find additional data. This data included offensive rebound percentages, per season averages, shooting percentages, and advanced statistics. This data was web scrapped from two different links. One of the links was focused on shooting percentages from different ranges including different distances from the basket, 2 point and 3 point shots, and the number of shots attempted and made. The other link was specifically advanced stats which we only used offensive rebound percentages. Next, we merged the two data sets by seasons, player, position, age, and team. At this point, we were able to merge all the seasons together by using a seasons identifier.

Using non-adjusted offensive/defensive/margin of victory because we are also factoring in the opponent's ratings. Adjusted offensive/defensive/margin of rating ratings account for the strength of the opponent (do not want to adjust for opponent twice). Although the offensive and

defensive ratings are per 100 possessions and not per game we still believe it is still a good metric for how efficient the teams offense and defense is.

D. Variable Names

Our variable names are all abbreviated. Below is a list of the full variable names:

PTS- Points	PCT- Percentage	MIN- Minutes
FG- Field Goal	FG3- 3 point field goal	OREB- Offensive rebound
STL- Steal	AST- Assist	REB- Rebound
BLK- Block	TO- Turnovers	PF- Personal fouls
OFF- Offensive	DEF- Defensive	RTG- Rating
POINT_DIFF- Point Differential		HOME- Home team
FGA/ FTA- Field goals attempted/ Free throw attempted		
FGM/ FTM- Field goals made/ Free throw made		

Most of these variables are self explanatory. However, point differential is the difference between points scored and points allowed. For example, in a game where the Knicks play the Celtics, the Knicks' POINT_DIFF is the difference between the points they scored and the points they allowed the Celtics to score.

E. Splitting into Training, Testing, and Validating

Once our data was finalized, we split it into three parts to ensure that we can train and cross-validate our models appropriately. As such, we allotted 70% of the sample for training, 20% for testing, and the last 10% for validating. These groups are chosen randomly, however we have set a seed to ensure that our results and models are deterministic and reproducible. The training portion was employed to train our different models and produce coefficients, while the testing dataset is used to evaluate the performance of the models against one another. Ultimately, testing also prevents overfitting to the training data. Lastly, the validation portion of our sample is used to report the performance of our final model of choice.

II. Methodology

When faced with the task of predicting the spread, total, and offensive rebounds our team first discussed whether we would predict at the team or player level. Ultimately, we decided to predict at the team level. This decision was made based on the idea that we have no way to know who will play on a given night especially due to injuries and COVID-19. We feel that it is more important how the team plays as a whole rather than how an individual player performs. For example, if the best player has an awful game, he will be replaced by an individual on the bench and his poor performance is no longer relevant to our predictions. Now, the player coming in off the bench will influence the game. Furthermore, while predicting at the player level allows for higher resolution inference, the error that would be accumulated by running the model for every player in a game would likely outweigh the benefits.

Thus, our approach is to train our model at the game level using past games from 2012 to 2021. While this approach is effective, it leads us to a road block. Our model expects us to give it current statistics such as field goal percentages which is not possible for future games. Instead, we decided to use the average or expected statistics based on the teams' average statistics for the season. These values can be found in the datasets we collected from outside resources (See Outside Data section).

The goal is to find the best model. In order to do this we have to train several models and use cross-validation to select the best one. We will be selecting based on the criteria of lowest root-mean-squared error (RMSE) as our objective is to minimize RMSE for future predictions. There are two aspects to testing models that are independent of each other: the type of model it is (linear, polynomial, KNN etc) and the variables we use to predict the outcome variable.

A. Spread and Total

Spread and total points are both derived from the home teams points and the away teams points, thus, we decided to generate a model that predicts a teams points based on their opponent. We then use this model to generate spread and total points by the difference between the home and away teams point and the sum of the home and away team points respectively.

We decided to use a bi-directional stepwise regression initially by predicting with season averages for each team. Our first iteration consisted of running the model with all of our initial predictors (PTS, FG_PCT, FGA, FGM, FT_PCT, FTA, FTM, FG3_PCT, FG3A, FG3M, AST, REB, OREB, OREB_PCT, MIN, STL, BLK, TO, PF, WIN_PCT, POINT_DIFF, OFF_RTG, DEF_RTG, HOME, OPP_PTS, OPP_FG_PCT, OPP_FGA, OPP_FGM, OPP_FT_PCT, OPP_FTA, OPP_FTM, OPP_FG3_PCT, OPP_FG3A, OPP_FG3M, OPP_AST, OPP_REB, OPP_OREB, OPP_OREB_PCT, OPP_MIN, OPP_STL, OPP_BLK, OPP_TO, OPP_PF, OPP_WIN_PCT,OPP_POINT_DIFF, OPP_OFF_RTG, OPP_DEF_RTG). Added squared and cubic terms of both FG_% and OPP_FG% and interaction term between FG% and OPPFG% After running the initial model, we received an RMSE value of 2.1384 for Spread and 0.5833 for Total points. The adjusted R-squared of 0.993, revealing that 99.3% of the data fit the regression model, signifying that the model is a very substantial representation of our data.

For our second iteration, we dropped variables field goals attempted, offensive rebound percentage, steals, opponent's free throw attempts, opponent's assists, opponent's offensive and defensive ratings, interaction between the team and opponent's field goal percentage as they were not significant after utilizing both the forward method and the backward method, and our stepwise regression terminated after this iteration, signifying that our model was optimal. Although this yielded the same adjusted R-squared value of 0.993, the RMSE for both spread and total points decreased slightly from 2.1384 to 2.1383 and from 0.5833 to 0.5785, respectively.

Our model predicts points scored by a team based on the teams season averages and their opponent's season averages; its adjusted R-squared value is 0.993. We run this model once to calculate predicted points for the home team and again to generate predicted points for the away

team. Our predicted spread is the difference between predicted home team points and away team points, while our predicted total points is the sum of home team points and away team points. Although the same model is used to form the components of spread and total points, they have different RMSE values as they are different variables.

The second type of model we utilized was the K-Nearest Neighbor Regression Model. This model takes the K nearest values and develops the predicted value based on the average of those K values. The subset of predictor variables from the final stepwise model (excluding interaction and polynomial terms) for points is a good basis of predictors to use as we know they are statistically significant for points. We tested four K-values: 5, 10, 25, 50 using the aforementioned basis from the final stepwise model. The range after removing outliers for points (scored by a single team) is 86 (minimum of 63 and maximum of 149); this is notable since the KNN model is predicted based on the values of other data points. After regressing each K-value on the testing data, we found the optimal K-value is K=10 which gave us the lowest RMSE for both spread (3.913420) and total points (3.630423).

The following table lists the different Points models that were attempted and their RMSE values produced from predicting on the testing dataset:

Spread and Total RMSE Results per Model		
Model	**RMSE for Spread**	**RMSE for Total**
Baseline Polynomial Regression	2.1384	0.5833
Bi-directional Stepwise Polynomial Regression	**2.1383**	**0.5785**
5-NN	3.9944	4.0679
10-NN	3.8870	3.7058
25-NN	4.0271	3.8726
50-NN	4.2316	4.1403

Best model for Spread and Total

After employing the bi-directional stepwise regression and the K-Nearest Neighbor Regression, we determined that the model that minimized the RMSE value was the completed stepwise regression model (2nd iteration after the removal of insignificant variables) as the RMSE for the optimal model for spread was 2.1383 and 0.5785 for total points while the RMSE for the KNN model was 3.9134 for spread and 3.6304 for total points. We ran our best model for predicting spread and total points, then completed stepwise, on our validating data subset and generated an RMSE for spread and total points of 2.1040 and 0.6218, respectively.

Additionally, our bi-directional stepwise regression model yielded an extremely high adjusted R-squared of 0.993. Since this adjusted R-squared is so close to 1, we are confident in our model's ability to predict the points of both the home and away teams. With this prediction, we then calculate the values for spread and total points for the remainder of the 2021 NBA season. This model uses the season averages of statistics for each team as the predictor variables. It is a polynomial regression model with squared and cubed terms of field goal percentage and opponent field goal percentage. We identified field goal percentage as a potentially key predictor for points scored. The inclusion of third degree polynomial terms of both the predicted and opponents field goal percentage as statistically significant predictors indicates that field goal percentage is a key predictor of points scored. This makes sense as the higher the opponent's field goal percentage the more points the predicted team would have to score to keep pace with the opposing team.

Results from the Best Model for Points (Spread and Total)

```
Coefficients:
                      Estimate Std. Error  t value Pr(>|t|)
(Intercept)           1.690e+00  5.297e-01    3.191 0.001419 **
poly(FG_PCT, 3)1      1.760e+02  4.527e+00   38.871  < 2e-16 ***
poly(FG_PCT, 3)2     -5.420e+00  1.041e+00   -5.207 1.94e-07 ***
poly(FG_PCT, 3)3     -1.369e+00  1.030e+00   -1.329 0.183905
FGM                   1.675e+00  6.825e-03  245.392  < 2e-16 ***
FT_PCT                1.429e+00  2.335e-01    6.117 9.74e-10 ***
FTA                  -2.098e-02  8.279e-03   -2.534 0.011300 *
FTM                   9.201e-01  1.055e-02   87.249  < 2e-16 ***
FG3_PCT              -7.032e-01  2.708e-01   -2.597 0.009416 **
FG3A                 -1.165e-02  4.035e-03   -2.887 0.003900 **
FG3M                  1.015e+00  1.072e-02   94.714  < 2e-16 ***
AST                   1.411e-02  2.256e-03    6.254 4.10e-10 ***
REB                   6.858e-02  3.617e-03   18.961  < 2e-16 ***
OREB                  7.026e-02  5.150e-03   13.642  < 2e-16 ***
MIN                  -4.059e-01  2.464e-03 -164.734  < 2e-16 ***
BLK                  -1.326e-02  3.715e-03   -3.569 0.000359 ***
TO                   -1.415e-01  4.541e-03  -31.170  < 2e-16 ***
PF                   -5.287e-03  3.256e-03   -1.624 0.104399
WIN_PCT              -6.817e-01  2.262e-01   -3.013 0.002589 **
MOV                   1.712e-01  4.184e-02    4.092 4.30e-05 ***
OFF_RTG              -1.353e-01  4.058e-02   -3.334 0.000857 ***
DEF_RTG               1.428e-01  4.050e-02    3.526 0.000423 ***
HOME                  3.413e-02  1.704e-02    2.003 0.045158 *
OPP_PTS               8.609e-01  3.954e-03  217.740  < 2e-16 ***
poly(OPP_FG_PCT, 3)1 -2.358e+02  1.140e+01  -20.682  < 2e-16 ***
poly(OPP_FG_PCT, 3)2  5.373e+00  1.067e+00    5.036 4.80e-07 ***
poly(OPP_FG_PCT, 3)3  2.053e+00  1.033e+00    1.987 0.046955 *
OPP_FGA              -1.556e-01  1.090e-02  -14.264  < 2e-16 ***
OPP_FGM              -1.214e+00  2.330e-02  -52.086  < 2e-16 ***
OPP_FT_PCT           -7.417e-01  1.071e-01   -6.925 4.54e-12 ***
OPP_FTM              -7.778e-01  4.536e-03 -171.472  < 2e-16 ***
OPP_FG3_PCT           6.218e-01  2.745e-01    2.265 0.023530 *
OPP_FG3A              1.016e-02  4.107e-03    2.473 0.013404 *
OPP_FG3M             -8.766e-01  1.158e-02  -75.674  < 2e-16 ***
OPP_REB               9.653e-03  3.621e-03    2.666 0.007681 **
OPP_OREB              3.548e-02  1.807e-02    1.963 0.049658 *
OPP_OREB_PCT         -4.561e+00  8.107e-01   -5.625 1.89e-08 ***
OPP_MIN               4.052e-01  2.476e-03  163.669  < 2e-16 ***
OPP_STL               1.107e-02  4.405e-03    2.513 0.011982 *
OPP_BLK               2.811e-02  3.690e-03    7.618 2.73e-14 ***
OPP_TO                1.198e-01  3.811e-03   31.436  < 2e-16 ***
OPP_PF                1.905e-02  3.237e-03    5.887 4.01e-09 ***
OPP_WIN_PCT           3.349e-01  2.236e-01    1.498 0.134230
OPP_MOV              -1.564e-02  7.360e-03   -2.126 0.033552 *
---
```

B. Offensive Rebounds

The bi-directional stepwise polynomial model for offensive rebounds predicts the offensive rebounds for a team records in a game based on their season average statistics and their opponent. To generate total offensive rebounds in a game, we run the stepwise model for each team and take the sum of their offensive rebounds. Our initial stepwise model consisted of our initial predictors with squared and cubic terms of both offensive rebound percentage and opponent offensive rebound percentage as well as the interaction between both teams' offensive rebound percentages. The first iteration of the model had an adjusted R-squared value of 0.9861 and an average RMSE of 0.6156. The polynomial and interaction terms of the predicted team's offensive rebound percentage were statistically significant. This indicates offensive rebound percentage is a strong predictor of offensive rebounds for the predicted team in a game. When engineering this variable we believed it would be a strong predictor of total offensive rebounds and this stepwise model supports that hypothesis.

The second iteration removed insignificant variables from the first iteration which were the assists, steals, win percentage for both the predicted and the opposing teams. Additionally, for the predicted team, personal fouls, binary indicators for home/away were removed. Field goals made, three point field goal percentage, three point attempts, free throw percentage, and the squared and cubic terms of offensive rebound percentage were removed for the opposing team. The second iteration of the model had an adjusted R-squared value of 0.9861 and an average RMSE of 0.6146. The adjusted R-squared value did not change even though we removed 14 variables from the first iteration to the second since the adjusted R-squared penalizes based on the number of predictors, and the second iteration has 14 fewer predictors than the first iteration.

Next, we performed the K-Nearest Neighbor Regression test. Similarly to the K-Nearest Neighbor Regression Models we used for Spread and Total, we used the subset of predictor variables from the final stepwise model with the exception of the interaction and polynomial terms. After removing the outliers, the data ranges from 0 to 27 which is important since the KNN model is predicted based on the values of other data points. Next we proceeded to run the model with multiple values for K: K = 5, 10, 25, and 50. Unlike in the spread and total model, K=25 resulted in the strongest KNN model since it had the lowest RMSE of 3.6176 not K=10

Finally, we decided to train a neural network model, using the same predictors that were supplied by the stepwise regression algorithm. OREB is a nuanced statistic in a basketball game with loose relationships to other predictors compared to Points. These nuanced relationships in the data are beyond human grasp but are often ideal for neural networks which are designed to find patterns in complex problems and data structures. This was the underlying reason behind attempting a neural net model. Sure enough, the model resulted in a better RMSE than the KNN Regression model, however, it was still not as strong as the Bi-directional Stepwise Polynomial Regression.

The following table lists the different OREB models that were attempted and their RMSE values produced from predicting on the testing dataset:

OREB RMSE Results per Model	
Model	**RMSE**
Baseline Polynomial Regression	0.6156
Bi-directional Stepwise Polynomial Regression	**0.6146**
5-NN	3.6792
10-NN	3.6196
25-NN	3.6176
50-NN	3.6938
Neural Net	0.8252

Best Model for Offensive Rebounds

The model that minimizes RMSE for offensive rebounds is the bi-directional stepwise regression (2nd iteration after the removal of insignificant variables). This is consistent with our results for predicting spread and total points. The KNN model and the neural network had higher RMSE values for offensive rebounds. The RMSE for total offensive rebounds is 0.6499 after running the best model through our validating subset of data.

Similar to our bi-directional stepwise regression model for points, the stepwise regression model for offensive rebounds also had an extremely high adjusted R-squared of 0.9861. This adjusted R-squared is so close to 1, we are very confident in our model's ability to predict the offensive of both the home and away teams. When predicting future games, the best model is used with the season averages of statistics for each team as the predictor variables. It is a polynomial regression model with squared and cubed terms for offensive rebounding percentage as well as an interaction term between the two teams' offensive rebounding percentages. OREB percentage was one of the variables we engineered and its polynomial and interaction terms appear to be statistically significant in this model indicating that our engineered variable is a key predictor of offensive rebounds.

Results from the Best Model for Offensive Rebounds

```
Coefficients: (1 not defined because of singularities)
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)            -5.960870   0.422597 -14.105  < 2e-16 ***
PTS                     0.016389   0.003161   5.184 2.19e-07 ***
FG_PCT                 12.926156   0.706251  18.302  < 2e-16 ***
FGA                     0.314813   0.003949  79.722  < 2e-16 ***
FGM                    -0.423243   0.009996 -42.340  < 2e-16 ***
FT_PCT                 -0.145724   0.101151  -1.441 0.149700
FTA                     0.014547   0.003594   4.048 5.19e-05 ***
FTM                    -0.024973   0.005395  -4.629 3.71e-06 ***
FG3_PCT                -0.878244   0.118502  -7.411 1.32e-13 ***
FG3A                   -0.018795   0.001774 -10.595  < 2e-16 ***
FG3M                    0.025668   0.005691   4.510 6.52e-06 ***
REB                     0.029084   0.001562  18.616  < 2e-16 ***
poly(OREB_PCT, 3)1    392.710526   1.720139 228.302  < 2e-16 ***
poly(OREB_PCT, 3)2      8.620436   0.447990  19.242  < 2e-16 ***
poly(OREB_PCT, 3)3     -8.009846   0.445835 -17.966  < 2e-16 ***
MIN                    -0.014178   0.001635  -8.669  < 2e-16 ***
BLK                     0.005491   0.001567   3.504 0.000460 ***
TO                      0.005187   0.001720   3.016 0.002567 **
MOV                     0.033272   0.018098   1.838 0.066015 .
OFF_RTG                -0.031565   0.017362  -1.818 0.069077 .
DEF_RTG                 0.027032   0.017334   1.559 0.118909
OPP_PTS                 0.007607   0.002843   2.676 0.007462 **
OPP_FG_PCT              0.939314   0.480928   1.953 0.050823 .
OPP_FGA                -0.027413   0.002842  -9.645  < 2e-16 ***
OPP_FTA                -0.018356   0.001452 -12.645  < 2e-16 ***
OPP_FTM                 0.008364   0.003227   2.592 0.009549 **
OPP_FG3M               -0.008176   0.003012  -2.715 0.006641 **
OPP_REB                -0.029776   0.001567 -19.000  < 2e-16 ***
OPP_OREB                0.062122   0.007436   8.354  < 2e-16 ***
OPP_MIN                 0.014125   0.001637   8.631  < 2e-16 ***
OPP_BLK                -0.007097   0.001603  -4.427 9.60e-06 ***
OPP_TO                 -0.005565   0.001711  -3.252 0.001149 **
OPP_PF                 -0.002481   0.001384  -1.792 0.073180 .
OPP_MOV                -0.004274   0.001254  -3.407 0.000657 ***
OPP_OFF_RTG             0.004591   0.001692   2.713 0.006667 **
OPP_OREB_PCT            2.139551   0.389115   5.499 3.89e-08 ***
OREB_PCT:OPP_OREB_PCT  -9.459874   0.747276 -12.659  < 2e-16 ***
---
```


mood when we submit
2016 NBA CHAMPS