STA302 Final Project:

Are NBA Players Underpaid or Overpaid?

Andy Vu
Timothy Regis
Kashaun Eghdam

Department of Statistical Sciences, University of Toronto
STA302: Methods of Data Analysis I
Prof. Nnenna Asidianya

August 12, 2020

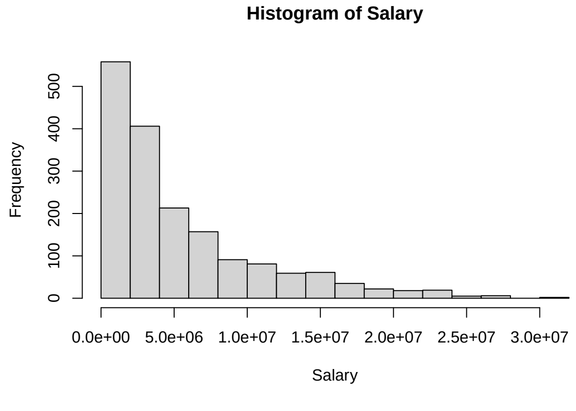
Introduction

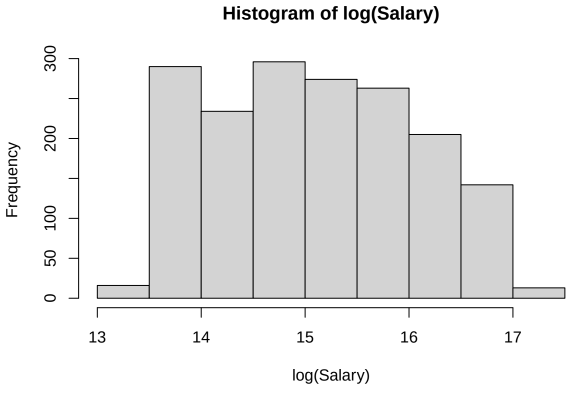
NBA superstars like Lebron James and Kevin Durant make up some of the highest-paid individuals in the US, taking home annual salaries upwards of $30 million from the NBA alone. Unlike a regular hourly wage job, however, these salaries are determined based on a number of unique and rapidly changing factors. Consequently, there is a great fascination with NBA players' salaries and a lot of speculation on what factors put these players at the top-earning bracket of their sport. Additionally, with an association that advocates heavily for inequality, it is important to gain an understanding of the varying levels of over or underpayment for players in the league and make predictions as to what their salaries should be. For this analysis we are going to be studying how factors like; points per game, effective field goal percentage, age, and more, affect a player's salary in the NBA. As a complementary, we will also get a look at the levels of mispayment estimated by our model. Through this we will attempt to identify the most significant factor(s) in determining an NBA player's salary. Our analysis will revolve around using multiple linear regression models to track a variety of factors' relationships with a player's salary. We have downloaded the dataset for this model, created by Fernando Blanco (2018), from kaggle. We cleaned the data up in excel, isolating for the specific response and predictor variables we wanted. Using R, we imported the data and developed a regression model around it using the players' salaries as the response variable and studied the relationship. We will then use a variety of testing methods to compare how our regression model changes at each step to eliminate predictor variables that matter very little in the relationship until we are left with only the most significant variables. We have decided to train our model on the data from the 2012-2016 NBA seasons and will be using the data from the 2017 NBA season for model validation.

Exploratory Data Analysis

Dependent Variable

The dependent variable we have chosen for our study is the salary of each player selected. This salary is purely based on what the NBA pays its players and does not include the additional money they may gain from outside endorsements.





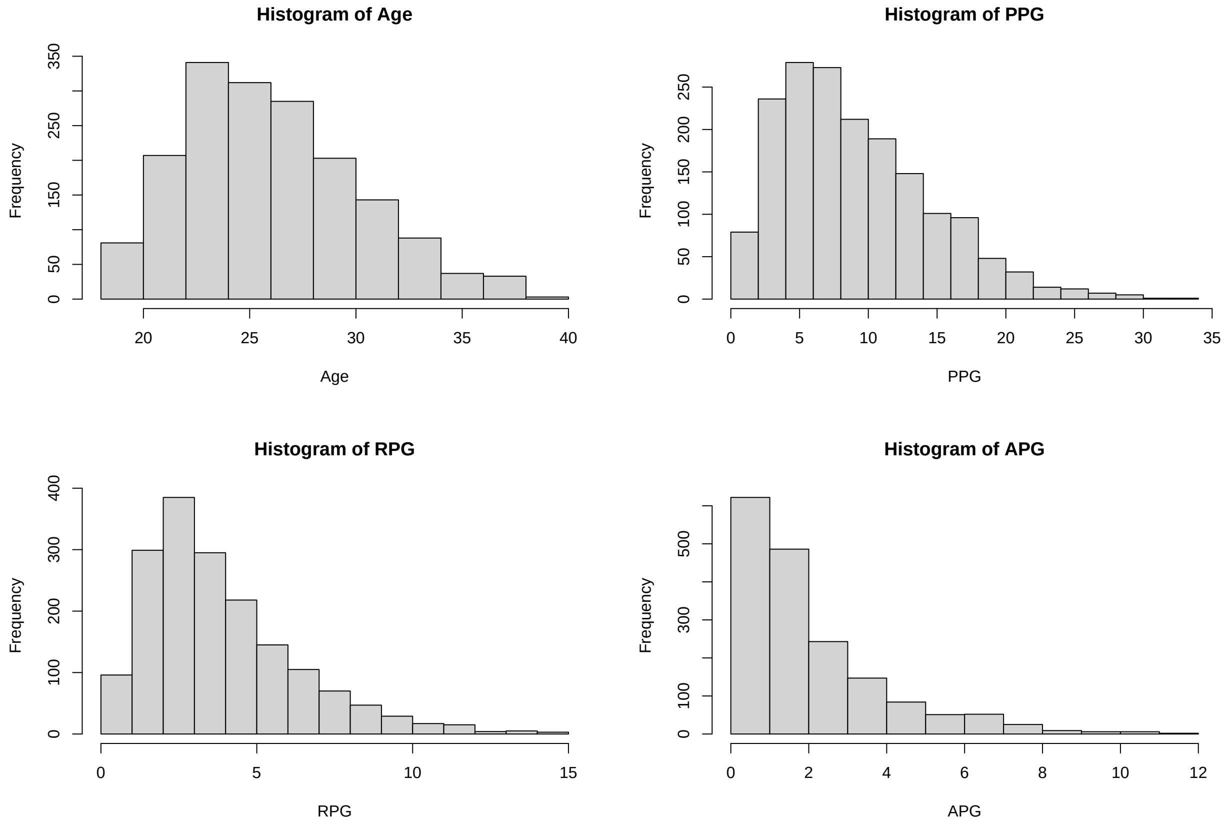
We notice that the distribution of our `Salary` values are heavily right skewed which is understandable as there are obviously fewer players earning the highest salaries and more players earning lower or entry level salaries. To compensate for the skewness, will apply the apply the log function to the values of `Salary`. This transformation has made the distribution for the `Salary` symmetric and no longer skewed, as seen on the right. This will ensure that the residuals of our model will be Gaussian.

Independent Variables

For our independent variables we chose a variety of different player statistics including; age, points per game (PPG), rebounds per game (RPG), assists per game (APG), minutes per game (MPG), and effective field goal percentage (eFG). `Age` is simply each player's age during each year of play, the per game statistics (`PPG`,`RPG`, `APG`, `MPG`) were all calculated by summing each player's respective values in each year and dividing by the number of games they were active in. Lastly, the effective field goal percentage (`eFG`) was calculated by $\frac{(\mathbf{FG}+0.5\times\mathbf{3P})}{\mathbf{FGA}}$, where **FG** is the number of Field Goals, **3P** is the number of 3-Point Field Goals, and **FGA** is the number of Field Goal Attempts (Basketball Reference, 2020). All of our chosen predictor variables are continuous.

We naturally chose these predictors as we feel that they are likely the most important factors in allowing players to earn top salaries. A basketball team wins by getting more points than their opponent in a game, as more wins bring teams closer to a championship ring, it is logical that `PPG` and `eFG` will be extremely influential in determining player salary due to their direct relation with points scored. A similar reasoning can be applied to `RPG` and `APG` as assists and rebounds often turn over into points scored.

There are an extensive number of factors that may play a role in determining salary that we decided to not include for a number of reasons. An example of which are Fouls. We decided to leave a score for fouls out of our model because unlike `PPG` or `RPG`, human error and quick judgement calls heavily affect these values. We often see things like higher value players not receiving the same foul call for a similar action taken by a lower valued player. This apparent bias toward these players brings up controversy and so we decided that fouls per game would not be a good predictor of player salary.

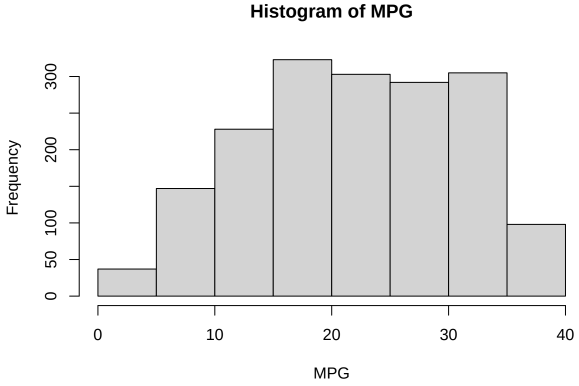


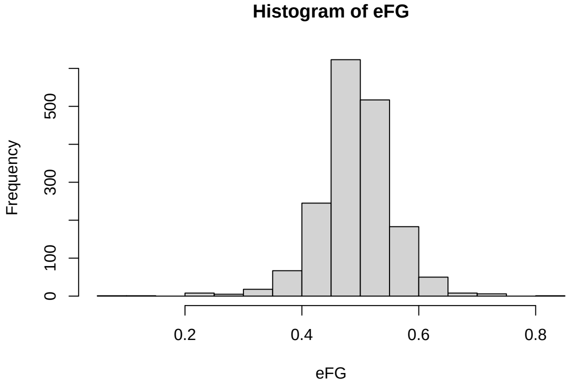
Histogram of Age

Histogram of PPG

Histogram of RPG

Histogram of APG





From the histogram of the `Age`, we notice that its distribution is somewhat right-skewed, however, this is negligible to our model. The histograms of `PPG`, `RPG` and `APG` have distributions that are all right skewed with `APG` being the most right-skewed. This can all be explained in the same way as the skewed salary distribution. `PPG`, `RPG`, and `APG` all share this skewness due to the decreasing amount of players who score higher values in these categories. Similarly, `Age` sees its skewness since the majority of players are active only for their healthiest years (20 - 30). Since the distributions of the covariates do not affect the error terms of our model, we will leave them as is for now. Lastly we notice that the histograms showing the distributions of `MPG` and `eFG` are symmetric which should be ideal for our model.

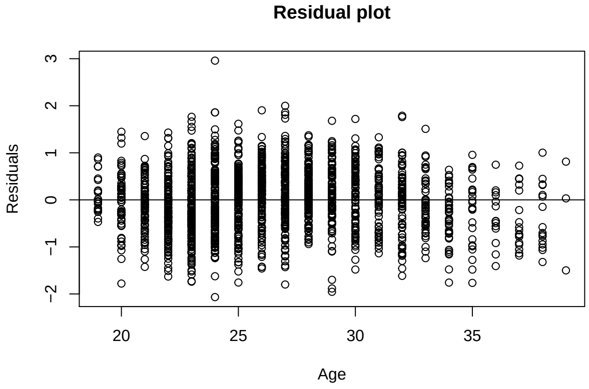
Model Development

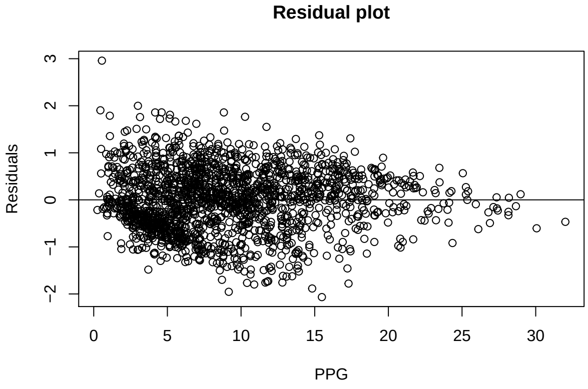
With the decision of the transformed response variable and predictor variables, we can finally produce our first model.

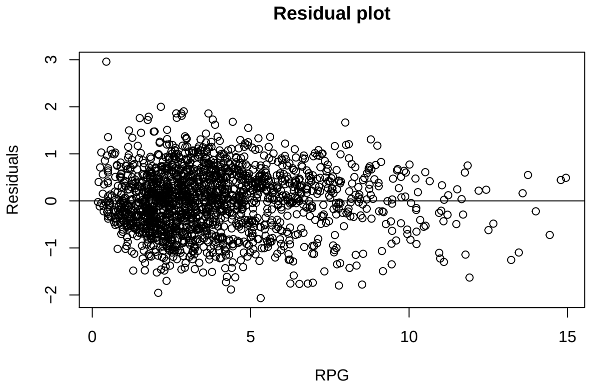
```
lm(formula = log(Salary) ~ Age + PPG + RPG + APG + MPG + eFG, data = nba)
```

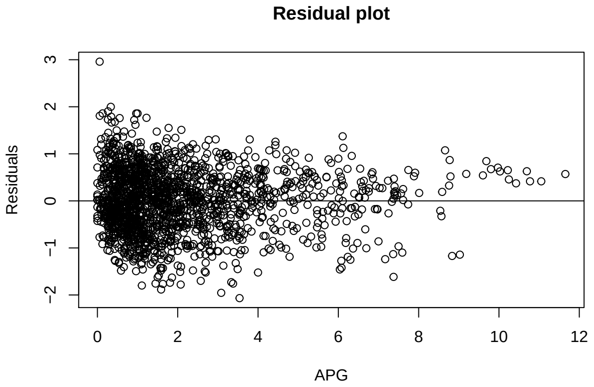
Four Assumptions

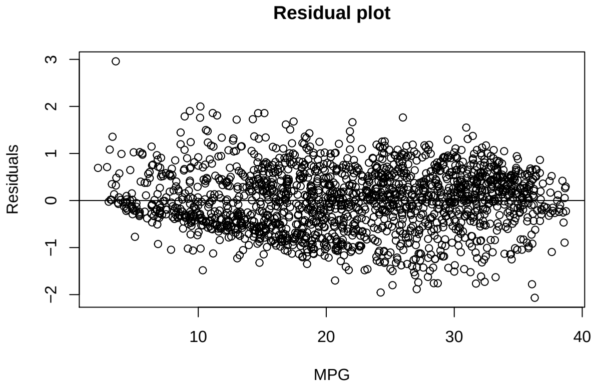
To begin developing our model we first must make sure that our four essential assumptions are satisfied. The first assumption is the linearity between our response and predictor variables, to check this we created a scatter plot for each predictor compared to the response and analyzed the relationships we saw.

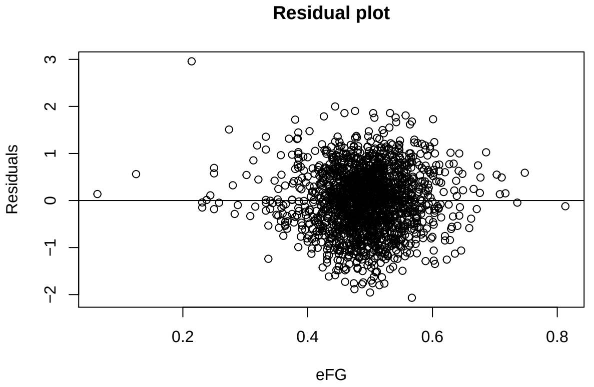






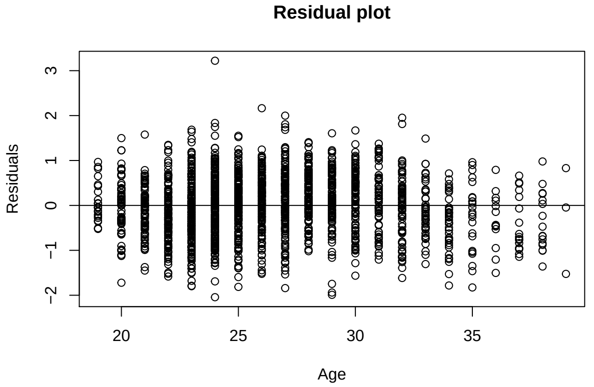


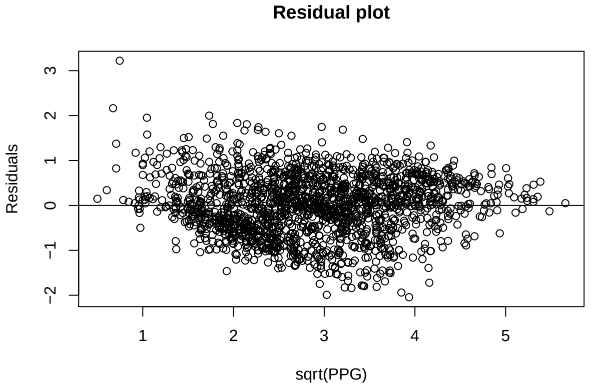


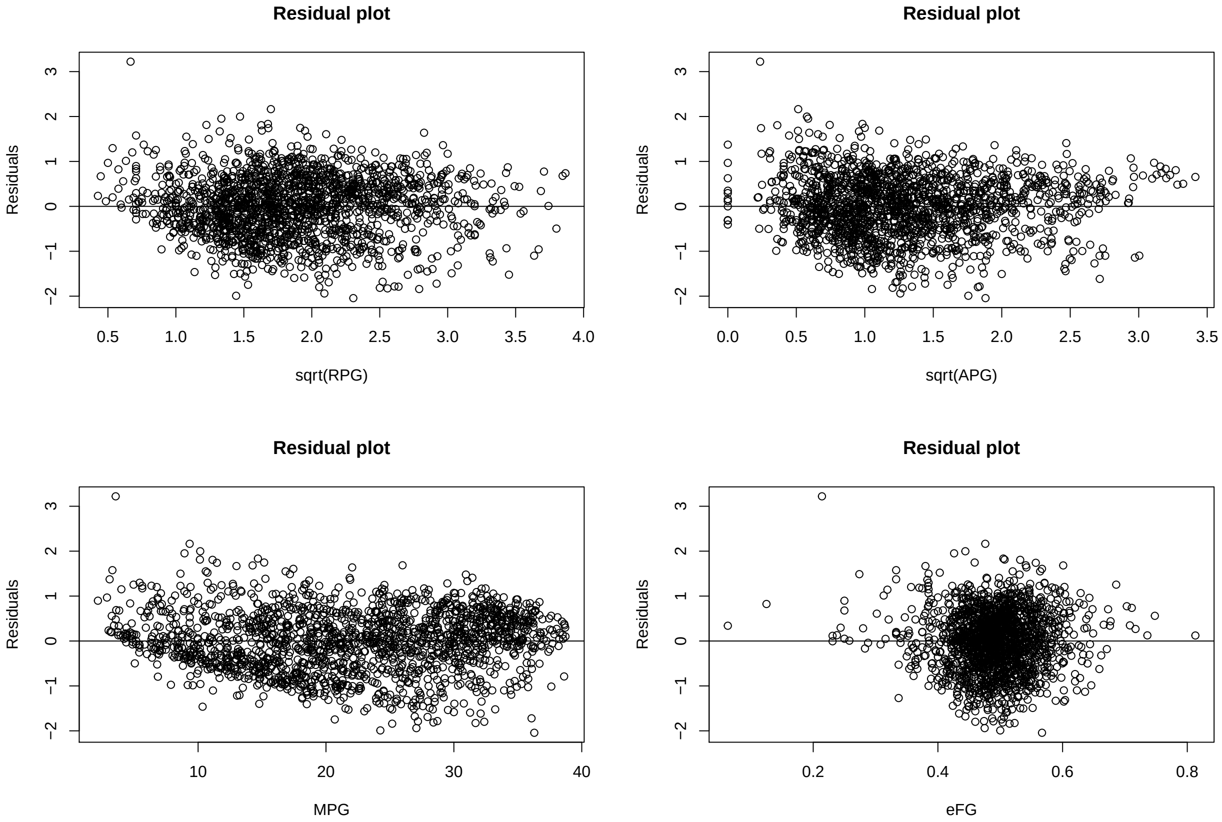


We notice that the residual plots of the covariates, `PPG`, `RPG`, and `APG`, are showing a fanning pattern, which violates the assumption of linearity. We conclude that this may be the result of their values being right skewed. To fix this, we will apply the square root function to each of the covariates, `PPG`, `RPG`, and `APG`. We will then create a new model with these transformations and check the residual plots of each covariate again. Our new model is then,

```
lm(formula = log(Salary) ~ Age + sqrt(PPG) + sqrt(RPG) + sqrt(APG) + MPG + eFG, data =
nba)
```





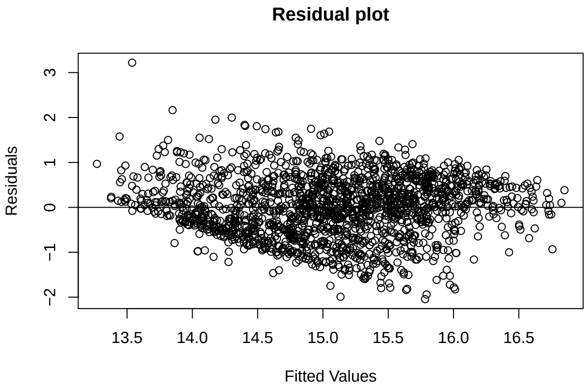


It appears now that all of our covariates, `Age`, `sqrt(PPG)`, `sqrt(RPG)`, `sqrt(APG)`, `MPG`, and `eFG` all satisfy this condition of linearity. That is, there is no clear pattern when plotting their values against the model's residuals. This implies randomness, which in turn suggests a linear relationship between the covariates and response variables.

The slight change in the scatter of residuals we can see in our predictors; `sqrt(PPG)`, `sqrt(RPG)`, and `sqrt(APG)`, can be explained by the fact that relatively few players tend to earn higher scores in these categories. This means we see fewer points on our graph and are thus less likely to see the same amount of residual spread as we do at lower values where the majority of our data lies.

The bizarre looking shape we see in the residual plot of `eFG` is due to the fact that the majority of all players in the NBA have relatively similar `eFG` scores (0.4 - 0.6). Much like our `sqrt(PPG)`, `sqrt(RPG)`, and `sqrt(APG)` plots, the change in randomness we see is due to relatively few players having `eFG` values outside of this cluster and thus we are less likely to see extreme errors at those values.

For the next assumption we are required to show that there is no relationship between our residual and fitted values, that is, that it is independent of errors. To do this we plotted our residuals against our fitted values and looked at what was displayed.

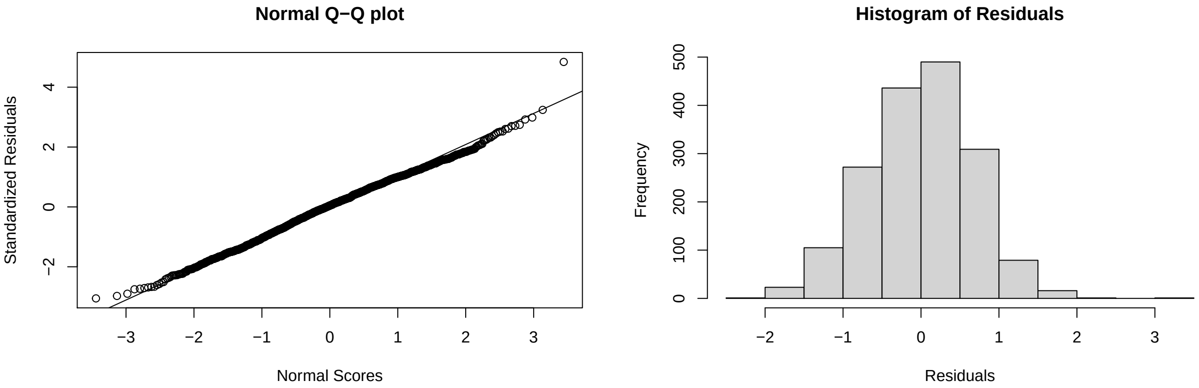


We can see that there is no relation between the residuals and the fitted values of the model. Hence, the variance of the residuals is the same for all the values of the covariates.

The sharp line we see at the lower end of this plot is due to the minimum salary the NBA must pay its players. In turn this baseline means that as salaries get lower, it becomes impossible to underpay these players and still meet the minimum requirement.

This brings us to our third assumption, homoscedasticity, that is, we want to see a constant variance of our error values for all the values of the covariates. Again we plotted our residuals against the covariates and studied the variance we saw at each step. As observed in the scatter plot in assumption 2, the variance of the residuals appeared to be consistent across all values of the x-axis.

Finally, our last assumption requires that our error terms are normally distributed. This simply required us to plot a normal quantile plot of our error terms and compare this to the line of fit.



As seen in the graphs above, our Q-Q plot and histogram of the residuals shows signs of normally distributed error terms. This was accomplished through initially applying a log transformation to our response variable, `Salary`, which in turn affected the error terms.

With our initial assumptions sorted out, we are now ready to move on to developing our model.

Backward Elimination

We first use Backward Elimination in hopes of removing insignificant covariates in our model. We will be using a significance level of $\alpha = 0.05$. Looking at the summary of our current model (Please see **Appendix: Model 2**), we notice that `sqrt(APG)` has a low t-score and p-value $> \alpha$. As a result of Backward Elimination, we are going to drop the `sqrt(APG)` predictor variable from our model. Our new model is then,

```
lm(formula = log(Salary) ~ Age + sqrt(PPG) + sqrt(RPG) + MPG + eFG, data = nba)
```

Correlation Matrix and Interaction Terms

We now check the correlation matrix to possibly remove covariates with high correlation or add interaction terms.

```
##                   Age sqrt(PPG)  sqrt(RPG)        MPG       eFG
## Age       1.00000000 0.0373677 0.02727833 0.07976718 0.1439617
## sqrt(PPG) 0.03736770 1.0000000 0.59352400 0.91842172 0.2661506
## sqrt(RPG) 0.02727833 0.5935240 1.00000000 0.64561051 0.2805158
## MPG       0.07976718 0.9184217 0.64561051 1.00000000 0.2139825
## eFG       0.14396168 0.2661506 0.28051578 0.21398246 1.0000000
```

We notice a high, nearly perfect, correaltion between our covariates `sqrt(PPG)` and `MPG`, with a correlation of 0.91842172. This is no surprise since players who score the most points, tend to be the ones with the most playtime. Looking at the summary of our current model (Please see **Appendix: Model 3**), we notice that the p-value of the covariate `sqrt(PPG)` is much less than that of the p-value of `MPG`. This tells us that `sqrt(PPG)` is much more significant to our model than `MPG` is. Hence we have decided to remove the predictor variable `MPG` from the model.

We also notice a somewhat high correlation between `sqrt(PPG)` and `sqrt(RPG)`. Since there is no empirical indication of a relation between `PPG` and `RPG`, we decided to resolve this multicollinearity by adding an interaction term, `sqrt(PPG*RPG)`. Since the addition of this interaction term may itself increase multicollinearity, we are going to center the data for `sqrt(PPG)` and `sqrt(RPG)`, calling them `sPPG` and `sRPG`, respectively. Then, this will also change the interaction term to `I(sPPG*sRPG)`. Our new model is then,

```
lm(formula = log(Salary) ~ Age + sPPG + sRPG + eFG + I(sPPG*sRPG), data = nba)
```

Criterion Table

With the remaining predictor variables, we want to determine which combination produces the best possible model. To achieve this we will create a criterion table with every combination to compare their Sum of Square Residuals, Coefficient of Determination, R^2_{adj}, and Akaike's Information Criterion.

Based on the results of the criterion table (Please see **Appendix: Criterion Table**), the model that we will select contains `Age`, `sPPG`, `sRPG`, `eFG`, and the interaction term `I(sPPG*sRPG)` as its only predictor variables. Through examining the AIC of each model, we can observe that our current model has the lowest AIC out of all the possible models. Our model also has the smallest SS_{res}. Minimizing the SS_{res} is important as the lower the SS_{res}, the higher the SS_{reg}. This means the model is better fit to the data. Furthermore, the model we selected has the highest R^2_{adj} and the decrease of it compared to the R^2 criterion is small, implying the significance of the covariates in minimizing the SS_{res}. In the end we favored the full model as we deemed having a combination of lower SS_{res} and lower AIC to be the most important factors in selecting our final model. Hence our final model is,

```
lm(formula = log(Salary) ~ Age + sPPG + sRPG + eFG + I(sPPG*sRPG), data = nba)
```

Model Validation

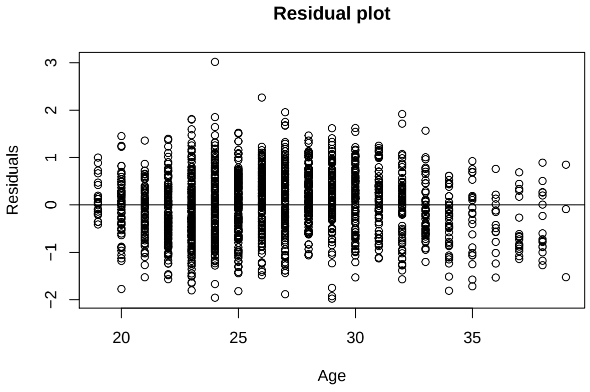
To validate our model, we first obtained an independent dataset. We used the NBA player statistics in the 2017 season, which was from the same source as our original model development dataset. We felt like this was a good selection for our validation dataset as it will test how well the model can predict salary in future years as our model development dataset was taken from 2013-2016 NBA seasons.

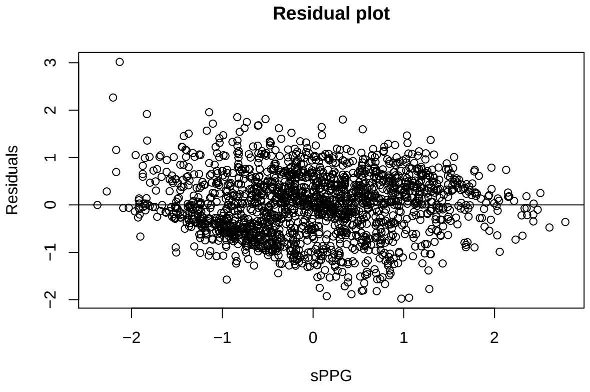
Through calculations in R we found the $MSPE$ to be 0.4656555. Compared to the MS_{res} from our model, which is 0.4479502, we find that the difference is 0.0177053. These results helped further prove the predictive ability of our model as the difference between these 2 values was incredibly small being less than 0.018.

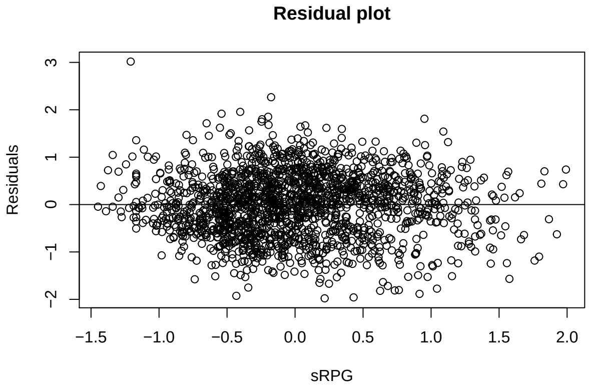
Diagnostics

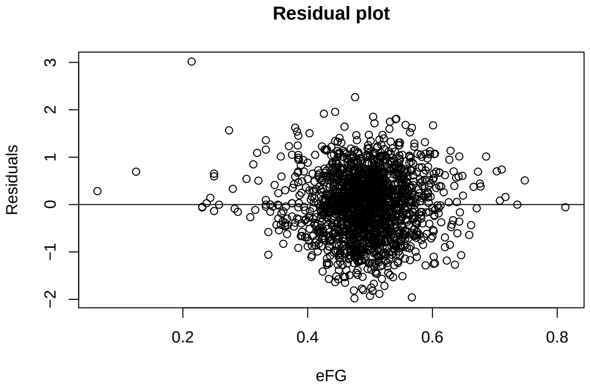
Improper Functional Form

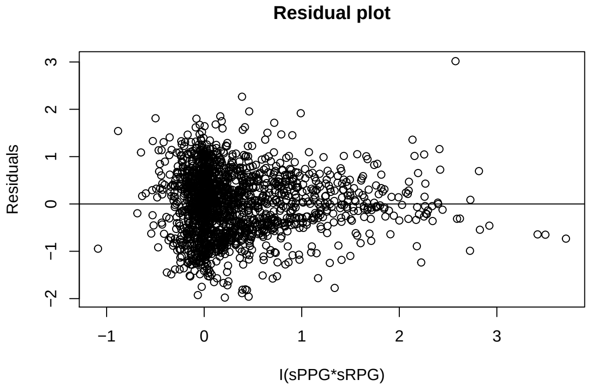
The first thing that we want to check is for improper functional form. This is done through plotting the residuals against the predictor variables and fitted values.

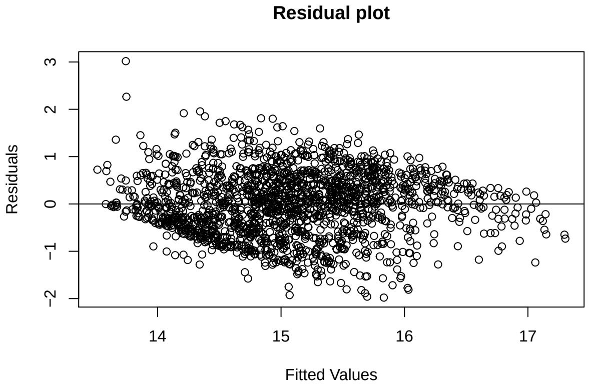






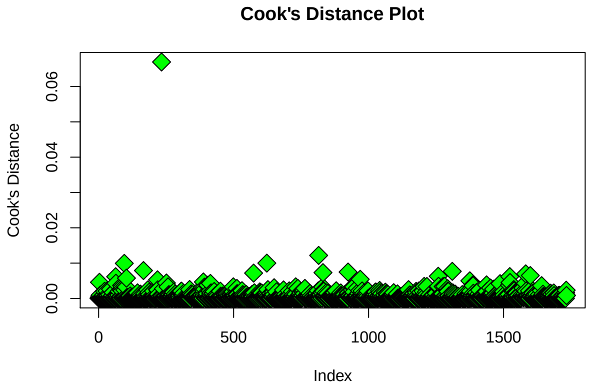






As seen in the Residual plots, there is a random pattern across all predictor variables and fitted values which indicates that the functional form is adequate. This is to say that we do see a subtle fanning effect in the residual plot against the fitted values. This implies that our model has a difficulty predicting values in the higher end. This is expected because we initially saw that the values of `Salary` are right skewed, meaning that there are not many samples on the higher end. This is not problematic because it is difficult to control the variance with a limitation on the data. Even with all the values fixed, it it still hard to control the variance.

Influential Observations

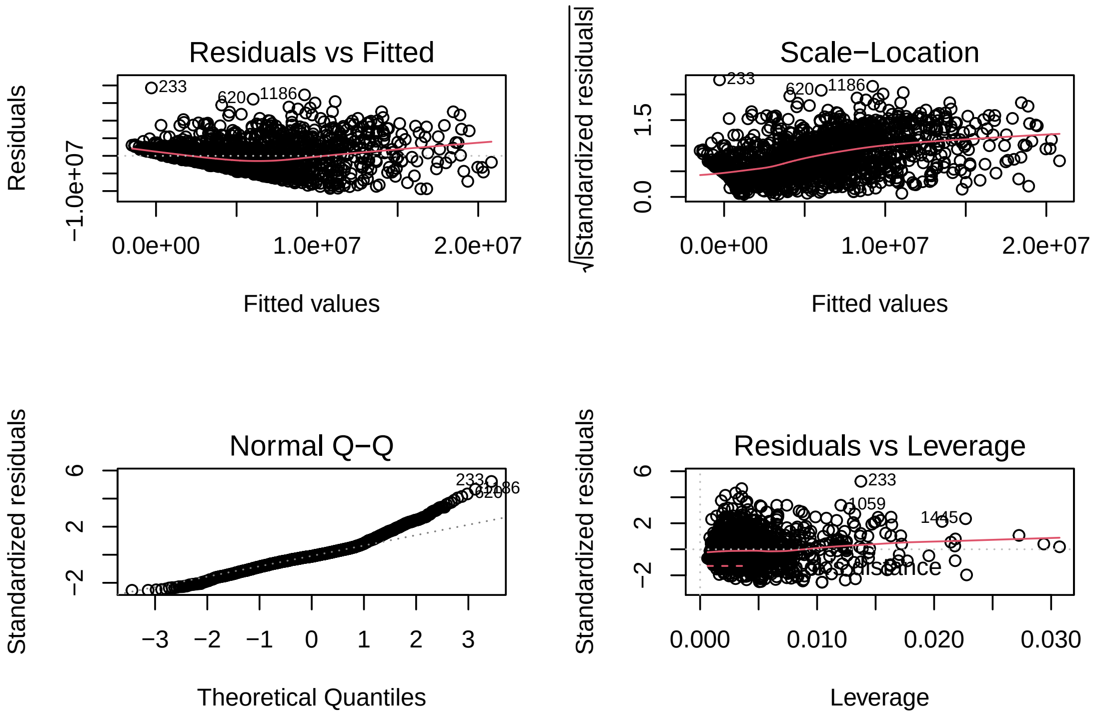


Through running calculations to find any outlying observations for salary, our tests concluded that observation 233 was the only outlying observation for NBA player Salary. The $\mid DFFITS_i \mid$ of this observation was 0.6374419, which is greater than $2\sqrt{\frac{p'}{n}} = 0.117681$, resulting in that observation being influential. Also, the Cook's distance for this observation is 0.0669484 which can be seen in the graph above as being significantly larger than the 20th percentile, further indicating the influence of this observation. We can conclude from this information that there is 1 influential outlying observation. Calculating the $DFBETAS$ for each predictor variable we found that observation 233 did not have an influence on any of our model coefficients as their respective $DFBETA$ values were all less than $\frac{2}{\sqrt{n}} = 0.04804307$. Through looking into the dataset we noticed there was an error in the salary given to the player as in reality he only made about 980K for the 2016 season, but the dataset mistakenly recorded that he made 19 million dollars. However, since our dataset consists of 1733 observations the impact of this singular observation on predicting Salary is overall not significant.

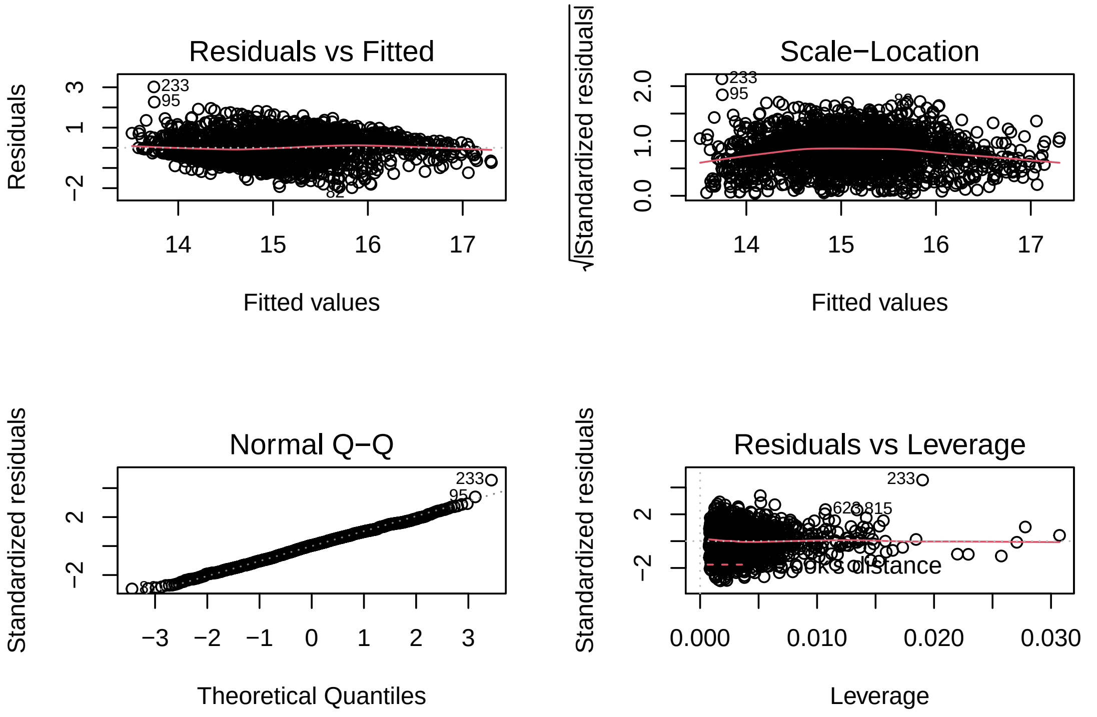
R Diagnostic Plots

We can generate the diagnostic plots for a model with all of the raw variables the diagnostic plots for our final model to see how improved it is.

lm(formula = Salary Age + PPG + RPG + APG + MPG + eFG, data = nba)



Final Model



Model Improvement

As we can see in our final model, all diagnostic plots show significant improvement from the original model we created. Both the Residuals vs Fitted and Scale-Location plots appear more randomly scattered and have lost the linear/curvilinear relationships they had displayed in our original plots. Additionally, in the Normal Q-Q plot, the extreme right skewed distribution we saw previously has been lost and we see almost all points lying along the Q-Q line as required. This provides further confirmation that our final model is significantly better at predicting NBA Salaries than our original model.

Residuals vs Fitted

As we can see here and saw while confirming our assumptions, there is no discernible pattern between the residuals and fitted values from our model. Additionally, there appears to be an equal spreading of our residual values around a nearly horizontal line, thus providing confirmation that our linearity assumption has been satisfied.

Scale-Location

This scatterplot allows us to test for homoscedasticity in our model and it is evident from this plot that the standardized residuals take on a randoming spreading around the central line. The slight curve we see in our central line is likely negligible as the individual points appear to hold a constantly random scatter, thus we can conclude that we have homoscedasticity (constant variance of errors) in our model.

Normal Q-Q

The residuals on the Normal Q-Q plot lie nearly perfectly in line with the line generated by the Q-Q plot with very little left or right skewness. As such, we can confirm that our normality assumption has not been violated and we remain to have normally distributed error terms.

Residuals vs Leverage

It is clear from this plot that no observations we studied ended up outside of the Cook's distance as no points lie in the extreme upper or lower right corners. This suggests that we do not have any influential observations that may interfere with our final model and its prediction capabilities.

Multicollinearity

VIF Table

	final.model	**Age**	**sPPG**	**sRPG**	**eFG**	**I(sPPG*sRPG)**
VIF		1.028772	1.574066	1.600244	1.159510	1.049168
$\overline{\text{VIF}}$	1.282352					

To determine if we had any serious cases of multicollinearity in our model we computed the VIF values for each predictor as well as their mean. We found that all predictors had VIF values less than 10, additionally, the mean of these values, 1.282352, is not significantly larger than 1. Thus we can conclude that there is no indication of serious multicollinearity in our model.

Conclusion

Usefulness

In a highly dynamic sport like basketball, there is a huge variety of factors which make a player seem more valuable to a team. In the NBA, when these factors affect how much money each player can take home, it becomes very difficult to discern which of these factors are the most significant. We developed this model to help get rid of this confusion and pin down exactly what factors put the most money in these players' pockets by the end of the year. By using a player's Age, PPG, RPG, eFG, and an interaction term between PPG and APG, our model can be used to make predictions on what a player's salary should be based off of their in game statistics. As a result of its predictive ability, our model can also allow us to determine whether a player is being under or overpaid in a specific year. Additionally, incoming NBA players can see the factors we selected as most significant to increasing a salary and gear their playstyle toward it, advancing their skills in each of these areas to maximize their earnings.

Final Model Interpretation

Our final linear regression model has the form,

$$\begin{aligned}\log(\widehat{\mathbf{Salary}_i}) = 14.10367708 + 0.04696916\mathbf{Age}_i + 0.57228160\sqrt{\mathbf{PPG}_i} + \\ 0.37321508\sqrt{\mathbf{RPG}_i} - 0.62644642\mathbf{eFG}_i + 0.12300309\sqrt{\mathbf{PPG}_i\mathbf{RPG}_i}\end{aligned}$$

$$\begin{aligned}\implies \widehat{\mathbf{Salary}}_i = \exp(14.10367708 + 0.04696916\mathbf{Age}_i + 0.57228160\sqrt{\mathbf{PPG}_i} + \\ 0.37321508\sqrt{\mathbf{RPG}_i} - 0.62644642\mathbf{eFG}_i + 0.12300309\sqrt{\mathbf{PPG}_i\mathbf{RPG}_i})\end{aligned}$$

$$\begin{aligned}\implies \widehat{\mathbf{Salary}}_i = \exp(14.10367708)\exp(0.04696916\mathbf{Age}_i)\exp(0.57228160\sqrt{\mathbf{PPG}_i}) \\ \exp(0.37321508\sqrt{\mathbf{RPG}_i})\exp(-0.62644642\mathbf{eFG}_i)\exp(0.12300309\sqrt{\mathbf{PPG}_i\mathbf{RPG}_i})\end{aligned}$$

Then, in the context of our data, if our model were additive, the intercept would not have an interpretation, however our model has a multiplicative relationship between the response and predictor variables. This means that the change in the mean response, **Salary**, is always multiplicative by a factor of exp(14.10367708).

As for **Age**, a unit increase in **Age**, will increase the mean response, **Salary**, by a multiplicative factor of exp(0.04696916). This is also similar to **eFG**, where its unit increase will increase the mean response, **Salary**, by a multiplicative factor of exp(−0.62644642).

The difference is in the change in **PPG** and **RPG**. The change in the mean response, **Salary**, with a unit squared increase in **PPG** when **RPG** is held constant is by a multiplicative factor of $\exp(0.57228160)\exp(0.12300309\mathbf{RPG})$. Conversely, the change in the mean response, **Salary**, with a unit squared increase in **RPG** when **PPG** is held constant is by a multiplicative factor of $\exp(0.37321508)\exp(0.12300309\mathbf{PPG})$.

Of all the covariates, we see that the multiplicative increase of the unit squared increase in **PPG** has the greatest effect on the response variable. This is expected as we discussed, **PPG** is directly related to score in which determines the winning of games.

Limitaions

Due to the constant inflation in the world, the salary of NBA players is increasing at a higher rate than all of our predictor variables. In the last 10 years alone, the salary cap each team was allotted to spend on their players has almost doubled from 58 million dollars in 2010 to 115 million dollars in 2020. Due to such a significant increase, this may result in our model returning underpredictions for players' salaries in future years.

An interesting factor that is much too complex for our model to track is the star power of each potential player. Ultimately, the NBA is an entertainment business, as such, players that are more famous are more likely to bring in a greater number of viewers and thus receive higher salaries in turn. A player's fame can be tracked by things like; their number of social media followers, number and scale of outside endorsements, as well as their 'hype' in the highschool scene. Despite this, these factors are extremely complicated and cannot be completely tracked, meaning that they could not be included in our model.

Another area our model cannot account for are the obscure and uncountable skills required by players that allow the teams to take home wins. Things like leadership, court presence, court vision, and instinct all work toward making these players shine above others and help them take in larger salaries. These factors however, have no definitive way to measure them, rendering us unable to add them into our model. The only recommendation that can be made regarding these factors is to work at improving them to the best of each player's ability as they are integral to performing well as a team.

References

Blanco, F. (2018). NBA - Advanced & Basic Season Stats (1950-2017). Retrieved 2020, from https://www.kaggle.com/whitefero/nba-players-advanced-season-stats-19782016.

Basketball Reference. (2020). Glossary. Retrieved August 08, 2020, from https://www.basketball-reference.com//about/glossary.html

Appendix

```r
library(tidyverse)
nba <- readr::read_csv(file="NBA2016.csv")
```

Exploratory Data Analysis

Dependent Variable

```r
hist(nba$Salary, main = "Histogram of Salary", xlab = "Salary")
hist(log(nba$Salary), main = "Histogram of log(Salary)", xlab = "log(Salary)")
```

Independent Variables

```r
hist(nba$Age, main = "Histogram of Age", xlab = "Age")
hist(nba$PPG, main = "Histogram of PPG", xlab = "PPG")
hist(nba$RPG, main = "Histogram of RPG", xlab = "RPG")
hist(nba$APG, main = "Histogram of APG", xlab = "APG")
hist(nba$MPG, main = "Histogram of MPG", xlab = "MPG")
hist(nba$eFG, main = "Histogram of eFG", xlab = "eFG")
```

Model Development

Model 1

```r
model1 <- lm(formula = log(Salary) ~ Age + PPG + RPG + APG + MPG + eFG, data = nba)
summary(model1)
```

```
## 
## Call:
## lm(formula = log(Salary) ~ Age + PPG + RPG + APG + MPG + eFG, 
##     data = nba)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -2.06699 -0.45476  0.01788  0.46492  2.95843 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 12.662505   0.153665  82.403  < 2e-16 ***
## Age          0.044409   0.003920  11.329  < 2e-16 ***
## PPG          0.069607   0.006359  10.947  < 2e-16 ***
## RPG          0.080997   0.009080   8.921  < 2e-16 ***
## APG          0.005026   0.012528   0.401    0.688    
## MPG          0.020697   0.004456   4.645 3.66e-06 ***
## eFG         -0.352026   0.272624  -1.291    0.197    
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.6604 on 1726 degrees of freedom
## Multiple R-squared:  0.5357, Adjusted R-squared:  0.5341
## F-statistic:   332 on 6 and 1726 DF,  p-value: < 2.2e-16
```

Four Assumptions

```
# Assumption 1
model1.res = resid(model1)

plot(nba$Age, model1.res, ylab = "Residuals", xlab= "Age", main = "Residual plot")
abline(0, 0)
plot(nba$PPG, model1.res, ylab = "Residuals", xlab= "PPG", main = "Residual plot")
abline(0, 0)
plot(nba$RPG, model1.res, ylab = "Residuals", xlab= "RPG", main = "Residual plot")
abline(0, 0)
plot(nba$APG, model1.res, ylab = "Residuals", xlab= "APG", main = "Residual plot")
abline(0, 0)
plot(nba$MPG, model1.res, ylab = "Residuals", xlab= "MPG", main = "Residual plot")
abline(0, 0)
plot(nba$eFG, model1.res, ylab = "Residuals", xlab= "eFG", main = "Residual plot")
abline(0, 0)
```

Model 2

```
model2 <- lm(formula = log(Salary) ~ Age + sqrt(PPG) + sqrt(RPG) + sqrt(APG) +
               MPG + eFG, data = nba)
summary(model2)
```

```
##
## Call:
## lm(formula = log(Salary) ~ Age + sqrt(PPG) + sqrt(RPG) + sqrt(APG) +
##     MPG + eFG, data = nba)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0433 -0.4638  0.0295  0.4727  3.2209
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 12.046490   0.160819  74.907  < 2e-16 ***
## Age          0.044079   0.004005  11.007  < 2e-16 ***
## sqrt(PPG)    0.418557   0.046171   9.065  < 2e-16 ***
## sqrt(RPG)    0.319382   0.040738   7.840 7.83e-15 ***
## sqrt(APG)   -0.052216   0.045304  -1.153  0.24924
## MPG          0.022302   0.005605   3.979 7.21e-05 ***
## eFG         -0.734714   0.284322  -2.584  0.00985 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
```

```
## Residual standard error: 0.6699 on 1726 degrees of freedom
## Multiple R-squared:  0.5222, Adjusted R-squared:  0.5206
## F-statistic: 314.4 on 6 and 1726 DF,  p-value: < 2.2e-16
```

```r
model2.res = resid(model2)

plot(nba$Age, model2.res, ylab = "Residuals", xlab= "Age", main = "Residual plot")
abline(0, 0)
plot(sqrt(nba$PPG), model2.res, ylab = "Residuals", xlab= "sqrt(PPG)",
     main = "Residual plot")
abline(0, 0)
plot(sqrt(nba$RPG), model2.res, ylab = "Residuals", xlab= "sqrt(RPG)",
     main = "Residual plot")
abline(0, 0)
plot(sqrt(nba$APG), model2.res, ylab = "Residuals", xlab= "sqrt(APG)",
     main = "Residual plot")
abline(0, 0)
plot(nba$MPG, model2.res, ylab = "Residuals", xlab= "MPG", main = "Residual plot")
abline(0, 0)
plot(nba$eFG, model2.res, ylab = "Residuals", xlab= "eFG", main = "Residual plot")
abline(0, 0)

# Assumptions 2 & 3
model2.fit = fitted(model2)

plot(model2.fit, model2.res,
     ylab="Residuals", xlab="Fitted Values",
     main="Residual plot")
abline(0, 0)

# Assumption 4
model2.stdres = rstandard(model2)

qqnorm(model2.stdres,
       ylab="Standardized Residuals",
       xlab="Normal Scores",
       main="Normal Q-Q plot")
qqline(model2.stdres)

hist(model2.res, main = "Histogram of Residuals", xlab = "Residuals")
```

Backward Elimination

Model 3

```r
model3 <- lm(formula = log(Salary) ~ Age + sqrt(PPG) + sqrt(RPG) + MPG + eFG, data = nba)
summary(model3)
```

```
##
## Call:
## lm(formula = log(Salary) ~ Age + sqrt(PPG) + sqrt(RPG) + MPG +
##     eFG, data = nba)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0412 -0.4613  0.0349  0.4686  3.2527
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 12.004844   0.156722  76.600  < 2e-16 ***
## Age          0.043372   0.003958  10.959  < 2e-16 ***
## sqrt(PPG)    0.411972   0.045820   8.991  < 2e-16 ***
## sqrt(RPG)    0.340514   0.036383   9.359  < 2e-16 ***
## MPG          0.019371   0.004995   3.878 0.000109 ***
## eFG         -0.660820   0.277026  -2.385 0.017167 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.67 on 1727 degrees of freedom
## Multiple R-squared:  0.5219, Adjusted R-squared:  0.5205
## F-statistic:   377 on 5 and 1727 DF,  p-value: < 2.2e-16
```

Correlation Matrix and Interaction Terms

```r
covaraiates <- cbind(nba$Age, sqrt(nba$PPG), sqrt(nba$RPG), nba$MPG, nba$eFG)
colnames(covaraiates) <- c("Age", "sqrt(PPG)", "sqrt(RPG)", "MPG", "eFG")
cor(covaraiates)
```

Model 4

```r
sPPG = sqrt(nba$PPG) - mean(sqrt(nba$PPG))
sRPG = sqrt(nba$RPG) - mean(sqrt(nba$RPG))
model4 <- lm(formula = log(Salary) ~ Age + sPPG + sRPG + eFG + I(sPPG*sRPG), data = nba)
summary(model4)
```

```
##
## Call:
## lm(formula = log(Salary) ~ Age + sPPG + sRPG + eFG + I(sPPG *
##     sRPG), data = nba)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.97849 -0.46860  0.02916  0.46324  3.01724
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    14.103677   0.164189  85.899  < 2e-16 ***
## Age             0.046969   0.003928  11.957  < 2e-16 ***
## sPPG            0.572282   0.022174  25.809  < 2e-16 ***
## sRPG            0.373215   0.034379  10.856  < 2e-16 ***
## eFG            -0.626446   0.277337  -2.259    0.024 *
## I(sPPG * sRPG)  0.123003   0.028527   4.312 1.71e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6693 on 1727 degrees of freedom
## Multiple R-squared:  0.5228, Adjusted R-squared:  0.5215
## F-statistic: 378.5 on 5 and 1727 DF,  p-value: < 2.2e-16
```

Criterion Table

Criteria for all the possible models

Independent Variable	SS_{res}	R^2	R^2_{adj}	AIC
None	395704.7	0	0	9411.601
Age	1546.904	0.04586114	0.04530994	-192.8662
sPPG	898.4849	0.4458096	0.4454894	-1134.407
sRPG	1140.523	0.296519	0.2961126	-721.0298
eFG	1566.838	0.0335657	0.03300739	-170.6766
I(sPPG*sRPG)	1617.776	0.002147151	0.00157069	-115.2336
Age, sPPG	840.3669	0.4816571	0.4810578	-1248.295
Age, sRPG	1076.079	0.3362683	0.335501	-819.8261
Age, eFG	1508.462	0.06957214	0.0684965	-234.4765
Age, I(sPPG*sRPG)	1538.97	0.05075491	0.04965751	-199.7776
sPPG, sRPG	843.4773	0.4797386	0.4791371	-1241.892
sPPG, eFG	898.4321	0.4458422	0.4452015	-1132.509
sPPG, I(sPPG*sRPG)	890.1314	0.4509621	0.4503274	-1148.594
sRPG, eFG	1138.891	0.297526	0.297526	-721.5121
sRPG, I(sPPG*sRPG)	1139.373	0.2972288	0.2964163	-720.7791
eFG, I(sPPG*sRPG)	1556.687	0.03982709	0.03871706	-179.941
Age, sPPG, sRPG	786.0719	0.5151466	0.5143053	-1362.042
Age, sPPG, eFG	839.6632	0.4820912	0.4811926	-1247.747
Age, sPPG, I(sPPG*sRPG)	826.468	0.49023	0.4893455	-1275.197
Age, sRPG, eFG	1076.06	0.3362802	0.3351286	-817.857
Age, sRPG, I(sPPG*sRPG)	1072.225	0.3386457	0.3374982	-824.0444
Age, eFG, I(sPPG*sRPG)	1493.601	0.07873857	0.07714009	-249.6344
sPPG, sRPG, eFG	842.5678	0.4802996	0.4793978	-1241.762
sPPG, sRPG, I(sPPG*sRPG)	837.9335	0.483158	0.4822612	-1251.32
sPPG, eFG, I(sPPG*sRPG)	889.602	0.4512886	0.4503365	-1147.625
sRPG, eFG, I(sPPG*sRPG)	1137.094	0.2986344	0.2974175	-722.2488
Age, sPPG, sRPG, eFG	781.9366	0.5176972	0.5165808	-1369.183
Age, sPPG, sRPG, I(sPPG*sRPG)	775.8941	0.5214243	0.5203165	-1382.627
Age, sPPG, eFG, I(sPPG*sRPG)	826.4003	0.4902718	0.4890919	-1273.339
Age, sRPG, eFG, I(sPPG*sRPG)	1071.98	0.3387971	0.3372666	-822.4413
sPPG, sRPG, eFG, I(sPPG*sRPG)	837.6505	0.4833326	0.4821366	-1249.905
Age, sPPG, sRPG, eFG, I(sPPG*sRPG)	773.6086	0.522834	0.5214525	-1385.739

```r
combo0 = lm(log(Salary)~0, data = nba)
combo1 = lm(log(Salary)~sPPG, data = nba)
combo2 = lm(log(Salary)~sRPG, data = nba)
combo3 = lm(log(Salary)~I(sPPG*sRPG), data = nba)
combo4 = lm(log(Salary)~ Age, data = nba)
combo5 = lm(log(Salary)~ eFG, data = nba)

combo12 = lm(log(Salary)~sPPG+ sRPG, data = nba)
```

```
combo13 = lm(log(Salary)~sPPG+ I(sPPG*sRPG), data = nba)
combo14 = lm(log(Salary)~sPPG+ Age, data = nba)
combo15 = lm(log(Salary)~sPPG+ eFG, data = nba)
combo23 = lm(log(Salary)~sRPG+ I(sPPG*sRPG), data = nba)
combo24 = lm(log(Salary)~sRPG+ Age, data = nba)
combo25 = lm(log(Salary)~sRPG+ eFG, data = nba)
combo34 = lm(log(Salary)~I(sPPG*sRPG) + Age, data = nba)
combo35 = lm(log(Salary)~I(sPPG*sRPG) + eFG, data = nba)
combo45 = lm(log(Salary)~Age+ eFG, data = nba)

combo123 = lm(log(Salary)~sPPG+ sRPG + I(sPPG*sRPG), data = nba)
combo124 = lm(log(Salary)~sPPG+ sRPG + Age, data = nba)
combo125 = lm(log(Salary)~sPPG+ sRPG + eFG, data = nba)
combo134 = lm(log(Salary)~sPPG+ I(sPPG*sRPG) + Age, data = nba)
combo135 = lm(log(Salary)~sPPG+ I(sPPG*sRPG) + eFG, data = nba)
combo145 = lm(log(Salary)~sPPG+ Age + eFG, data = nba)
combo234 = lm(log(Salary)~sRPG + I(sPPG*sRPG)+ Age, data = nba)
combo235 = lm(log(Salary)~sRPG + I(sPPG*sRPG)+ eFG, data = nba)
combo245 = lm(log(Salary)~sRPG+ Age + eFG, data = nba)
combo345 = lm(log(Salary)~I(sPPG*sRPG)+ Age + eFG, data = nba)


combo1234 = lm(log(Salary)~sPPG+ sRPG + I(sPPG*sRPG) + Age, data = nba)
combo1235 = lm(log(Salary)~sPPG+ sRPG + I(sPPG*sRPG) + eFG, data = nba)
combo1245 = lm(log(Salary)~sPPG+ sRPG + Age + eFG, data = nba)
combo1345 = lm(log(Salary)~sPPG + I(sPPG*sRPG) + Age + eFG, data = nba)
combo2345 = lm(log(Salary)~sRPG + I(sPPG*sRPG) + Age +eFG, data = nba)

combo12345 = lm(log(Salary)~sPPG+ sRPG + I(sPPG*sRPG)+ Age + eFG, data = nba)

#combo0
s <- summary(combo0)$sigma
SSres <- sum(combo0$residuals^2)
Rsq <- summary(combo0)$r.squared
Rsq_adj <- summary(combo0)$adj.r.squared
p_prime <- length(combo0$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo1
s <- summary(combo1)$sigma
SSres <- sum(combo1$residuals^2)
Rsq <- summary(combo1)$r.squared
Rsq_adj <- summary(combo1)$adj.r.squared
p_prime <- length(combo1$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo2
s <- summary(combo2)$sigma
SSres <- sum(combo2$residuals^2)
Rsq <- summary(combo2)$r.squared
Rsq_adj <- summary(combo2)$adj.r.squared
p_prime <- length(combo2$coefficients)
n <- length(log(nba$Salary))
```

```r
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo3
s <- summary(combo3)$sigma
SSres <- sum(combo3$residuals^2)
Rsq <- summary(combo3)$r.squared
Rsq_adj <- summary(combo3)$adj.r.squared
p_prime <- length(combo3$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo4
s <- summary(combo4)$sigma
SSres <- sum(combo4$residuals^2)
Rsq <- summary(combo4)$r.squared
Rsq_adj <- summary(combo4)$adj.r.squared
p_prime <- length(combo4$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo5
s <- summary(combo5)$sigma
SSres <- sum(combo5$residuals^2)
Rsq <- summary(combo5)$r.squared
Rsq_adj <- summary(combo5)$adj.r.squared
p_prime <- length(combo5$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime

#combo12
s <- summary(combo12)$sigma
SSres <- sum(combo12$residuals^2)
Rsq <- summary(combo12)$r.squared
Rsq_adj <- summary(combo12)$adj.r.squared
p_prime <- length(combo12$coefficients)
n <- length(log(nba$Salary))
C <- SSres/s^2 + 2*p_prime - n
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo13
s <- summary(combo13)$sigma
SSres <- sum(combo13$residuals^2)
Rsq <- summary(combo13)$r.squared
Rsq_adj <- summary(combo13)$adj.r.squared
p_prime <- length(combo13$coefficients)
n <- length(log(nba$Salary))
C <- SSres/s^2 + 2*p_prime - n
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo14
s <- summary(combo14)$sigma
SSres <- sum(combo14$residuals^2)
Rsq <- summary(combo14)$r.squared
Rsq_adj <- summary(combo14)$adj.r.squared
p_prime <- length(combo14$coefficients)
n <- length(log(nba$Salary))
C <- SSres/s^2 + 2*p_prime - n
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
```

```r
#combo15
s <- summary(combo15)$sigma
SSres <- sum(combo15$residuals^2)
Rsq <- summary(combo15)$r.squared
Rsq_adj <- summary(combo15)$adj.r.squared
p_prime <- length(combo15$coefficients)
C <- SSres/s^2 + 2*p_prime - n
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo23
s <- summary(combo23)$sigma
SSres <- sum(combo23$residuals^2)
Rsq <- summary(combo23)$r.squared
Rsq_adj <- summary(combo23)$adj.r.squared
p_prime <- length(combo23$coefficients)
C <- SSres/s^2 + 2*p_prime - n
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo24
s <- summary(combo24)$sigma
SSres <- sum(combo24$residuals^2)
Rsq <- summary(combo24)$r.squared
Rsq_adj <- summary(combo24)$adj.r.squared
p_prime <- length(combo24$coefficients)
C <- SSres/s^2 + 2*p_prime - n
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo25
s <- summary(combo25)$sigma
SSres <- sum(combo25$residuals^2)
Rsq <- summary(combo25)$r.squared
Rsq_adj <- summary(combo25)$adj.r.squared
p_prime <- length(combo25$coefficients)
C <- SSres/s^2 + 2*p_prime - n
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo34
s <- summary(combo34)$sigma
SSres <- sum(combo34$residuals^2)
Rsq <- summary(combo34)$r.squared
Rsq_adj <- summary(combo34)$adj.r.squared
p_prime <- length(combo34$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo35
s <- summary(combo35)$sigma
SSres <- sum(combo35$residuals^2)
Rsq <- summary(combo35)$r.squared
Rsq_adj <- summary(combo35)$adj.r.squared
p_prime <- length(combo35$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo45
s <- summary(combo45)$sigma
SSres <- sum(combo45$residuals^2)
Rsq <- summary(combo45)$r.squared
Rsq_adj <- summary(combo45)$adj.r.squared
p_prime <- length(combo45$coefficients)
C <- SSres/s^2 + 2*p_prime - n
```

```
AIC <- n*log(SSres) - n*log(n) + 2*p_prime

#combo123
s <- summary(combo123)$sigma
SSres <- sum(combo123$residuals^2)
Rsq <- summary(combo123)$r.squared
Rsq_adj <- summary(combo123)$adj.r.squared
p_prime <- length(combo123$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo124
s <- summary(combo124)$sigma
SSres <- sum(combo124$residuals^2)
Rsq <- summary(combo124)$r.squared
Rsq_adj <- summary(combo124)$adj.r.squared
p_prime <- length(combo124$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo125
s <- summary(combo125)$sigma
SSres <- sum(combo125$residuals^2)
Rsq <- summary(combo125)$r.squared
Rsq_adj <- summary(combo125)$adj.r.squared
p_prime <- length(combo125$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo134
s <- summary(combo134)$sigma
SSres <- sum(combo134$residuals^2)
Rsq <- summary(combo134)$r.squared
Rsq_adj <- summary(combo134)$adj.r.squared
p_prime <- length(combo134$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo135
s <- summary(combo135)$sigma
SSres <- sum(combo135$residuals^2)
Rsq <- summary(combo135)$r.squared
Rsq_adj <- summary(combo135)$adj.r.squared
p_prime <- length(combo135$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo145
s <- summary(combo145)$sigma
SSres <- sum(combo145$residuals^2)
Rsq <- summary(combo145)$r.squared
Rsq_adj <- summary(combo145)$adj.r.squared
p_prime <- length(combo145$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo234
s <- summary(combo234)$sigma
SSres <- sum(combo234$residuals^2)
Rsq <- summary(combo234)$r.squared
Rsq_adj <- summary(combo234)$adj.r.squared
p_prime <- length(combo234$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo235
s <- summary(combo235)$sigma
```

```
SSres <- sum(combo235$residuals^2)
Rsq <- summary(combo235)$r.squared
Rsq_adj <- summary(combo235)$adj.r.squared
p_prime <- length(combo235$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo245
s <- summary(combo245)$sigma
SSres <- sum(combo245$residuals^2)
Rsq <- summary(combo245)$r.squared
Rsq_adj <- summary(combo245)$adj.r.squared
p_prime <- length(combo245$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo345
s <- summary(combo345)$sigma
SSres <- sum(combo345$residuals^2)
Rsq <- summary(combo345)$r.squared
Rsq_adj <- summary(combo345)$adj.r.squared
p_prime <- length(combo345$coefficients)
AIC <- n*log(SSres) - n*log(n) + 2*p_prime

#combo1234
s <- summary(combo1234)$sigma
SSres <- sum(combo1234$residuals^2)
Rsq <- summary(combo1234)$r.squared
Rsq_adj <- summary(combo1234)$adj.r.squared
p_prime <- length(combo1234$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo1235
s <- summary(combo1235)$sigma
SSres <- sum(combo1235$residuals^2)
Rsq <- summary(combo1235)$r.squared
Rsq_adj <- summary(combo1235)$adj.r.squared
p_prime <- length(combo1235$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo1245
s <- summary(combo1245)$sigma
SSres <- sum(combo1245$residuals^2)
Rsq <- summary(combo1245)$r.squared
Rsq_adj <- summary(combo1245)$adj.r.squared
p_prime <- length(combo1245$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo1345
s <- summary(combo1345)$sigma
SSres <- sum(combo1345$residuals^2)
Rsq <- summary(combo1345)$r.squared
Rsq_adj <- summary(combo1345)$adj.r.squared
p_prime <- length(combo1345$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
#combo2345
```

```r
s <- summary(combo2345)$sigma
SSres <- sum(combo2345$residuals^2)
Rsq <- summary(combo2345)$r.squared
Rsq_adj <- summary(combo2345)$adj.r.squared
p_prime <- length(combo2345$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime

#combo12345
s <- summary(combo12345)$sigma
SSres <- sum(combo12345$residuals^2)
Rsq <- summary(combo12345)$r.squared
Rsq_adj <- summary(combo12345)$adj.r.squared
p_prime <- length(combo12345$coefficients)
n <- length(log(nba$Salary))
AIC <- n*log(SSres) - n*log(n) + 2*p_prime
```

Final Model

```r
final.model <- model4
summary(final.model)
```

```
##
## Call:
## lm(formula = log(Salary) ~ Age + sPPG + sRPG + eFG + I(sPPG *
##     sRPG), data = nba)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.97849 -0.46860  0.02916  0.46324  3.01724
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     14.103677   0.164189  85.899  < 2e-16 ***
## Age              0.046969   0.003928  11.957  < 2e-16 ***
## sPPG             0.572282   0.022174  25.809  < 2e-16 ***
## sRPG             0.373215   0.034379  10.856  < 2e-16 ***
## eFG             -0.626446   0.277337  -2.259    0.024 *
## I(sPPG * sRPG)  0.123003   0.028527   4.312 1.71e-05 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.6693 on 1727 degrees of freedom
## Multiple R-squared:  0.5228, Adjusted R-squared:  0.5215
## F-statistic: 378.5 on 5 and 1727 DF,  p-value: < 2.2e-16
```

Model Validation

```r
nba2 <- readr::read_csv(file="NBA2017.csv")
```

```
psPPG = sqrt(nba2$PPG) - mean(sqrt(nba2$PPG))
psRPG = sqrt(nba2$RPG) - mean(sqrt(nba2$RPG))
modelp = lm(formula = log(Salary) ~ Age + psPPG + psRPG + eFG + I(psPPG*psRPG), data = nba2)
mspe = sum(resid(modelp)^2)/length(log(nba2$Salary))
```

Diagnostics

```
# Improper Functional Form Check
final.model.res = resid(final.model)

plot(nba$Age, final.model.res, ylab = "Residuals", xlab= "Age",
     main = "Residual plot")
abline(0, 0)
plot(sPPG, final.model.res, ylab = "Residuals", xlab= "sPPG",
     = "Residual plot")
abline(0, 0)
plot(sRPG, final.model.res, ylab = "Residuals", xlab= "sRPG",
     main = "Residual plot")
abline(0, 0)
plot(nba$eFG, final.model.res, ylab = "Residuals", xlab= "eFG",
     main = "Residual plot")
abline(0, 0)
plot(I(sPPG*sRPG), final.model.res, ylab = "Residuals", xlab= "I(sPPG*sRPG)",
     main = "Residual plot")
abline(0, 0)

final.model.fit = fitted(final.model)

plot(final.model.fit, final.model.res,
     ylab="Residuals", xlab="Fitted Values",
     main="Residual plot")
abline(0, 0)

# Statistical test for outliers
t <- rstudent(final.model)
Pii <- hatvalues(final.model)

n <- length(log(nba$Salary))
alpha <- 0.05
p_prime = length(coef(final.model))
t_crit <- qt(1-alpha/(2*n),n-p_prime-1)
which(abs(t) > t_crit)

# DFFITS
dffits <- dffits(final.model)[233]
cooks <- cooks.distance(final.model)[233]

# Cooks distance Di
plot(cooks.distance(final.model), pch=23, bg='green', cex=2, ylab="Cook's Distance",
     main = "Cook's Distance Plot")

# Diagnostic Plots
```

```r
layout(matrix(c(1,2,3,4),2,2))
plot(lm(formula = Salary ~ Age + PPG + RPG + APG + MPG + eFG, data = nba))

layout(matrix(c(1,2,3,4),2,2))
plot(final.model)

# Multicollinearity
library(car)
vif <- vif(final.model)
meanvif <- mean(vif)
```

Project Contribution

We are satisfied with each group members' contributions to the final project.