DATA603 L02 - NBA Player Point Per Game Modeling

Tomasz Szymczyk, Daniel Zhou

December 7, 2021

Chapter 1: Introduction

We are doing our study on modeling the National Basketball Association (NBA) player stats.

A little background info on the NBA, it's a basketball league that has seasons running from October of a year to April of the next year, with each team playing 82 games. Teams are made up of many players. Each player can score points for their team, and at the end of a game, the team with the most points win the game.[NBA, 2021]

The objective of our study is to create a Multiple Linear Regression model for modeling points per game of an NBA player in a particular season where the player played 10 or more regular season games based on several independent variables, we will describe those variables in more detail in the Methodology section below.

Our research questions are:

1. Do player attributes during a particular season affect a player's points per game stat for that season? Specifically, their age, height, and weight at the start of the season
2. Do other box score statistics from the season affect a player's points per game stat for that season? Specifically, total games played in a season, assists per game, rebounds per game, and net rating per game
3. Does which round the player get drafted in the NBA draft affect the player's points per game stat?

We expect to find that at least some of the variables will affect the players points per game stat.

Topic Importance

This topic is important to our group because we recognize that sports analytics is an important area of application for statistical analysis and modeling. We want learn and gain some experience in statistical analysis and model building in sports analytics.

Data sources

The data is sourced from Kaggle.com, it is an open dataset related to basketball data with public license [Kaggle, 2021] and contains the per season stats per NBA player from 1996 season all the way up to 2021 season. The specific columns of the dataset that we will be using will be described in more detail in the Methodology section below. We plan to use the data from the 10 seasons that is between 2010-2020 seasons for our model building. Our population and sample are defined as per follows:

- we define our population as NBA players in any season that plays 10 or more games in a season

- we define our sample as NBA players from the 10 seasons between 2010-2020 who have played 10 or more games in a season (sample size of 4,473 after some data wrangling)

Please note that during our project checkpoint, we originally proposed to do our project on an Life Expectancy data set. However, we had to switch our topic / dataset to the current one after we found that the Life Expectancy data set was of terrible quality (lots of missing data and more importantly, lots of erroneous data).

Chapter 2: Methodology

Descriptive Statistics

Table 1: Descriptive Statistics of Research Variables

Variable Category	Variable Name	Variable Description / Measurement	Variable Type
Response	pts	Average number of points scored per game	Quantitative
Predictor	age	Age of the player at season start	Quantitative
Predictor	height	Height of the player (cm)	Quantitative
Predictor	weight	Weight of the player (kg)	Quantitative
Predictor	d_round	The draft round the player was picked (factor with 5 levels • Undrafted • 0 • 1st round draft pick • 2nd round draft pick • 4th round draft pick	Qualitative
Predictor	gp	Games played throughout the season	Quantitative
Predictor	reb	Average number of rebounds grabbed per game	Quantitative
Predictor	ast	Average number of assists distributed	Quantitative
Predictor	net_rating	Team's point differential per 100 possessions while the player is on the court	Quantitative

Project Plan

Project plan and all methods for modelling:

1) Create best first order model

   a. Create a full model with response variable and all predictor variables, conduct individual t-test to identify significant predictors
   b. Complete All-Possible-Regressions Selection Procedure [ols_step_best() function] and interpret Cp, AIC, Adjusted R2, and R2 numerical results and plots to identify the best possible predictors.
   c. Calculate Stepwise model to identify significant variables (pent = 0.05, prem = 0.1)
   d. Calculate Forward model to identify significant variables (pent = 0.05)
   e. Calculate Backward model to identify significant variables (prem = 0.05)
   f. Compare all models, identify the best predictors based on highest R2adj and lowest RMSE and comparing All-Possible-Regression predictors.

2) Create the reduced first order model and complete Partial F test to validate that dropping the selected predictors was correct.

3) Test for multicollinearity across the selected predictors in the reduced first order model, remove variables with high VIF if present.

4) Check for interactions across variables and identify the interaction model with only significant interactions remaining. Complete Partial F test to verify dropping all the selected interactions simultaneously is correct.

5) Use ggplot to check correlation and identify potential variables that benefit from being higher order (correlations ~0.5 or greater)

6) Create a higher order model on the high correlation variables using the poly() function, leaving only significant higher order terms

7) Remove insignificant interactions from the model, verify dropping them simultaneously is correct using Partial F test

8) Complete model diagnostics

    a. Linearity assumption check (Residual vs. Fitted plot)
    b. Equal Variance assumption check
        i. Residual vs Fitted Values plot
        ii. Scale-Location Plot: Standardized Residual vs Fitted Values
        iii. Breusch-Pagan test
    c. Normality assumption check
        i. Residual vs Fitted Values plot
        ii. Interpret Histogram of Residuals
        iii. Shapiro-Wilk test
    d. Identify Outliers (and if they are significant)
        i. Residual vs. Leverage plot
        ii. Cook's Distance plot
        iii. Leverage points at 2p/n & 3p/n, check the model with influential points removed to see if it makes it better by comparing R2adj and RMSE. Check data points manually to see if they look like typo's or errors.

9) If the model fails Equal Variance or Normality assumptions, complete Box-Cox transformation

    a. Redo model assumption tests to see if the model meets assumptions
    b. If there are insignificant interactions after transform, drop them and verify.

10) State best final model and interpret

11) Test prediction of the model by supplying test data based on real NBA player scores!

Chapter 3: Main Results of the Analysis

Data Wrangling

```r
# import data
bb_raw=read.csv("all_seasons.csv",header=TRUE)
# shorten the column names
colnames(bb_raw)<- c("X", "player_name","team_abbrev","age","height",
                     "weight","college","country","d_year","d_round",
                     "d_number","gp","pts","reb","ast","net_rating","oreb_pct",
                     "dreb_pct","usg_pct","ts_pct","ast_pct","season")
# shorten the "Undrafted" values under the draft related columns to "U"
bb_raw["d_year"][bb_raw["d_year"] == "Undrafted"] <- "U"
bb_raw["d_round"][bb_raw["d_round"] == "Undrafted"] <- "U"
bb_raw["d_number"][bb_raw["d_number"] == "Undrafted"] <- "U"
head(bb_raw)
```

```
##   X     player_name team_abbrev age height   weight
## 1 0   Travis Knight         LAL  22 213.36 106.5941
## 2 1       Matt Fish         MIA  27 210.82 106.5941
## 3 2    Matt Bullard         HOU  30 208.28 106.5941
## 4 3    Marty Conlon         BOS  29 210.82 111.1300
## 5 4 Martin Muursepp         DAL  22 205.74 106.5941
## 6 5    Martin Lewis         TOR  22 198.12 102.0582
##                          college country d_year d_round d_number gp pts reb
## 1                    Connecticut     USA   1996       1       29 71 4.8 4.5
## 2       North Carolina-Wilmington     USA   1992       2       50  6 0.3 0.8
## 3                           Iowa     USA      U       U        U 71 4.5 1.6
## 4                     Providence     USA      U       U        U 74 7.8 4.4
## 5                           None     USA   1996       1       25 42 3.7 1.6
## 6 Seward County Community College     USA   1995       2       50  9 1.6 0.7
##   ast net_rating oreb_pct dreb_pct usg_pct ts_pct ast_pct  season
## 1 0.5        6.2    0.127    0.182   0.142  0.536   0.052 1996-97
## 2 0.0      -15.1    0.143    0.267   0.265  0.333   0.000 1996-97
## 3 0.9        0.9    0.016    0.115   0.151  0.535   0.099 1996-97
## 4 1.4       -9.0    0.083    0.152   0.167  0.542   0.101 1996-97
## 5 0.5      -14.5    0.109    0.118   0.233  0.482   0.114 1996-97
## 6 0.4       -3.5    0.087    0.045   0.135  0.470   0.125 1996-97
```

```r
colnames(bb_raw)
```

```
##  [1] "X"           "player_name" "team_abbrev" "age"         "height"
##  [6] "weight"      "college"     "country"     "d_year"      "d_round"
## [11] "d_number"    "gp"          "pts"         "reb"         "ast"
## [16] "net_rating"  "oreb_pct"    "dreb_pct"    "usg_pct"     "ts_pct"
## [21] "ast_pct"     "season"
```

```r
# remove null, filter to only include seasons the 10 seasons between 2010-2020
# this is to limit our analysis to recent seasons and limit
nrow(bb_raw)
```

```
## [1] 11700
```

```r
bb <- na.omit(bb_raw)
# filter out players who have played less than 10 games in a particular season
bb = filter(bb, gp >= 10)
nrow(bb)
```

```
## [1] 10685
```

```
# we define our population as players in any season that plays 10 or more games in a season
# we define our sample as players from the 10 seasons between 2010 and 2020
# who have played 10 or more games in a season
bb = filter(bb, (season=='2010-11'|season=='2011-12'|season=='2012-13'
                 |season=='2013-14'|season=='2014-15'|season=='2015-16'
                 |season=='2016-17'|season=='2017-18'|season=='2018-19'
                 |season=='2019-20'))
nrow(bb)
```

```
## [1] 4473
```

First Order Model

```
fullmodel<-lm(pts~age+height+weight+factor(d_round)+gp+reb+ast+net_rating, data = bb)
summary(fullmodel)
```

```
##
## Call:
## lm(formula = pts ~ age + height + weight + factor(d_round) +
##     gp + reb + ast + net_rating, data = bb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.0784  -1.9670  -0.3645   1.5798  14.8123
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        4.378167   3.812950   1.148 0.250933
## age               -0.056669   0.012113  -4.678 2.98e-06 ***
## height            -0.041518   0.010978  -3.782 0.000158 ***
## weight            -0.014393   0.007848  -1.834 0.066741 .
## factor(d_round)1   7.480495   3.351711   2.232 0.025675 *
## factor(d_round)2   6.496369   3.352925   1.938 0.052744 .
## factor(d_round)4   4.308698   4.739716   0.909 0.363366
## factor(d_round)U   6.113294   3.353761   1.823 0.068398 .
## gp                 0.035489   0.002819  12.590  < 2e-16 ***
## reb                1.053677   0.030746  34.270  < 2e-16 ***
## ast                1.449659   0.038736  37.424  < 2e-16 ***
## net_rating         0.031154   0.007503   4.152 3.36e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.349 on 4461 degrees of freedom
## Multiple R-squared:  0.6636, Adjusted R-squared:  0.6627
## F-statistic: 799.9 on 11 and 4461 DF,  p-value: < 2.2e-16
```

```
nomodel<-lm(pts~1, data = bb)
# check to see if at least one variable is significant
anova(nomodel, fullmodel)
```

```
## Analysis of Variance Table
##
## Model 1: pts ~ 1
## Model 2: pts ~ age + height + weight + factor(d_round) + gp + reb + ast +
##     net_rating
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1   4472 148762
## 2   4461  50047 11     98715 799.92 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

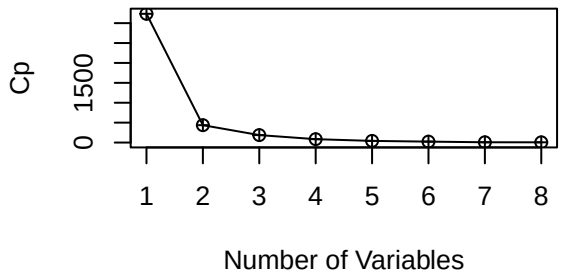
```
# Partial F test - we test to see if any of the variables are significant

# Ho: B1 = B2 = .... = Bi = o
# Ha: at least one Bi != 0

# we confirm that at least 1 variable is significant because we have p-values < 0.05,
# meaning we reject the null hypothesis.

# check to see if there are aliases in the model, remove alias if needed
alias(fullmodel)
```

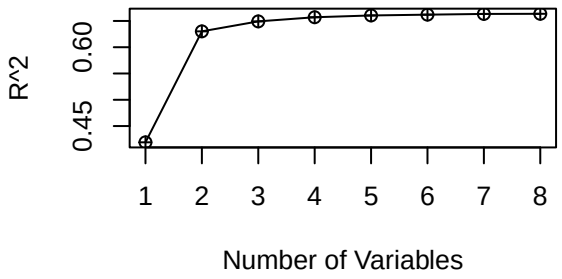
```
## Model :
## pts ~ age + height + weight + factor(d_round) + gp + reb + ast +
##     net_rating
```

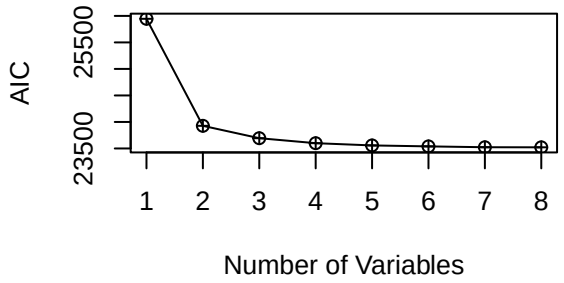
```
# Start model selection
# All-Possible-Regressions Selection Procedure
#Model selection by exhaustive search, forward or backward stepwise, or sequential
ks=ols_step_best_subset(fullmodel, details=TRUE)
# for the output interpretation
cp<-c(ks$cp)
rsquare<-c(ks$rsq)
AIC<-c(ks$aic)
AdjustedR<-c(ks$adjr)
cbind(cp,rsquare,AIC,AdjustedR)
```

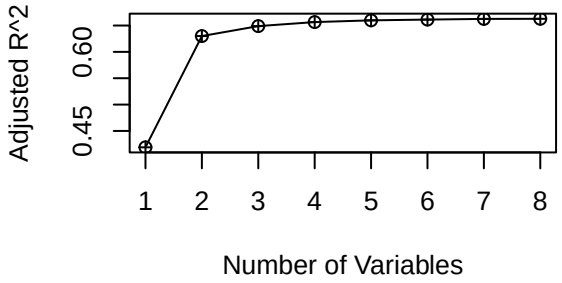
```
##               cp   rsquare      AIC AdjustedR
## [1,] 3234.324754 0.4190592 25945.18 0.4189293
## [2,]  436.864218 0.6301786 23927.05 0.6300131
## [3,]  187.341806 0.6491469 23693.53 0.6489114
## [4,]   84.618820 0.6570446 23599.69 0.6565069
## [5,]   41.578007 0.6604413 23557.17 0.6598328
## [6,]   23.244115 0.6619747 23538.92 0.6612931
## [7,]    7.363052 0.6633232 23523.04 0.6625687
## [8,]    6.000000 0.6635769 23521.67 0.6627473
```

```
par(mfrow=c(2,2)) # split the plotting panel into a 2 x 2 grid
plot(ks$cp,type = "o",pch=10, xlab="Number of Variables",ylab= "Cp")
plot(ks$rsq,type = "o",pch=10, xlab="Number of Variables",ylab= "R^2")
#plot(ks$rss, xlab="Number of Variables",ylab= "RMSE")
plot(ks$aic,type = "o",pch=10, xlab="Number of Variables",ylab= "AIC")
plot(ks$adjr,type = "o",pch=10, xlab="Number of Variables",ylab= "Adjusted R^2")
```










```
par(mfrow=c(1,1)) # change the plotting panel back into a 1 x 1 grid

# we can see that model with 8 variables has the lowest AIC, CP
# and the model with 7 variables has the highest the R^2, R^2_adj
# and the values are all very close between the 7 variable and 8 variable models
# so in this case, we decide to defer to Mallows's Cp Criterion,
# and choose a small cp value that is also close to p +1
# as that is a property that indicates that slight (or no) bias exists in
# the subset regression model.
# for model with 7 variables, p+1=8, Cp = 7.363052
# for model with 8 variables, p+1=9, Cp = 6.000000
# we can see that the model with 7 variables has p+1 most near to it's Cp
# so we'll go with a 7 variable model

# build model with stepwise, backward, forward methods
stepw=ols_step_both_p(fullmodel,pent = 0.05, prem = 0.1, details=FALSE)
backmodel=ols_step_backward_p(fullmodel, prem = 0.05, details=FALSE)
formodel=ols_step_forward_p(fullmodel, penter = 0.05, details=FALSE)
summary(fullmodel)
```

```
##
## Call:
## lm(formula = pts ~ age + height + weight + factor(d_round) +
##     gp + reb + ast + net_rating, data = bb)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.0784  -1.9670  -0.3645   1.5798  14.8123
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       4.378167   3.812950   1.148 0.250933
## age              -0.056669   0.012113  -4.678 2.98e-06 ***
## height           -0.041518   0.010978  -3.782 0.000158 ***
## weight           -0.014393   0.007848  -1.834 0.066741 .
## factor(d_round)1  7.480495   3.351711   2.232 0.025675 *
## factor(d_round)2  6.496369   3.352925   1.938 0.052744 .
## factor(d_round)4  4.308698   4.739716   0.909 0.363366
## factor(d_round)U  6.113294   3.353761   1.823 0.068398 .
## gp                0.035489   0.002819  12.590  < 2e-16 ***
## reb               1.053677   0.030746  34.270  < 2e-16 ***
## ast               1.449659   0.038736  37.424  < 2e-16 ***
## net_rating        0.031154   0.007503   4.152 3.36e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.349 on 4461 degrees of freedom
## Multiple R-squared:  0.6636, Adjusted R-squared:  0.6627
## F-statistic: 799.9 on 11 and 4461 DF,  p-value: < 2.2e-16
```

```
summary(stepw$model)
```

```
##
## Call:
## lm(formula = paste(response, "~", paste(preds, collapse = " + ")),
##     data = l)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.0975  -1.9652  -0.3589   1.5627  14.8259
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       5.566361   3.758492   1.481   0.1387
## reb               1.037147   0.029403  35.274  < 2e-16 ***
## ast               1.459321   0.038386  38.017  < 2e-16 ***
## gp                0.035851   0.002813  12.746  < 2e-16 ***
## factor(d_round)1  7.525851   3.352507   2.245   0.0248 *
## factor(d_round)2  6.525369   3.353775   1.946   0.0518 .
## factor(d_round)4  4.431459   4.740498   0.935   0.3499
## factor(d_round)U  6.156520   3.354566   1.835   0.0665 .
## height           -0.054158   0.008546  -6.337 2.57e-10 ***
## age              -0.060770   0.011908  -5.103 3.48e-07 ***
## net_rating        0.031704   0.007499   4.227 2.41e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.35 on 4462 degrees of freedom
## Multiple R-squared:  0.6633, Adjusted R-squared:  0.6626
```

```
## F-statistic: 879.1 on 10 and 4462 DF,  p-value: < 2.2e-16
```

```
summary(backmodel$model)
```

```
##
## Call:
## lm(formula = paste(response, "~", paste(preds, collapse = " + ")),
##     data = l)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.0975  -1.9652  -0.3589   1.5627  14.8259
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       5.566361   3.758492   1.481   0.1387
## age              -0.060770   0.011908  -5.103 3.48e-07 ***
## height           -0.054158   0.008546  -6.337 2.57e-10 ***
## factor(d_round)1  7.525851   3.352507   2.245   0.0248 *
## factor(d_round)2  6.525369   3.353775   1.946   0.0518 .
## factor(d_round)4  4.431459   4.740498   0.935   0.3499
## factor(d_round)U  6.156520   3.354566   1.835   0.0665 .
## gp                0.035851   0.002813  12.746  < 2e-16 ***
## reb               1.037147   0.029403  35.274  < 2e-16 ***
## ast               1.459321   0.038386  38.017  < 2e-16 ***
## net_rating        0.031704   0.007499   4.227 2.41e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.35 on 4462 degrees of freedom
## Multiple R-squared:  0.6633, Adjusted R-squared:  0.6626
## F-statistic: 879.1 on 10 and 4462 DF,  p-value: < 2.2e-16
```

```
summary(formodel$model)
```

```
##
## Call:
## lm(formula = paste(response, "~", paste(preds, collapse = " + ")),
##     data = l)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.0975  -1.9652  -0.3589   1.5627  14.8259
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       5.566361   3.758492   1.481   0.1387
## reb               1.037147   0.029403  35.274  < 2e-16 ***
## ast               1.459321   0.038386  38.017  < 2e-16 ***
## gp                0.035851   0.002813  12.746  < 2e-16 ***
## factor(d_round)1  7.525851   3.352507   2.245   0.0248 *
## factor(d_round)2  6.525369   3.353775   1.946   0.0518 .
## factor(d_round)4  4.431459   4.740498   0.935   0.3499
```

```
## factor(d_round)U  6.156520   3.354566   1.835   0.0665 .
## height           -0.054158   0.008546  -6.337 2.57e-10 ***
## age              -0.060770   0.011908  -5.103 3.48e-07 ***
## net_rating        0.031704   0.007499   4.227 2.41e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.35 on 4462 degrees of freedom
## Multiple R-squared:  0.6633, Adjusted R-squared:  0.6626
## F-statistic: 879.1 on 10 and 4462 DF,  p-value: < 2.2e-16
```

```r
# full model significant variables (individual t test)
# age, height, d_round, gp, reb, ast, net_rating

# stepwise model significant variables
# age, height, d_round, gp, reb, ast, net_rating

# forward model significant variables
# age, height, d_round, gp, reb, ast, net_rating

# backward model significant variables
# age, height, d_round, gp, reb, ast, net_rating


# build reduced model based on our All-Possible-Regressions Selection Procedure,
# Individual t-test, stepwise, forward, backward (all recommends 7 variables model)

# Complete partial F test to validate that dropping all but 7 variables
# is statistically significant

# Ho: Beta_weight = 0 (Beta's corresponding to weight, )
# Ha: Beta_weight != 0

reducedmodel<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating, data = bb)
summary(reducedmodel)
```

```
##
## Call:
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating, data = bb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.0975  -1.9652  -0.3589   1.5627  14.8259
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       5.566361   3.758492   1.481   0.1387
## age              -0.060770   0.011908  -5.103 3.48e-07 ***
## height           -0.054158   0.008546  -6.337 2.57e-10 ***
## factor(d_round)1  7.525851   3.352507   2.245   0.0248 *
## factor(d_round)2  6.525369   3.353775   1.946   0.0518 .
## factor(d_round)4  4.431459   4.740498   0.935   0.3499
## factor(d_round)U  6.156520   3.354566   1.835   0.0665 .
```

```
## gp                0.035851   0.002813  12.746  < 2e-16 ***
## reb               1.037147   0.029403  35.274  < 2e-16 ***
## ast               1.459321   0.038386  38.017  < 2e-16 ***
## net_rating        0.031704   0.007499   4.227 2.41e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.35 on 4462 degrees of freedom
## Multiple R-squared:  0.6633, Adjusted R-squared:  0.6626
## F-statistic: 879.1 on 10 and 4462 DF,  p-value: < 2.2e-16
```

```
anova(reducedmodel, fullmodel)
```

```
## Analysis of Variance Table
##
## Model 1: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating
## Model 2: pts ~ age + height + weight + factor(d_round) + gp + reb + ast +
##     net_rating
##   Res.Df   RSS Df Sum of Sq      F  Pr(>F)
## 1   4462 50085
## 2   4461 50047  1    37.729 3.3631 0.06674 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# The Partial F test gives a p-value = 0.06674 > alpha = 0.05, therefore we fail
# to reject Ho and conclude the weight variable should be dropped from the model
```

Multicollinearity

```
# test for multicollinearity
vif(reducedmodel)
```

```
##                     GVIF Df GVIF^(1/(2*Df))
## age             1.036771  1        1.018220
## height          2.246024  1        1.498674
## factor(d_round) 1.173895  4        1.020243
## gp              1.379724  1        1.174616
## reb             2.035243  1        1.426619
## ast             1.901594  1        1.378983
## net_rating      1.169999  1        1.081665
```

The VIF output concludes that there are no statistically significant variables with multicollinearity because all VIF values are <3.

Interactions between variables

```
# check for interaction and revise models
interacmodel<-lm(pts~(age+height+factor(d_round)+gp+reb+ast+net_rating)^2, data = bb)
summary(interacmodel)
```

```
##
## Call:
## lm(formula = pts ~ (age + height + factor(d_round) + gp + reb +
##     ast + net_rating)^2, data = bb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4748  -1.7855  -0.2299   1.4306  17.1657
##
## Coefficients: (12 not defined because of singularities)
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  -2.253e+01  1.272e+01  -1.771 0.076555 .
## age                           6.242e-01  3.976e-01   1.570 0.116519
## height                        1.037e-01  6.193e-02   1.674 0.094143 .
## factor(d_round)1              9.093e+00  5.668e+00   1.604 0.108740
## factor(d_round)2              1.239e+01  6.042e+00   2.051 0.040327 *
## factor(d_round)4              3.332e+00  4.523e+00   0.737 0.461350
## factor(d_round)U              3.591e+00  3.205e+00   1.120 0.262576
## gp                            1.578e-01  8.939e-02   1.765 0.077616 .
## reb                           9.036e+00  7.699e-01  11.737  < 2e-16 ***
## ast                          -1.067e+01  8.290e-01 -12.874  < 2e-16 ***
## net_rating                    2.013e-01  2.243e-01   0.898 0.369357
## age:height                   -3.512e-03  1.995e-03  -1.761 0.078388 .
## age:factor(d_round)1          5.129e-02  3.781e-02   1.357 0.174913
## age:factor(d_round)2          1.119e-01  4.386e-02   2.552 0.010741 *
## age:factor(d_round)4                 NA         NA      NA       NA
## age:factor(d_round)U                 NA         NA      NA       NA
## age:gp                        1.035e-03  6.443e-04   1.607 0.108161
## age:reb                      -9.350e-03  7.162e-03  -1.305 0.191816
## age:ast                      -3.042e-02  8.963e-03  -3.394 0.000696 ***
## age:net_rating               -1.301e-03  1.624e-03  -0.801 0.423187
## height:factor(d_round)1      -3.980e-02  2.308e-02  -1.724 0.084748 .
## height:factor(d_round)2      -6.355e-02  2.540e-02  -2.502 0.012400 *
## height:factor(d_round)4              NA         NA      NA       NA
## height:factor(d_round)U              NA         NA      NA       NA
## height:gp                    -7.450e-04  4.459e-04  -1.671 0.094877 .
## height:reb                   -3.765e-02  3.579e-03 -10.521  < 2e-16 ***
## height:ast                    6.615e-02  4.096e-03  16.147  < 2e-16 ***
## height:net_rating            -9.080e-04  1.120e-03  -0.811 0.417618
## factor(d_round)1:gp           2.410e-03  7.611e-03   0.317 0.751552
## factor(d_round)2:gp           4.998e-03  8.528e-03   0.586 0.557849
## factor(d_round)4:gp                  NA         NA      NA       NA
## factor(d_round)U:gp                  NA         NA      NA       NA
## factor(d_round)1:reb          5.744e-01  1.128e-01   5.094 3.66e-07 ***
## factor(d_round)2:reb          3.066e-01  1.210e-01   2.533 0.011344 *
## factor(d_round)4:reb                 NA         NA      NA       NA
## factor(d_round)U:reb                 NA         NA      NA       NA
## factor(d_round)1:ast          3.477e-01  1.367e-01   2.544 0.010991 *
## factor(d_round)2:ast          1.620e-01  1.592e-01   1.018 0.308775
## factor(d_round)4:ast                 NA         NA      NA       NA
## factor(d_round)U:ast                 NA         NA      NA       NA
## factor(d_round)1:net_rating   1.380e-02  2.002e-02   0.689 0.490599
## factor(d_round)2:net_rating -2.453e-02  2.101e-02  -1.168 0.243014
## factor(d_round)4:net_rating          NA         NA      NA       NA
```

```
## factor(d_round)U:net_rating         NA         NA      NA       NA
## gp:reb                       -4.065e-03  1.539e-03  -2.641 0.008306 **
## gp:ast                        2.437e-03  2.312e-03   1.054 0.291991
## gp:net_rating                -4.410e-04  3.836e-04  -1.150 0.250287
## reb:ast                      -1.185e-01  1.724e-02  -6.873 7.15e-12 ***
## reb:net_rating                1.139e-02  4.457e-03   2.556 0.010628 *
## ast:net_rating                1.677e-02  6.294e-03   2.665 0.007722 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.183 on 4435 degrees of freedom
## Multiple R-squared:  0.6979, Adjusted R-squared:  0.6954
## F-statistic:   277 on 37 and 4435 DF,  p-value: < 2.2e-16
```

```
# reduce model by taking out insignificant interactions terms
reducedmodel2<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating
                  +age:factor(d_round)+age:ast
                  +height:factor(d_round)+height:reb+height:ast
                  +factor(d_round):reb+factor(d_round):ast
                  +gp:reb+reb:ast+reb:net_rating+ast:net_rating
                  ,data = bb)
summary(reducedmodel2)
```

```
##
## Call:
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating + age:factor(d_round) + age:ast + height:factor(d_round) +
##     height:reb + height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:ast + reb:net_rating + ast:net_rating, data = bb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4200  -1.8125  -0.2069   1.4890  17.4703
##
## Coefficients: (8 not defined because of singularities)
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -0.476984   5.253319  -0.091 0.927658
## age                      -0.058895   0.034038  -1.730 0.083656 .
## height                   -0.009873   0.020461  -0.483 0.629445
## factor(d_round)1         11.439660   5.557775   2.058 0.039618 *
## factor(d_round)2         12.230594   5.996763   2.040 0.041456 *
## factor(d_round)4          2.976173   4.527946   0.657 0.511029
## factor(d_round)U          3.429278   3.210703   1.068 0.285544
## gp                        0.046426   0.004850   9.573  < 2e-16 ***
## reb                       9.066040   0.709778  12.773  < 2e-16 ***
## ast                      -9.910314   0.775347 -12.782  < 2e-16 ***
## net_rating               -0.030862   0.012001  -2.572 0.010156 *
## age:factor(d_round)1      0.039860   0.035805   1.113 0.265652
## age:factor(d_round)2      0.093878   0.042310   2.219 0.026549 *
## age:factor(d_round)4            NA         NA      NA       NA
## age:factor(d_round)U            NA         NA      NA       NA
## age:ast                  -0.023744   0.006800  -3.492 0.000484 ***
## height:factor(d_round)1 -0.051480   0.022450  -2.293 0.021888 *
## height:factor(d_round)2 -0.060182   0.025096  -2.398 0.016524 *
```

```
## height:factor(d_round)4        NA         NA      NA       NA
## height:factor(d_round)U        NA         NA      NA       NA
## height:reb               -0.039188   0.003460 -11.326  < 2e-16 ***
## height:ast                0.062048   0.003932  15.778  < 2e-16 ***
## factor(d_round)1:reb      0.651239   0.103410   6.298 3.31e-10 ***
## factor(d_round)2:reb      0.328966   0.111395   2.953 0.003162 **
## factor(d_round)4:reb           NA         NA      NA       NA
## factor(d_round)U:reb           NA         NA      NA       NA
## factor(d_round)1:ast      0.330426   0.132574   2.492 0.012725 *
## factor(d_round)2:ast      0.179608   0.155302   1.157 0.247538
## factor(d_round)4:ast           NA         NA      NA       NA
## factor(d_round)U:ast           NA         NA      NA       NA
## gp:reb                   -0.004868   0.001203  -4.046 5.31e-05 ***
## reb:ast                  -0.112881   0.016542  -6.824 1.00e-11 ***
## reb:net_rating            0.007599   0.003205   2.371 0.017784 *
## ast:net_rating            0.021902   0.004592   4.770 1.90e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.19 on 4447 degrees of freedom
## Multiple R-squared:  0.6958, Adjusted R-squared:  0.6941
## F-statistic: 406.9 on 25 and 4447 DF,  p-value: < 2.2e-16
```

```
anova(reducedmodel2, interacmodel)
```

```
## Analysis of Variance Table
##
## Model 1: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:factor(d_round) + age:ast + height:factor(d_round) +
##     height:reb + height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:ast + reb:net_rating + ast:net_rating
## Model 2: pts ~ (age + height + factor(d_round) + gp + reb + ast + net_rating)^2
##   Res.Df   RSS Df Sum of Sq      F   Pr(>F)
## 1   4447 45253
## 2   4435 44935 12    317.63 2.6125 0.001788 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Partial F test on interacmodel (all possible interactions) and
# reducedmodel2 with insignificant interactions removed.

# Ho: B1 = B2 = .... = Bi = o
# Ha: at least one Bi != 0

# The Partial F test gives a p-value = 0.001788 <alpha = 0.05, therefore we
# reject Ho and conclude that we shouldn't have simultaneously dropped all
# of the interactions from the model that we did.


# this reduction is no good, we took out a significant term,
# put back the term with the next lowest p value from the individual t test
# of the full interaction model
# and that's age:height with a p-value of 0.078388
```

```
reducedmodel3<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating
                  +age:height+age:factor(d_round)+age:ast
                  +height:factor(d_round)+height:reb+height:ast
                  +factor(d_round):reb+factor(d_round):ast
                  +gp:reb+reb:ast+reb:net_rating+ast:net_rating
                  ,data = bb)
summary(reducedmodel3)
```

```
##
## Call:
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating + age:height + age:factor(d_round) + age:ast +
##     height:factor(d_round) + height:reb + height:ast + factor(d_round):reb +
##     factor(d_round):ast + gp:reb + reb:ast + reb:net_rating +
##     ast:net_rating, data = bb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.3314  -1.8249  -0.2162   1.4984  17.4371
##
## Coefficients: (8 not defined because of singularities)
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -28.969375   9.837933  -2.945 0.003250 **
## age                        0.989746   0.308162   3.212 0.001329 **
## height                     0.132014   0.046207   2.857 0.004296 **
## factor(d_round)1          11.123940   5.551853   2.004 0.045169 *
## factor(d_round)2          13.140414   5.995439   2.192 0.028450 *
## factor(d_round)4           2.836564   4.522681   0.627 0.530569
## factor(d_round)U           3.403213   3.206848   1.061 0.288642
## gp                         0.046321   0.004844   9.563  < 2e-16 ***
## reb                        9.105749   0.709019  12.843  < 2e-16 ***
## ast                       -9.815323   0.774911 -12.666  < 2e-16 ***
## net_rating                -0.032987   0.012003  -2.748 0.006014 **
## age:height                -0.005221   0.001525  -3.424 0.000623 ***
## age:factor(d_round)1       0.067319   0.036650   1.837 0.066304 .
## age:factor(d_round)2       0.114647   0.042692   2.685 0.007270 **
## age:factor(d_round)4             NA         NA      NA       NA
## age:factor(d_round)U             NA         NA      NA       NA
## age:ast                   -0.035864   0.007659  -4.683 2.91e-06 ***
## height:factor(d_round)1   -0.053541   0.022431  -2.387 0.017029 *
## height:factor(d_round)2   -0.067441   0.025156  -2.681 0.007369 **
## height:factor(d_round)4          NA         NA      NA       NA
## height:factor(d_round)U          NA         NA      NA       NA
## height:reb                -0.039301   0.003456 -11.372  < 2e-16 ***
## height:ast                 0.063267   0.003944  16.042  < 2e-16 ***
## factor(d_round)1:reb       0.635553   0.103387   6.147 8.57e-10 ***
## factor(d_round)2:reb       0.326163   0.111264   2.931 0.003391 **
## factor(d_round)4:reb             NA         NA      NA       NA
## factor(d_round)U:reb             NA         NA      NA       NA
## factor(d_round)1:ast       0.328010   0.132417   2.477 0.013282 *
## factor(d_round)2:ast       0.158585   0.155237   1.022 0.307042
## factor(d_round)4:ast             NA         NA      NA       NA
## factor(d_round)U:ast             NA         NA      NA       NA
```

```
## gp:reb                   -0.004964   0.001202  -4.129 3.70e-05 ***
## reb:ast                  -0.112910   0.016522  -6.834 9.37e-12 ***
## reb:net_rating            0.008579   0.003214   2.669 0.007628 **
## ast:net_rating            0.021410   0.004588   4.666 3.16e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.186 on 4446 degrees of freedom
## Multiple R-squared:  0.6966, Adjusted R-squared:  0.6948
## F-statistic: 392.6 on 26 and 4446 DF,  p-value: < 2.2e-16
```

```
anova(reducedmodel3, interacmodel)
```

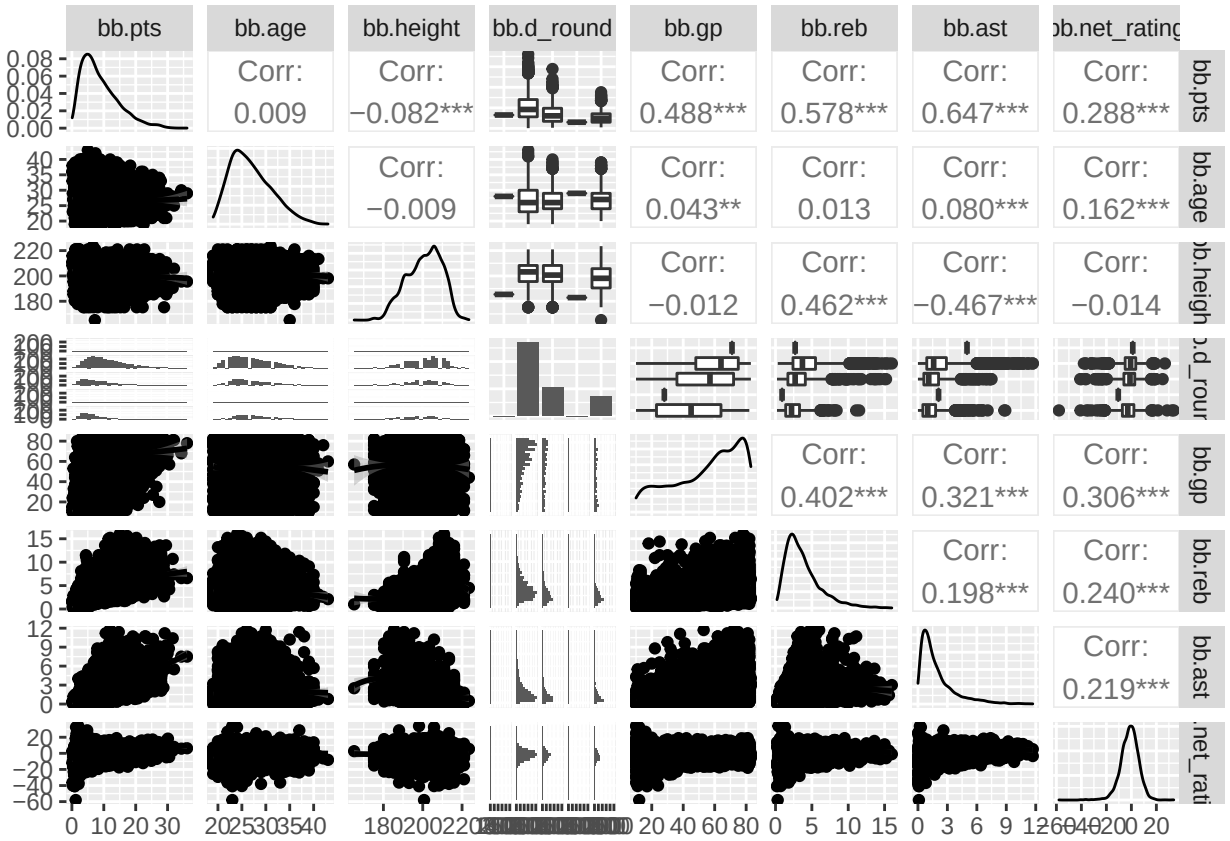
```
## Analysis of Variance Table
##
## Model 1: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:height + age:factor(d_round) + age:ast + height:factor(d_round) +
##     height:reb + height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:ast + reb:net_rating + ast:net_rating
## Model 2: pts ~ (age + height + factor(d_round) + gp + reb + ast + net_rating)^2
##   Res.Df   RSS Df Sum of Sq      F  Pr(>F)
## 1   4446 45134
## 2   4435 44935 11    198.64 1.7823 0.05142 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Partial F test on interac model (all possible interactions) and
# reducedmodel3 with insignificant interactions removed.

# Ho: B1 = B2 = .... = Bi = o
# Ha: at least one Bi != 0

# The Partial F test gives a p-value = 0.05142 > alpha = 0.05, therefore we fail
# to reject Ho and conclude all that we should have simultaneously dropped all
#  of the interactions from the model that we did.
```

Higher order variables & further insigificant interactions

```
# check for need of higher order model
# gg pairs plot
pairsdf <-data.frame(bb$pts,bb$age,bb$height,bb$d_round,bb$gp,bb$reb,bb$ast,bb$net_rating)

ggpairs(pairsdf,lower = list(continuous = "smooth_loess", combo =
                               "facethist", discrete = "facetbar", na = "na"))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# get best higher order models, try gp, reb, ast,  because they have
# correlation values that are close to 0.5 or greater
highermodel<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating
                +age:height+age:factor(d_round)+age:ast
                +height:factor(d_round)+height:reb+height:ast
                +factor(d_round):reb+factor(d_round):ast
                +gp:reb+reb:ast+reb:net_rating+ast:net_rating
                +poly(ast,2,raw=TRUE)+poly(reb,2,raw=TRUE)+poly(gp,2,raw=TRUE)
                ,data = bb)
summary(highermodel)
```

```
##
## Call:
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating + age:height + age:factor(d_round) + age:ast +
##     height:factor(d_round) + height:reb + height:ast + factor(d_round):reb +
##     factor(d_round):ast + gp:reb + reb:ast + reb:net_rating +
##     ast:net_rating + poly(ast, 2, raw = TRUE) + poly(reb, 2,
##     raw = TRUE) + poly(gp, 2, raw = TRUE), data = bb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4570  -1.8040  -0.1638   1.4555  15.6432
##
## Coefficients: (11 not defined because of singularities)
##                              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)               -3.089e+01  9.668e+00  -3.195 0.001407 **
## age                        7.259e-01  3.022e-01   2.402 0.016358 *
## height                     1.418e-01  4.543e-02   3.120 0.001818 **
## factor(d_round)1           6.280e+00  5.505e+00   1.141 0.254055
## factor(d_round)2           1.353e+01  5.866e+00   2.306 0.021155 *
## factor(d_round)4           2.910e+00  4.421e+00   0.658 0.510483
## factor(d_round)U           3.897e+00  3.135e+00   1.243 0.213812
## gp                         1.488e-02  1.218e-02   1.222 0.221791
## reb                        6.838e+00  8.129e-01   8.412  < 2e-16 ***
## ast                       -6.570e-01  9.902e-01  -0.664 0.507020
## net_rating                -3.588e-02  1.176e-02  -3.052 0.002286 **
## poly(ast, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2 -1.948e-01  1.401e-02 -13.907  < 2e-16 ***
## poly(reb, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(reb, 2, raw = TRUE)2 -2.738e-02  7.725e-03  -3.544 0.000398 ***
## poly(gp, 2, raw = TRUE)1          NA         NA      NA       NA
## poly(gp, 2, raw = TRUE)2   2.477e-04  1.281e-04   1.933 0.053252 .
## age:height                -3.945e-03  1.495e-03  -2.638 0.008359 **
## age:factor(d_round)1       4.528e-02  3.590e-02   1.261 0.207285
## age:factor(d_round)2       1.053e-01  4.176e-02   2.521 0.011742 *
## age:factor(d_round)4              NA         NA      NA       NA
## age:factor(d_round)U              NA         NA      NA       NA
## age:ast                   -2.645e-02  7.527e-03  -3.514 0.000445 ***
## height:factor(d_round)1   -2.511e-02  2.244e-02  -1.119 0.263348
## height:factor(d_round)2   -6.561e-02  2.463e-02  -2.664 0.007751 **
## height:factor(d_round)4           NA         NA      NA       NA
## height:factor(d_round)U           NA         NA      NA       NA
## height:reb                -2.821e-02  4.098e-03  -6.885 6.57e-12 ***
## height:ast                 1.815e-02  4.967e-03   3.654 0.000261 ***
## factor(d_round)1:reb       5.244e-01  1.025e-01   5.118 3.21e-07 ***
## factor(d_round)2:reb       3.240e-01  1.091e-01   2.969 0.003008 **
## factor(d_round)4:reb              NA         NA      NA       NA
## factor(d_round)U:reb              NA         NA      NA       NA
## factor(d_round)1:ast       6.432e-01  1.330e-01   4.837 1.36e-06 ***
## factor(d_round)2:ast       1.421e-01  1.523e-01   0.933 0.350828
## factor(d_round)4:ast              NA         NA      NA       NA
## factor(d_round)U:ast              NA         NA      NA       NA
## gp:reb                    -3.761e-03  1.303e-03  -2.888 0.003902 **
## reb:ast                    3.334e-02  1.950e-02   1.710 0.087421 .
## reb:net_rating             6.812e-03  3.161e-03   2.155 0.031206 *
## ast:net_rating             2.589e-02  4.519e-03   5.728 1.08e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.114 on 4443 degrees of freedom
## Multiple R-squared:  0.7103, Adjusted R-squared:  0.7084
## F-statistic: 375.7 on 29 and 4443 DF,  p-value: < 2.2e-16
```

```
anova(reducedmodel3,highermodel)
```

```
## Analysis of Variance Table
##
## Model 1: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:height + age:factor(d_round) + age:ast + height:factor(d_round) +
```

```
##     height:reb + height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:ast + reb:net_rating + ast:net_rating
## Model 2: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:height + age:factor(d_round) + age:ast + height:factor(d_round) +
##     height:reb + height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:ast + reb:net_rating + ast:net_rating + poly(ast,
##     2, raw = TRUE) + poly(reb, 2, raw = TRUE) + poly(gp, 2, raw = TRUE)
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1   4446 45134
## 2   4443 43091  3    2042.5 70.198 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Partial F test on reducedmodel3 (best first order model with interactions) and
# highermodel (adding 2 higher order terms simultaneously)

# Ho: Beta_ast^2 = Beta_reb^2 = 0 (Beta's associated with higher order)
# Ha: at least one Beta_i != 0

# The Partial F test gives a p-value = 2.2e-16 < alpha = 0.05, therefore we
# reject Ho and conclude all adding the two higher order terms is correct.


# the higher order term for gp is insignificant, we'll take that out
highermodel2<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating
                 +age:height+age:factor(d_round)+age:ast
                 +height:factor(d_round)+height:reb+height:ast
                 +factor(d_round):reb+factor(d_round):ast
                 +gp:reb+reb:ast+reb:net_rating+ast:net_rating
                 +poly(ast,2,raw=TRUE)+poly(reb,2,raw=TRUE)
                 ,data = bb)
summary(highermodel2)
```

```
##
## Call:
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating + age:height + age:factor(d_round) + age:ast +
##     height:factor(d_round) + height:reb + height:ast + factor(d_round):reb +
##     factor(d_round):ast + gp:reb + reb:ast + reb:net_rating +
##     ast:net_rating + poly(ast, 2, raw = TRUE) + poly(reb, 2,
##     raw = TRUE), data = bb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4805  -1.8173  -0.1786   1.4540  15.7086
##
## Coefficients: (10 not defined because of singularities)
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -31.729009   9.661669  -3.284 0.001031 **
## age                         0.730917   0.302306   2.418 0.015655 *
## height                      0.144728   0.045422   3.186 0.001451 **
## factor(d_round)1            6.055901   5.505519   1.100 0.271405
## factor(d_round)2           13.425280   5.867640   2.288 0.022183 *
## factor(d_round)4            2.867152   4.422264   0.648 0.516795
```

```
## factor(d_round)U            3.883823   3.135653   1.239 0.215559
## gp                          0.036401   0.004940   7.368 2.05e-13 ***
## reb                         6.894047   0.812611   8.484  < 2e-16 ***
## ast                        -0.643261   0.990498  -0.649 0.516093
## net_rating                 -0.035850   0.011761  -3.048 0.002314 **
## poly(ast, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2  -0.195752   0.014003 -13.979  < 2e-16 ***
## poly(reb, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(reb, 2, raw = TRUE)2  -0.027422   0.007728  -3.549 0.000391 ***
## age:height                 -0.003984   0.001496  -2.664 0.007760 **
## age:factor(d_round)1        0.049028   0.035857   1.367 0.171589
## age:factor(d_round)2        0.107982   0.041753   2.586 0.009734 **
## age:factor(d_round)4              NA         NA      NA       NA
## age:factor(d_round)U              NA         NA      NA       NA
## age:ast                    -0.027068   0.007523  -3.598 0.000324 ***
## height:factor(d_round)1    -0.024704   0.022449  -1.100 0.271201
## height:factor(d_round)2    -0.065663   0.024637  -2.665 0.007723 **
## height:factor(d_round)4           NA         NA      NA       NA
## height:factor(d_round)U           NA         NA      NA       NA
## height:reb                 -0.028752   0.004090  -7.031 2.37e-12 ***
## height:ast                  0.018227   0.004969   3.668 0.000247 ***
## factor(d_round)1:reb        0.525218   0.102491   5.125 3.11e-07 ***
## factor(d_round)2:reb        0.325762   0.109161   2.984 0.002858 **
## factor(d_round)4:reb              NA         NA      NA       NA
## factor(d_round)U:reb              NA         NA      NA       NA
## factor(d_round)1:ast        0.649967   0.132962   4.888 1.05e-06 ***
## factor(d_round)2:ast        0.146920   0.152331   0.964 0.334857
## factor(d_round)4:ast              NA         NA      NA       NA
## factor(d_round)U:ast              NA         NA      NA       NA
## gp:reb                     -0.002831   0.001211  -2.338 0.019417 *
## reb:ast                     0.031517   0.019485   1.617 0.105851
## reb:net_rating              0.006666   0.003161   2.109 0.034994 *
## ast:net_rating              0.026427   0.004512   5.857 5.04e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.115 on 4444 degrees of freedom
## Multiple R-squared:  0.7101, Adjusted R-squared:  0.7083
## F-statistic: 388.7 on 28 and 4444 DF,  p-value: < 2.2e-16
```

```
# we note that the interaction term (reb*ast) became insignificant,
# since no higher order term depends on that interaction term
# we remove that and test with anova() to see if that is indeed insignificant and can be removed
reducedmodel4<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating
                  +age:height+age:factor(d_round)+age:ast
                  +height:factor(d_round)+height:reb+height:ast
                  +factor(d_round):reb+factor(d_round):ast
                  +gp:reb+reb:net_rating+ast:net_rating
                  +poly(ast,2,raw=TRUE)+poly(reb,2,raw=TRUE)
                  ,data = bb)
summary(reducedmodel4)
```

```
##
## Call:
```

```
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating + age:height + age:factor(d_round) + age:ast +
##     height:factor(d_round) + height:reb + height:ast + factor(d_round):reb +
##     factor(d_round):ast + gp:reb + reb:net_rating + ast:net_rating +
##     poly(ast, 2, raw = TRUE) + poly(reb, 2, raw = TRUE), data = bb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.2409  -1.8155  -0.1664   1.4605  15.8963
##
## Coefficients: (10 not defined because of singularities)
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -32.050455   9.661380  -3.317 0.000916 ***
## age                          0.740616   0.302302   2.450 0.014327 *
## height                       0.146424   0.045418   3.224 0.001274 **
## factor(d_round)1             5.665894   5.501236   1.030 0.303098
## factor(d_round)2            13.097290   5.865201   2.233 0.025595 *
## factor(d_round)4             2.842855   4.423043   0.643 0.520428
## factor(d_round)U             3.829283   3.136042   1.221 0.222130
## gp                           0.035713   0.004923   7.255 4.72e-13 ***
## reb                          7.442718   0.738557  10.077  < 2e-16 ***
## ast                         -1.567628   0.809146  -1.937 0.052762 .
## net_rating                  -0.038505   0.011648  -3.306 0.000954 ***
## poly(ast, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2  -0.185547   0.012504 -14.839  < 2e-16 ***
## poly(reb, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(reb, 2, raw = TRUE)2  -0.023207   0.007276  -3.189 0.001435 **
## age:height                 -0.004029   0.001496  -2.694 0.007093 **
## age:factor(d_round)1        0.049382   0.035863   1.377 0.168592
## age:factor(d_round)2        0.107248   0.041758   2.568 0.010251 *
## age:factor(d_round)4              NA         NA      NA       NA
## age:factor(d_round)U              NA         NA      NA       NA
## age:ast                    -0.027628   0.007516  -3.676 0.000240 ***
## height:factor(d_round)1    -0.023389   0.022439  -1.042 0.297312
## height:factor(d_round)2    -0.064328   0.024628  -2.612 0.009032 **
## height:factor(d_round)4           NA         NA      NA       NA
## height:factor(d_round)U           NA         NA      NA       NA
## height:reb                 -0.031448   0.003735  -8.420  < 2e-16 ***
## height:ast                  0.023307   0.003851   6.053 1.54e-09 ***
## factor(d_round)1:reb        0.538950   0.102158   5.276 1.39e-07 ***
## factor(d_round)2:reb        0.327982   0.109172   3.004 0.002677 **
## factor(d_round)4:reb              NA         NA      NA       NA
## factor(d_round)U:reb              NA         NA      NA       NA
## factor(d_round)1:ast        0.664217   0.132694   5.006 5.78e-07 ***
## factor(d_round)2:ast        0.165472   0.151926   1.089 0.276141
## factor(d_round)4:ast              NA         NA      NA       NA
## factor(d_round)U:ast              NA         NA      NA       NA
## gp:reb                     -0.002770   0.001211  -2.288 0.022169 *
## reb:net_rating              0.007052   0.003152   2.237 0.025321 *
## ast:net_rating              0.027681   0.004445   6.227 5.20e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.116 on 4445 degrees of freedom
```

```
## Multiple R-squared:  0.7099, Adjusted R-squared:  0.7082
## F-statistic: 402.9 on 27 and 4445 DF,  p-value: < 2.2e-16
```

```
anova(reducedmodel4, highermodel2)
```

```
## Analysis of Variance Table
##
## Model 1: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:height + age:factor(d_round) + age:ast + height:factor(d_round) +
##     height:reb + height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:net_rating + ast:net_rating + poly(ast, 2, raw = TRUE) +
##     poly(reb, 2, raw = TRUE)
## Model 2: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:height + age:factor(d_round) + age:ast + height:factor(d_round) +
##     height:reb + height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:ast + reb:net_rating + ast:net_rating + poly(ast,
##     2, raw = TRUE) + poly(reb, 2, raw = TRUE)
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1   4445 43153
## 2   4444 43127  1    25.389 2.6162 0.1059
```

```
# Partial F test on highermodel2 (higher order model with interactions) and
# reducedmodel4 (higher order model with reb:ast interaction removed)

# Ho: Beta_reb:ast = 0
# Ha: Beta_reb:ast != 0

# The Partial F test gives a p-value = 0.1059 > alpha = 0.05, therefore we fail
# to reject Ho and conclude that we should have dropped the reb:ast
# interaction from the model.
# Note: the individual t test from summary() would have done the trick as well

# we try even higher order terms, specifically ast^3 and then reb^3,
highermodel3<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating
                 +age:height+age:factor(d_round)+age:ast
                 +height:factor(d_round)+height:reb+height:ast
                 +factor(d_round):reb+factor(d_round):ast+gp:reb
                 +reb:ast+reb:net_rating+ast:net_rating
                 +poly(ast,3,raw=TRUE)+poly(reb,2,raw=TRUE)
                 ,data = bb)
summary(highermodel3)
```

```
##
## Call:
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating + age:height + age:factor(d_round) + age:ast +
##     height:factor(d_round) + height:reb + height:ast + factor(d_round):reb +
##     factor(d_round):ast + gp:reb + reb:ast + reb:net_rating +
##     ast:net_rating + poly(ast, 3, raw = TRUE) + poly(reb, 2,
##     raw = TRUE), data = bb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -13.5210  -1.8117  -0.1899   1.4567  15.6845
##
## Coefficients: (10 not defined because of singularities)
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -31.102758   9.728743  -3.197 0.001398 **
## age                          0.725732   0.302476   2.399 0.016467 *
## height                       0.141582   0.045781   3.093 0.001997 **
## factor(d_round)1             6.063348   5.505966   1.101 0.270855
## factor(d_round)2            13.486545   5.869148   2.298 0.021615 *
## factor(d_round)4             2.923326   4.423780   0.661 0.508762
## factor(d_round)U             3.917134   3.136478   1.249 0.211769
## gp                           0.036717   0.004973   7.382 1.84e-13 ***
## reb                          6.888909   0.812728   8.476  < 2e-16 ***
## ast                         -0.865459   1.069174  -0.809 0.418291
## net_rating                  -0.035545   0.011774  -3.019 0.002552 **
## poly(ast, 3, raw = TRUE)1         NA         NA      NA       NA
## poly(ast, 3, raw = TRUE)2  -0.170325   0.048126  -3.539 0.000405 ***
## poly(ast, 3, raw = TRUE)3  -0.001755   0.003178  -0.552 0.580812
## poly(reb, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(reb, 2, raw = TRUE)2  -0.027634   0.007738  -3.571 0.000359 ***
## age:height                 -0.003959   0.001496  -2.645 0.008188 **
## age:factor(d_round)1        0.049219   0.035861   1.372 0.169984
## age:factor(d_round)2        0.108114   0.041757   2.589 0.009653 **
## age:factor(d_round)4              NA         NA      NA       NA
## age:factor(d_round)U              NA         NA      NA       NA
## age:ast                    -0.026779   0.007541  -3.551 0.000388 ***
## height:factor(d_round)1    -0.024580   0.022452  -1.095 0.273684
## height:factor(d_round)2    -0.065800   0.024641  -2.670 0.007604 **
## height:factor(d_round)4           NA         NA      NA       NA
## height:factor(d_round)U           NA         NA      NA       NA
## height:reb                 -0.028685   0.004092  -7.011 2.73e-12 ***
## height:ast                  0.018883   0.005109   3.696 0.000222 ***
## factor(d_round)1:reb        0.526709   0.102535   5.137 2.91e-07 ***
## factor(d_round)2:reb        0.327293   0.109205   2.997 0.002741 **
## factor(d_round)4:reb              NA         NA      NA       NA
## factor(d_round)U:reb              NA         NA      NA       NA
## factor(d_round)1:ast        0.646534   0.133118   4.857 1.23e-06 ***
## factor(d_round)2:ast        0.142413   0.152561   0.933 0.350623
## factor(d_round)4:ast              NA         NA      NA       NA
## factor(d_round)U:ast              NA         NA      NA       NA
## gp:reb                     -0.002877   0.001214  -2.370 0.017834 *
## reb:ast                     0.030493   0.019575   1.558 0.119353
## reb:net_rating              0.006596   0.003164   2.085 0.037138 *
## ast:net_rating              0.026337   0.004515   5.833 5.83e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.115 on 4443 degrees of freedom
## Multiple R-squared:  0.7101, Adjusted R-squared:  0.7082
## F-statistic: 375.3 on 29 and 4443 DF,  p-value: < 2.2e-16
```

```
# the higher orders terms with ast were insignificant based on individual
# t-test p-value = 0.580812 > alpha = 0.05 for ast^3
```

```r
# we now try reb^3,
highermodel4<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating
                 +age:height+age:factor(d_round)+age:ast
                 +height:factor(d_round)+height:reb+height:ast
                 +factor(d_round):reb+factor(d_round):ast+gp:reb
                 +reb:ast+reb:net_rating+ast:net_rating
                 +poly(ast,2,raw=TRUE)+poly(reb,3,raw=TRUE)
                 ,data = bb)
summary(highermodel4)
```

```
##
## Call:
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating + age:height + age:factor(d_round) + age:ast +
##     height:factor(d_round) + height:reb + height:ast + factor(d_round):reb +
##     factor(d_round):ast + gp:reb + reb:ast + reb:net_rating +
##     ast:net_rating + poly(ast, 2, raw = TRUE) + poly(reb, 3,
##     raw = TRUE), data = bb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4499  -1.8023  -0.1801   1.4581  15.7124
##
## Coefficients: (10 not defined because of singularities)
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 -29.346924   9.751475  -3.009 0.002632 **
## age                           0.716520   0.302341   2.370 0.017835 *
## height                        0.131550   0.046010   2.859 0.004267 **
## factor(d_round)1              6.386158   5.507297   1.160 0.246282
## factor(d_round)2             13.107620   5.868918   2.233 0.025572 *
## factor(d_round)4              2.859261   4.421187   0.647 0.517848
## factor(d_round)U              3.883414   3.134887   1.239 0.215495
## gp                            0.033645   0.005176   6.501 8.87e-11 ***
## reb                           6.575165   0.831910   7.904 3.39e-15 ***
## ast                          -0.700582   0.990779  -0.707 0.479540
## net_rating                   -0.037576   0.011798  -3.185 0.001457 **
## poly(ast, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2  -0.195101   0.014004 -13.931  < 2e-16 ***
## poly(reb, 3, raw = TRUE)1         NA         NA      NA       NA
## poly(reb, 3, raw = TRUE)2  -0.086662   0.034149  -2.538 0.011190 *
## poly(reb, 3, raw = TRUE)3   0.002746   0.001542   1.781 0.074991 .
## age:height                  -0.003902   0.001496  -2.608 0.009127 **
## age:factor(d_round)1         0.046675   0.035872   1.301 0.193282
## age:factor(d_round)2         0.107497   0.041743   2.575 0.010050 *
## age:factor(d_round)4               NA         NA      NA       NA
## age:factor(d_round)U               NA         NA      NA       NA
## age:ast                     -0.026582   0.007526  -3.532 0.000416 ***
## height:factor(d_round)1     -0.026259   0.022461  -1.169 0.242425
## height:factor(d_round)2     -0.063955   0.024650  -2.595 0.009503 **
## height:factor(d_round)4           NA         NA      NA       NA
## height:factor(d_round)U           NA         NA      NA       NA
## height:reb                  -0.025850   0.004401  -5.873 4.59e-09 ***
## height:ast                   0.018217   0.004967   3.667 0.000248 ***
```

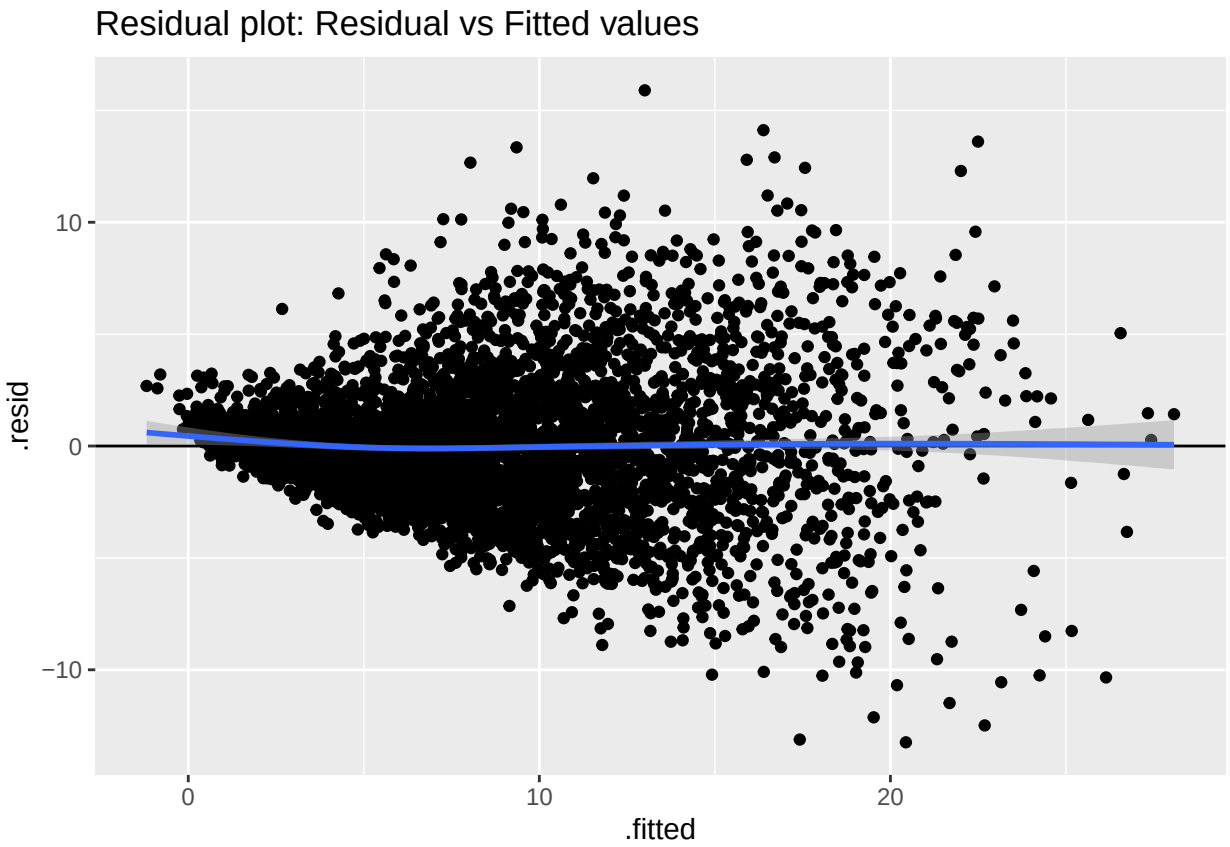
```
## factor(d_round)1:reb        0.544023   0.103009   5.281 1.34e-07 ***
## factor(d_round)2:reb        0.325636   0.109134   2.984 0.002862 **
## factor(d_round)4:reb              NA         NA      NA       NA
## factor(d_round)U:reb              NA         NA      NA       NA
## factor(d_round)1:ast        0.645534   0.132953   4.855 1.24e-06 ***
## factor(d_round)2:ast        0.149744   0.152302   0.983 0.325561
## factor(d_round)4:ast              NA         NA      NA       NA
## factor(d_round)U:ast              NA         NA      NA       NA
## gp:reb                     -0.002225   0.001258  -1.770 0.076848 .
## reb:ast                     0.040211   0.020083   2.002 0.045320 *
## reb:net_rating              0.007029   0.003167   2.220 0.026479 *
## ast:net_rating              0.026520   0.004511   5.879 4.43e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.114 on 4443 degrees of freedom
## Multiple R-squared:  0.7103, Adjusted R-squared:  0.7084
## F-statistic: 375.6 on 29 and 4443 DF,  p-value: < 2.2e-16
```

```
# the higher orders terms with ast were insignificant based on individual
# t-test p-value = 0.074991 > alpha = 0.05 for reb^3

# We conclude that the best model so far is the one with reb^2 and ast^2 for
# higher order.
bestmodelsofar = reducedmodel4
```

Model Diagnostics

```
# Check Linearity Assumption
ggplot(bestmodelsofar, aes(x=.fitted, y=.resid)) +
geom_point() +
geom_hline(yintercept = 0) +
geom_smooth()+
ggtitle("Residual plot: Residual vs Fitted values")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

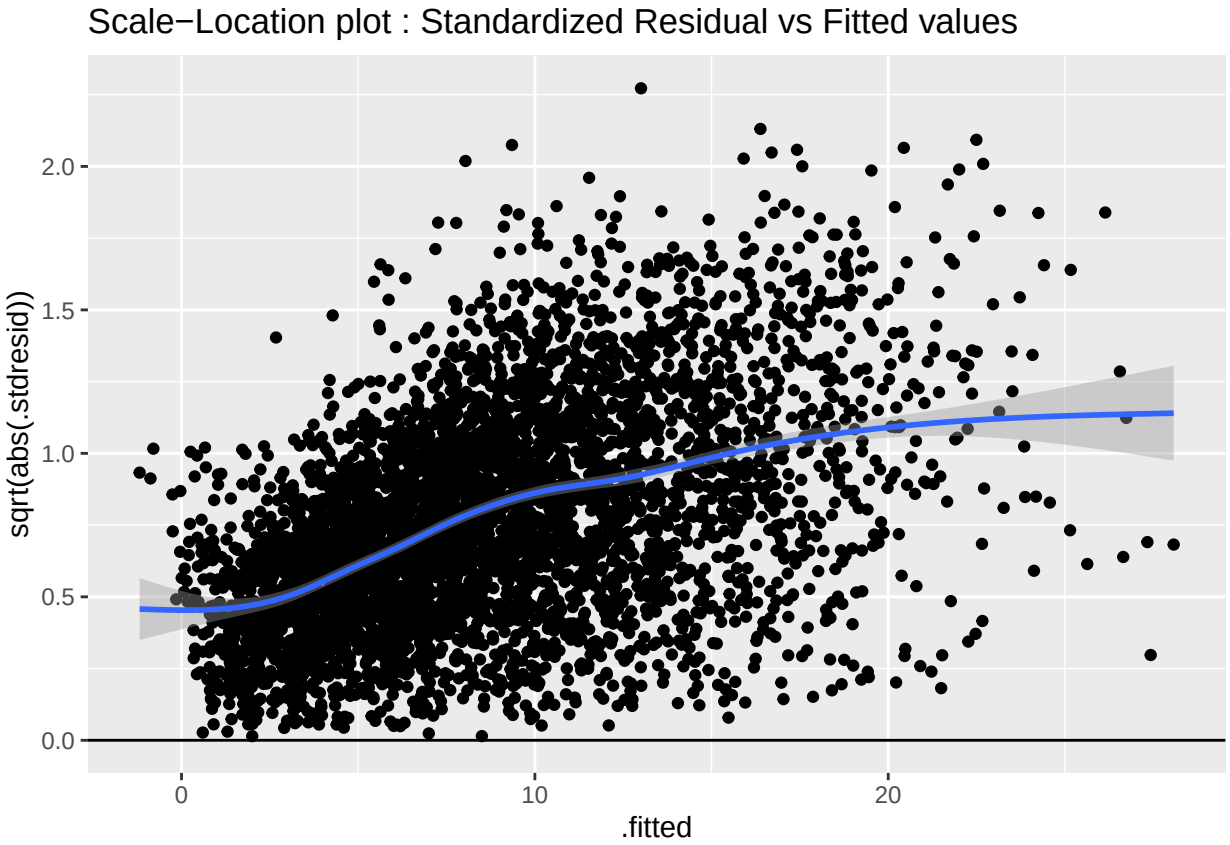



```
# Linearity Assumption Conclusion:
# The Residual vs. Fitted plot shows a clear and distinct straight line
# We conclude that the linearity assumption is met.

# Check Equal Variance Assumption
ggplot(bestmodelsofar, aes(x=.fitted, y=sqrt(abs(.stdresid)))) +
geom_point() +
geom_hline(yintercept = 0) +
geom_smooth()+
ggtitle("Scale-Location plot : Standardized Residual vs Fitted values")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

```
## Warning: Removed 2 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 2 rows containing missing values (geom_point).
```

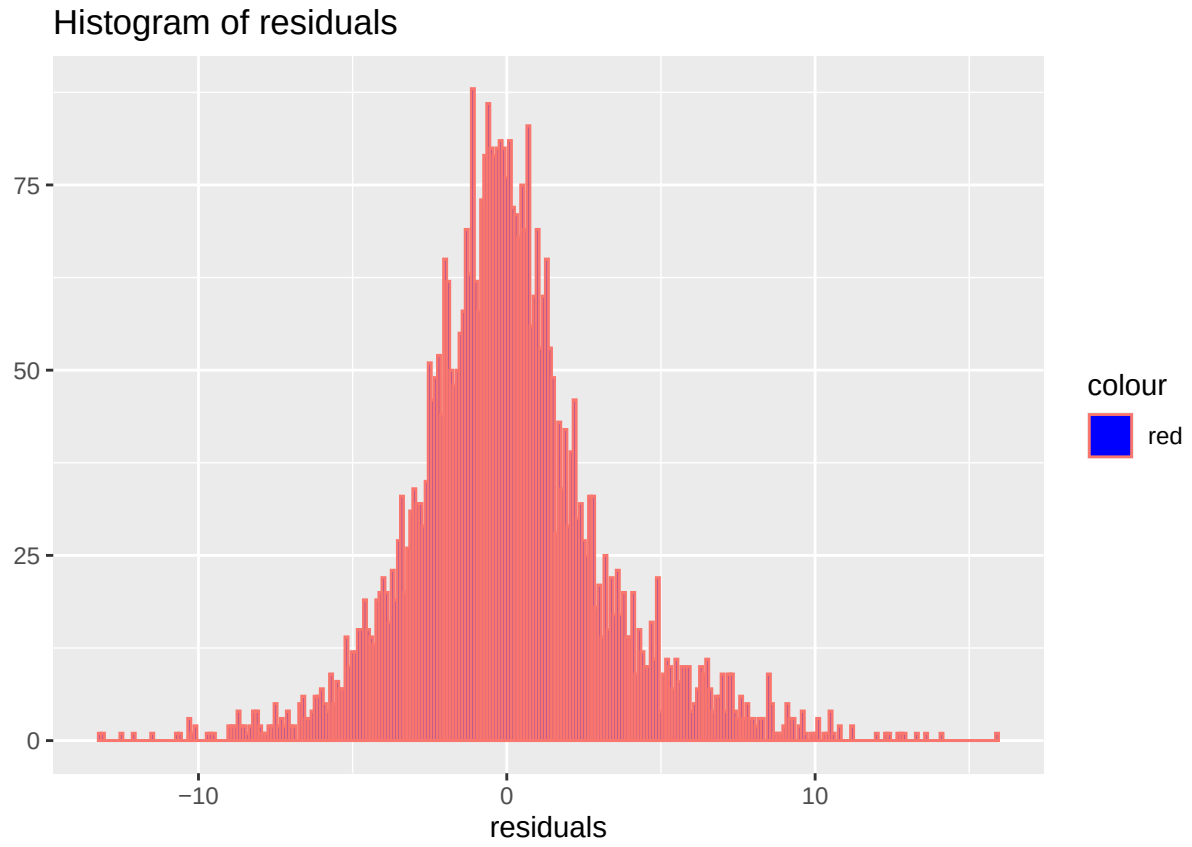



```
# Breusch-Pagan test (for equal variance)
# Ho: heteroscedasticity is not present (homoscedasticity)
# Ha: heteroscedasticity is present

bptest(bestmodelsofar)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  bestmodelsofar
## BP = 806.2, df = 27, p-value < 2.2e-16
```

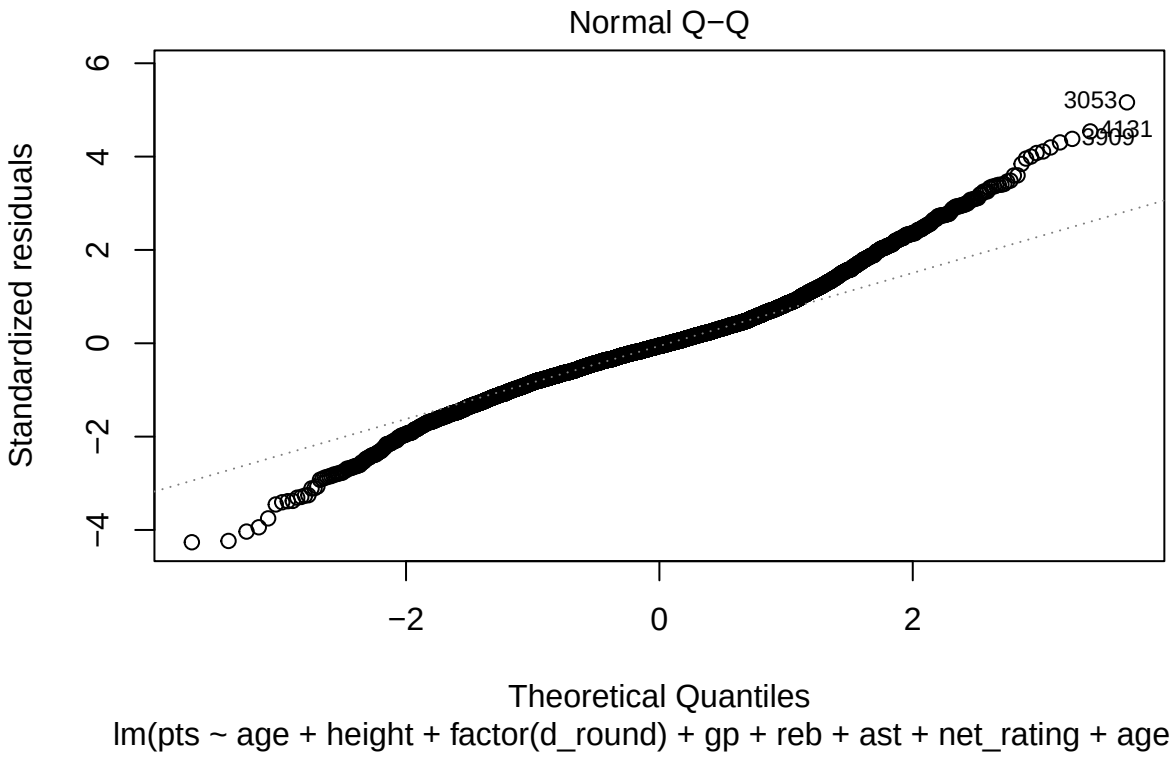
```
# Equal Variance Assumption Conclusion:
# The Residual vs Fitted plot clearly shows a funnel shape, hinting that
# equal variance assumption is not met
# The Scale-Location plot shows a slight funnel shape, but it is hardly
# noticable
# The Breush-Pagan test returns a p-value = 2.2e-16 < alpha = 0.05, strongly
# concluding that the equal variance assumption is not met.


# Check Normality Assumption - histogram and qq plot and Shapiro wilk
qplot(residuals(bestmodelsofar),
geom="histogram",
binwidth = 0.1,
```

```
main = "Histogram of residuals",
xlab = "residuals", color="red",
fill=I("blue"))
```




```
plot(bestmodelsofar, which=2)
```

```
## Warning: not plotting observations with leverage one:
##   692, 4273
```




```
#test normality with Shapiro-Wilk test
#Ho: the sample data are significantly normally distributed
#Ha: the sample data are not significantly normally distributed

shapiro.test(residuals(bestmodelsofar))
```

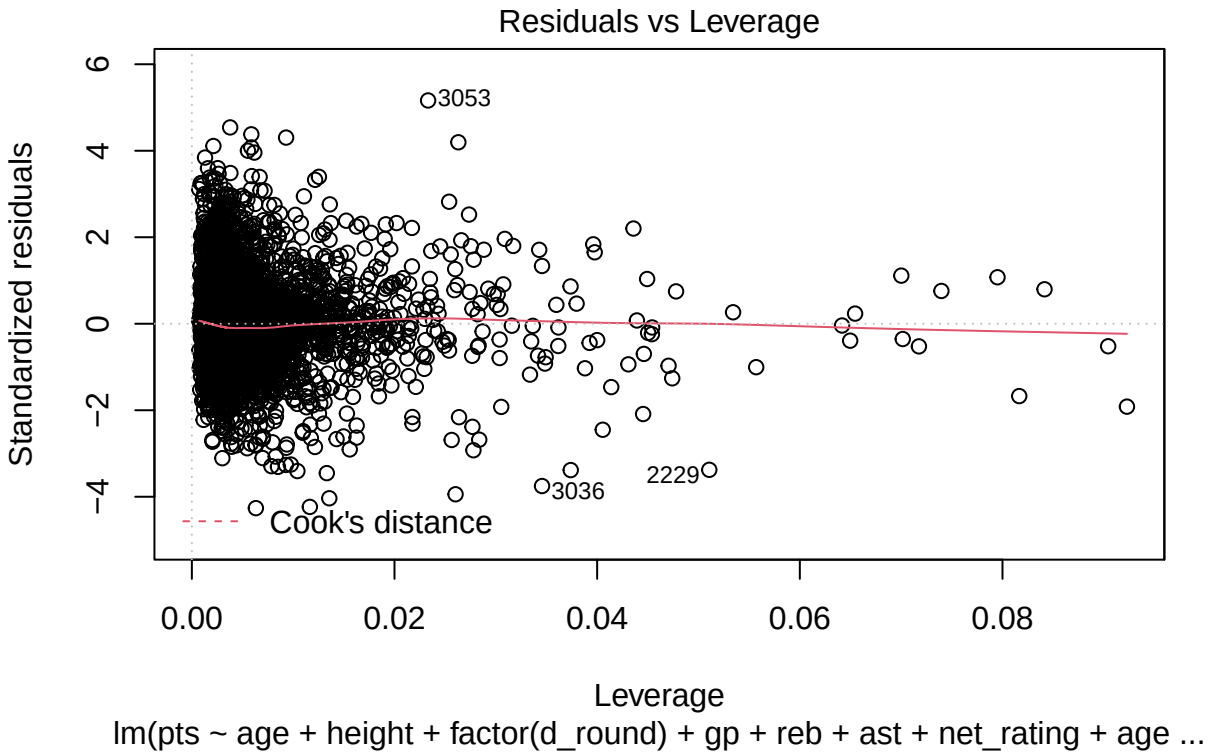
```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(bestmodelsofar)
## W = 0.9735, p-value < 2.2e-16
```

```
# Normality Assumption Conclusion:
# The histogram does hint the normality assumption is met, however, there does
# appear to be multiple peaks and some skewness

# The Normal Q-Q plot shows significant deviation from the diagonal line at
# both tails

# The Shapiro-Wilks test returns a p-value = 2.2e-16 < alpha 0.05, strongly
# concluding that the Normality assumption is not met.
```

Outliers Check

```
# outliers
# 1. Residuals vs Leverage plot
plot(bestmodelsofar,which=5)
```

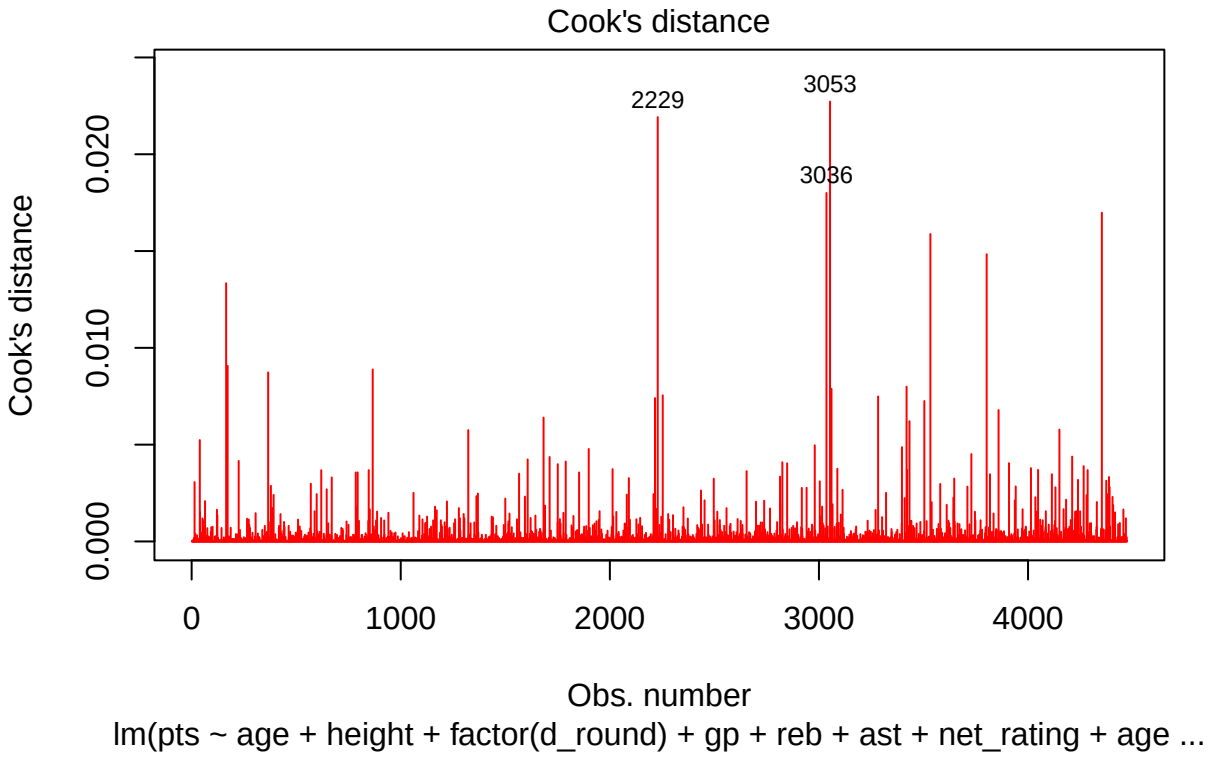
```
## Warning: not plotting observations with leverage one:
##   692, 4273
```




```
# 2. Cook's Distance
bb[cooks.distance(bestmodelsofar)>0.5,]
```

```
##       X player_name team_abbrev age height weight college country d_year
## NA   NA        <NA>        <NA>  NA     NA     NA    <NA>    <NA>   <NA>
## NA.1 NA        <NA>        <NA>  NA     NA     NA    <NA>    <NA>   <NA>
##      d_round d_number gp pts reb ast net_rating oreb_pct dreb_pct usg_pct
## NA      <NA>     <NA> NA  NA  NA  NA         NA       NA       NA      NA
## NA.1    <NA>     <NA> NA  NA  NA  NA         NA       NA       NA      NA
##      ts_pct ast_pct season
## NA       NA      NA   <NA>
## NA.1     NA      NA   <NA>
```

```
plot(bestmodelsofar,pch=18,col="red",which=c(4))
```




```
# 3. Leverage points
lev=hatvalues(bestmodelsofar)
p = length(coef(bestmodelsofar))
n = nrow(bb)
outlier = lev[lev>(2*p/n)]
print(outlier)
```

```
##         40         62         64         70        121        149        158
## 0.02598792 0.04542447 0.08414910 0.02492674 0.07397090 0.04392312 0.01829889
##        165        172        217        225        278        302        306
## 0.09228418 0.04055226 0.01927184 0.03053312 0.02122716 0.02306537 0.01969235
##        366        376        379        392        440        458        499
## 0.02778924 0.04778675 0.02450839 0.02555206 0.03616278 0.01935109 0.02083468
##        516        545        559        567        596        602        670
## 0.03037689 0.02165186 0.01909238 0.01739085 0.01777736 0.02979435 0.04138849
##        687        692        704        720        741        785        847
## 0.02271870 1.00000000 0.03364906 0.06545506 0.03737317 0.07950818 0.01914338
##        864        866        871        898        905        912        922
## 0.02391481 0.08165524 0.03491819 0.01868581 0.02031875 0.01954906 0.01774400
##        927       1008       1074       1086       1119       1163       1164
## 0.01833245 0.01924999 0.01744733 0.02133884 0.09043212 0.01930192 0.04493199
##       1213       1215       1255       1260       1272       1301       1327
## 0.07175438 0.02021224 0.03073338 0.02138233 0.03044808 0.04308071 0.01866106
```

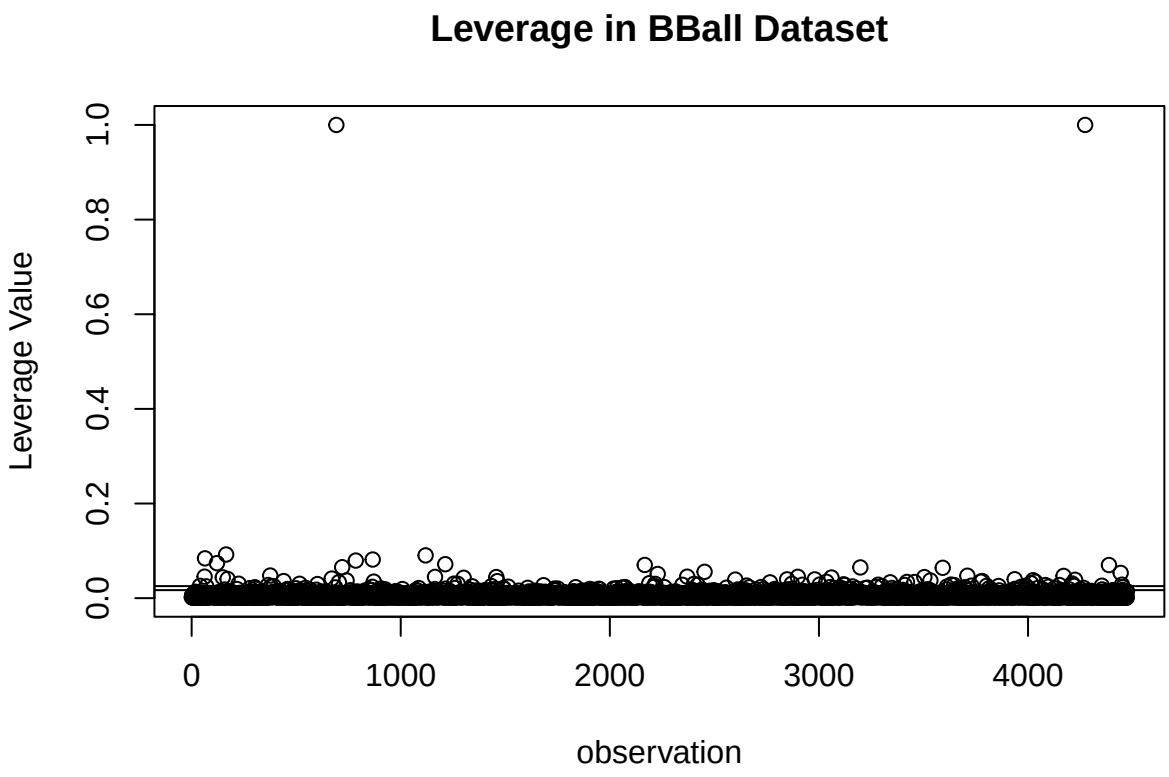
```
##       1341       1434       1435       1439       1456       1457       1462
## 0.02517939 0.02537977 0.01842060 0.01717340 0.01928093 0.04460827 0.03597294
##       1491       1493       1514       1607       1683       1736       1751
## 0.02140693 0.01979451 0.02427237 0.02177508 0.02735407 0.02038716 0.02020925
##       1837       1904       1919       1948       1951       2018       2032
## 0.02317173 0.01943456 0.01764958 0.01987663 0.01879019 0.02022112 0.02093260
##       2058       2071       2073       2167       2189       2209       2216
## 0.02237469 0.02349356 0.02294155 0.07015987 0.03158453 0.02362774 0.02539145
##       2217       2229       2261       2347       2370       2402       2420
## 0.03018004 0.05106689 0.02325251 0.02820546 0.04540163 0.03008829 0.02823354
##       2441       2453       2558       2600       2611       2626       2631
## 0.01821991 0.05569638 0.02210462 0.03925127 0.01817010 0.01805886 0.01752413
##       2654       2666       2697       2724       2732       2766       2786
## 0.02658949 0.02287275 0.01808732 0.02353280 0.01881784 0.03336954 0.01718695
##       2788       2848       2849       2870       2872       2899       2921
## 0.01945303 0.03975862 0.01852780 0.03037896 0.01995187 0.04502930 0.02829988
##       2980       2991       3004       3036       3053       3060       3074
## 0.03961530 0.01726704 0.02881900 0.03456017 0.02331702 0.04358234 0.02143778
##       3097       3105       3113       3118       3122       3147       3153
## 0.01778039 0.02172192 0.01903049 0.02925069 0.02768956 0.01905832 0.01930635
##       3161       3179       3198       3222       3232       3273       3283
## 0.02474353 0.02058531 0.06496959 0.02127460 0.02212849 0.02360690 0.02836128
##       3301       3341       3349       3369       3402       3410       3421
## 0.02534751 0.03348705 0.02159173 0.01814312 0.01745135 0.02781900 0.03428226
##       3457       3504       3518       3529       3533       3592       3611
## 0.03416389 0.04453566 0.01877803 0.01748722 0.03739078 0.06417022 0.01846283
##       3612       3632       3647       3656       3659       3668       3698
## 0.02434630 0.02869473 0.02753849 0.01784939 0.01972441 0.01894405 0.02299183
##       3710       3729       3773       3781       3802       3813       3826
## 0.04740808 0.02636831 0.03484947 0.03617957 0.02602655 0.02047101 0.01788241
##       3859       3936       3937       3941       3956       3966       3985
## 0.02564202 0.04000162 0.01958259 0.01768913 0.01768072 0.02379559 0.01973956
##       3989       4006       4014       4024       4035       4048       4070
## 0.02620886 0.01960563 0.03171260 0.03796950 0.03455690 0.02176095 0.01857266
##       4083       4098       4126       4150       4170       4197       4211
## 0.02766778 0.02420007 0.02083623 0.02769496 0.04704576 0.01854609 0.03088588
##       4216       4221       4225       4251       4266       4273       4353
## 0.02741486 0.02590808 0.03879765 0.01871478 0.02171786 1.00000000 0.02630111
##       4354       4387       4407       4443       4449       4451       4452
## 0.01837414 0.07001612 0.01706260 0.05341404 0.02848674 0.01719696 0.02285979
```

```
outliers_2p_over_n = strtoi(names(outlier))
outlier = lev[lev>(3*p/n)]
print(outlier)
```

```
##         40         62         64        121        149        165        172
## 0.02598792 0.04542447 0.08414910 0.07397090 0.04392312 0.09228418 0.04055226
##        225        366        376        392        440        516        602
## 0.03053312 0.02778924 0.04778675 0.02555206 0.03616278 0.03037689 0.02979435
##        670        692        704        720        741        785        866
## 0.04138849 1.00000000 0.03364906 0.06545506 0.03737317 0.07950818 0.08165524
##        871       1119       1164       1213       1255       1272       1301
## 0.03491819 0.09043212 0.04493199 0.07175438 0.03073338 0.03044808 0.04308071
##       1457       1462       1683       2167       2189       2217       2229
```

```
## 0.04460827 0.03597294 0.02735407 0.07015987 0.03158453 0.03018004 0.05106689
##       2347       2370       2402       2420       2453       2600       2654
## 0.02820546 0.04540163 0.03008829 0.02823354 0.05569638 0.03925127 0.02658949
##       2766       2848       2870       2899       2921       2980       3004
## 0.03336954 0.03975862 0.03037896 0.04502930 0.02829988 0.03961530 0.02881900
##       3036       3060       3118       3122       3198       3283       3341
## 0.03456017 0.04358234 0.02925069 0.02768956 0.06496959 0.02836128 0.03348705
##       3410       3421       3457       3504       3533       3592       3632
## 0.02781900 0.03428226 0.03416389 0.04453566 0.03739078 0.06417022 0.02869473
##       3647       3710       3729       3773       3781       3802       3859
## 0.02753849 0.04740808 0.02636831 0.03484947 0.03617957 0.02602655 0.02564202
##       3936       3989       4014       4024       4035       4083       4150
## 0.04000162 0.02620886 0.03171260 0.03796950 0.03455690 0.02766778 0.02769496
##       4170       4211       4216       4221       4225       4273       4353
## 0.04704576 0.03088588 0.02741486 0.02590808 0.03879765 1.00000000 0.02630111
##       4387       4443       4449
## 0.07001612 0.05341404 0.02848674
```

```
outliers_3p_over_n = strtoi(names(outlier))


plot(rownames(bb),lev, main = "Leverage in BBall Dataset",
xlab="observation",ylab = "Leverage Value")
abline(h = 2 *p/n, lty = 1)
abline(h = 3 *p/n, lty = 1)
```



```r
nrow(bb)
```

```
## [1] 4473
```

```r
# remove outliers for 3p/n remodel based on reducedmodel3
outlier_removed_data3pn <- bb[-c(outliers_3p_over_n), ]
nrow(outlier_removed_data3pn)
```

```
## [1] 4386
```

```r
outlier_removed_model1<-lm(bestmodelsofar$call$formula
                  ,data = outlier_removed_data3pn)
summary(outlier_removed_model1)
```

```
##
## Call:
## lm(formula = bestmodelsofar$call$formula, data = outlier_removed_data3pn)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4367  -1.7662  -0.1719   1.4352  15.7978
##
## Coefficients: (2 not defined because of singularities)
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -37.298515   9.317719  -4.003 6.36e-05 ***
## age                         1.106452   0.318610   3.473 0.000520 ***
## height                      0.176572   0.045325   3.896 9.94e-05 ***
## factor(d_round)2            7.236567   4.063527   1.781 0.075005 .
## factor(d_round)U            2.352562   4.927577   0.477 0.633081
## gp                          0.037900   0.005170   7.331 2.70e-13 ***
## reb                         9.070628   0.790120  11.480  < 2e-16 ***
## ast                        -1.550844   0.969977  -1.599 0.109927
## net_rating                 -0.048136   0.012220  -3.939 8.31e-05 ***
## poly(ast, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2  -0.185127   0.016746 -11.055  < 2e-16 ***
## poly(reb, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(reb, 2, raw = TRUE)2  -0.004147   0.009428  -0.440 0.660075
## age:height                 -0.005534   0.001538  -3.599 0.000323 ***
## age:factor(d_round)2        0.043675   0.029585   1.476 0.139951
## age:factor(d_round)U       -0.050312   0.036319  -1.385 0.166033
## age:ast                    -0.036142   0.008588  -4.209 2.62e-05 ***
## height:factor(d_round)2    -0.039696   0.020224  -1.963 0.049732 *
## height:factor(d_round)U     0.001021   0.024627   0.041 0.966923
## height:reb                 -0.037516   0.004035  -9.299  < 2e-16 ***
## height:ast                  0.027777   0.004418   6.288 3.54e-10 ***
## factor(d_round)2:reb       -0.154746   0.072423  -2.137 0.032679 *
## factor(d_round)U:reb       -0.342039   0.115654  -2.957 0.003119 **
## factor(d_round)2:ast       -0.405336   0.117852  -3.439 0.000589 ***
## factor(d_round)U:ast       -0.851582   0.167916  -5.071 4.11e-07 ***
## gp:reb                     -0.003527   0.001323  -2.666 0.007693 **
## reb:net_rating              0.006885   0.003359   2.050 0.040461 *
## ast:net_rating              0.036122   0.004916   7.348 2.39e-13 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.072 on 4360 degrees of freedom
## Multiple R-squared:  0.7095, Adjusted R-squared:  0.7078
## F-statistic: 425.9 on 25 and 4360 DF,  p-value: < 2.2e-16
```

```
# remove outliers for 2p/n remodel based on reducedmodel3
outlier_removed_data2pn <- bb[-c(outliers_2p_over_n), ]
nrow(outlier_removed_data2pn)
```

```
## [1] 4277
```

```
outlier_removed_model2<-lm(bestmodelsofar$call$formula
                  ,data = outlier_removed_data2pn)
summary(outlier_removed_model2)
```

```
##
## Call:
## lm(formula = bestmodelsofar$call$formula, data = outlier_removed_data2pn)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4839  -1.7609  -0.1913   1.4449  14.4551
##
## Coefficients: (2 not defined because of singularities)
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -37.447113   9.779054  -3.829 0.000130 ***
## age                         1.199378   0.336987   3.559 0.000376 ***
## height                      0.176090   0.047526   3.705 0.000214 ***
## factor(d_round)2            4.924516   4.285302   1.149 0.250553
## factor(d_round)U            2.686023   5.271486   0.510 0.610402
## gp                          0.036683   0.005437   6.747 1.71e-11 ***
## reb                         9.389822   0.828123  11.339  < 2e-16 ***
## ast                        -2.804819   1.063726  -2.637 0.008400 **
## net_rating                 -0.051726   0.013309  -3.886 0.000103 ***
## poly(ast, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2  -0.177615   0.019187  -9.257  < 2e-16 ***
## poly(reb, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(reb, 2, raw = TRUE)2  -0.011521   0.010511  -1.096 0.273093
## age:height                 -0.005954   0.001624  -3.666 0.000249 ***
## age:factor(d_round)2        0.037242   0.030033   1.240 0.215022
## age:factor(d_round)U       -0.038890   0.039759  -0.978 0.328056
## age:ast                    -0.039717   0.009319  -4.262 2.07e-05 ***
## height:factor(d_round)2    -0.026895   0.021347  -1.260 0.207788
## height:factor(d_round)U    -0.002100   0.026443  -0.079 0.936707
## height:reb                 -0.038893   0.004227  -9.201  < 2e-16 ***
## height:ast                  0.034345   0.004853   7.078 1.71e-12 ***
## factor(d_round)2:reb       -0.184566   0.077738  -2.374 0.017630 *
## factor(d_round)U:reb       -0.335494   0.127003  -2.642 0.008281 **
## factor(d_round)2:ast       -0.441889   0.131433  -3.362 0.000780 ***
## factor(d_round)U:ast       -0.877617   0.186332  -4.710 2.56e-06 ***
## gp:reb                     -0.002954   0.001420  -2.081 0.037538 *
```

```
## reb:net_rating              0.005826   0.003564   1.635 0.102185
## ast:net_rating              0.039465   0.005432   7.265 4.41e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.063 on 4251 degrees of freedom
## Multiple R-squared:  0.7048, Adjusted R-squared:  0.703
## F-statistic: 405.9 on 25 and 4251 DF,  p-value: < 2.2e-16
```

```
# these models produced lower RMSE but also lower R^2_adj,
# so its inconclusive whether they are better,
# by checking our data manually, we noticed that at h_i > 3p/n
# a couple of erronious data were removed (d_round = 0 and d_round = 4)
# and at h_i > 2p/n, no additional bad data were removed,
# so we use the outlier_removed_model1, that is where h_i > 3p/n

# we reduce outlier_removed_model1 by taking out insignificant terms
# which are age:factor(d_round) and second order term for reb
reducedmodel5<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating
                  +age:height+age:ast
                  +height:factor(d_round)+height:reb+height:ast
                  +factor(d_round):reb+factor(d_round):ast
                  +gp:reb+reb:net_rating+ast:net_rating
                  +poly(ast,2,raw=TRUE)
                  ,data = outlier_removed_data3pn)
summary(reducedmodel5)
```

```
##
## Call:
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating + age:height + age:ast + height:factor(d_round) +
##     height:reb + height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:net_rating + ast:net_rating + poly(ast, 2, raw = TRUE),
##     data = outlier_removed_data3pn)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4676  -1.7593  -0.1878   1.4354  15.7779
##
## Coefficients: (1 not defined because of singularities)
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 -36.260017   8.928900  -4.061 4.97e-05 ***
## age                           1.026953   0.308379   3.330 0.000875 ***
## height                        0.171658   0.043528   3.944 8.15e-05 ***
## factor(d_round)2              8.196167   3.969824   2.065 0.039018 *
## factor(d_round)U              1.827083   4.798105   0.381 0.703376
## gp                            0.038629   0.004831   7.996 1.64e-15 ***
## reb                           9.296627   0.695161  13.373  < 2e-16 ***
## ast                          -1.538839   0.964463  -1.596 0.110664
## net_rating                   -0.046039   0.012088  -3.809 0.000142 ***
## poly(ast, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2  -0.186222   0.016572 -11.237  < 2e-16 ***
## age:height                 -0.005142   0.001498  -3.433 0.000603 ***
## age:ast                    -0.034847   0.008309  -4.194 2.80e-05 ***
```

```
## height:factor(d_round)2    -0.039116   0.019827  -1.973 0.048570 *
## height:factor(d_round)U    -0.002980   0.024102  -0.124 0.901612
## height:reb                 -0.038797   0.003361 -11.544  < 2e-16 ***
## height:ast                  0.027586   0.004397   6.274 3.86e-10 ***
## factor(d_round)2:reb       -0.140478   0.070132  -2.003 0.045232 *
## factor(d_round)U:reb       -0.341468   0.112270  -3.042 0.002368 **
## factor(d_round)2:ast       -0.379303   0.116022  -3.269 0.001087 **
## factor(d_round)U:ast       -0.887396   0.165708  -5.355 8.99e-08 ***
## gp:reb                     -0.003681   0.001259  -2.924 0.003477 **
## reb:net_rating              0.006412   0.003324   1.929 0.053771 .
## ast:net_rating              0.036073   0.004916   7.338 2.57e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.073 on 4363 degrees of freedom
## Multiple R-squared:  0.7091, Adjusted R-squared:  0.7077
## F-statistic: 483.5 on 22 and 4363 DF,  p-value: < 2.2e-16
```

```
anova(reducedmodel5, outlier_removed_model1)
```

```
## Analysis of Variance Table
##
## Model 1: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:height + age:ast + height:factor(d_round) + height:reb +
##     height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:net_rating + ast:net_rating + poly(ast, 2, raw = TRUE)
## Model 2: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:height + age:factor(d_round) + age:ast + height:factor(d_round) +
##     height:reb + height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:net_rating + ast:net_rating + poly(ast, 2, raw = TRUE) +
##     poly(reb, 2, raw = TRUE)
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1   4363 41206
## 2   4360 41157  3    49.604 1.7516 0.1542
```

```
# Partial F test on outlier_removed_model1 (reducedmodel4 with high leverage
# data points removed) and reducedmodel5 (reducedmodel4 with high leverage data
# points removed, age:factor(d_round) interaction removed, and reb^2 removed)

# Ho: Beta_age:factor(d_round) = Beta_reb^2 = 0
# Ha: Beta_age:factor(d_round) = Beta_reb^2 != 0

# The Partial F test gives a p-value = 0.1542 > alpha = 0.05, therefore we fail
# to reject Ho and conclude that we should have dropped the
# age:factor(d_round) interaction and reb^2 variables from the model.


# we continue reduce reducedmodel5 by taking out insignificant terms
# which is reb:net_rating
reducedmodel6<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating
                  +age:height+age:ast
                  +height:factor(d_round)+height:reb+height:ast
                  +factor(d_round):reb+factor(d_round):ast
```

```
                  +gp:reb+ast:net_rating
                  +poly(ast,2,raw=TRUE)
                  ,data = outlier_removed_data3pn)
summary(reducedmodel6)
```

```
##
## Call:
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating + age:height + age:ast + height:factor(d_round) +
##     height:reb + height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + ast:net_rating + poly(ast, 2, raw = TRUE), data = outlier_removed_data3pn)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4857  -1.7797  -0.1712   1.4421  15.7386
##
## Coefficients: (1 not defined because of singularities)
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -33.900547   8.847499  -3.832 0.000129 ***
## age                         0.966288   0.306867   3.149 0.001650 **
## height                      0.160352   0.043145   3.717 0.000204 ***
## factor(d_round)2            8.138098   3.970947   2.049 0.040482 *
## factor(d_round)U            1.629202   4.798504   0.340 0.734232
## gp                          0.036703   0.004728   7.762 1.03e-14 ***
## reb                         9.130284   0.690008  13.232  < 2e-16 ***
## ast                        -1.636815   0.963425  -1.699 0.089399 .
## net_rating                 -0.031769   0.009564  -3.322 0.000902 ***
## poly(ast, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2  -0.186830   0.016574 -11.272  < 2e-16 ***
## age:height                 -0.004835   0.001490  -3.246 0.001180 **
## age:ast                    -0.034759   0.008311  -4.182 2.94e-05 ***
## height:factor(d_round)2    -0.038816   0.019832  -1.957 0.050383 .
## height:factor(d_round)U    -0.001933   0.024104  -0.080 0.936093
## height:reb                 -0.038143   0.003345 -11.404  < 2e-16 ***
## height:ast                  0.028119   0.004390   6.406 1.65e-10 ***
## factor(d_round)2:reb       -0.140284   0.070153  -2.000 0.045597 *
## factor(d_round)U:reb       -0.348235   0.112250  -3.102 0.001932 **
## factor(d_round)2:ast       -0.381016   0.116055  -3.283 0.001035 **
## factor(d_round)U:ast       -0.880877   0.165725  -5.315 1.12e-07 ***
## gp:reb                     -0.003162   0.001230  -2.570 0.010198 *
## ast:net_rating              0.039486   0.004588   8.607  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.074 on 4364 degrees of freedom
## Multiple R-squared:  0.7089, Adjusted R-squared:  0.7075
## F-statistic: 506.1 on 21 and 4364 DF,  p-value: < 2.2e-16
```

```
anova(reducedmodel6, reducedmodel5)
```

```
## Analysis of Variance Table
##
## Model 1: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
```

```
##     age:height + age:ast + height:factor(d_round) + height:reb +
##     height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + ast:net_rating + poly(ast, 2, raw = TRUE)
## Model 2: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:height + age:ast + height:factor(d_round) + height:reb +
##     height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + reb:net_rating + ast:net_rating + poly(ast, 2, raw = TRUE)
##   Res.Df   RSS Df Sum of Sq      F  Pr(>F)
## 1   4364 41242
## 2   4363 41206  1    35.151 3.7218 0.05377 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Partial F test on reducedmodel5 and
# reducedmodel6 (reducedmodel5 with reb:net_rating interaction removed)

# Ho: Beta_reb:net_rating = 0
# Ha: Beta_reb:net_rating != 0

# The Partial F test gives a p-value = 0.05377   > alpha = 0.05, therefore we fail
# to reject Ho and conclude that we should have dropped the
# reb:net_rating interaction variable from the model.

# we continue reduce reducedmodel6 by taking out insignificant terms
# which is height:factor(d_round)
reducedmodel7<-lm(pts~age+height+factor(d_round)+gp+reb+ast+net_rating
                  +age:height+age:ast
                  +height:reb+height:ast
                  +factor(d_round):reb+factor(d_round):ast
                  +gp:reb+ast:net_rating
                  +poly(ast,2,raw=TRUE)
                  ,data = outlier_removed_data3pn)
summary(reducedmodel7)
```

```
##
## Call:
## lm(formula = pts ~ age + height + factor(d_round) + gp + reb +
##     ast + net_rating + age:height + age:ast + height:reb + height:ast +
##     factor(d_round):reb + factor(d_round):ast + gp:reb + ast:net_rating +
##     poly(ast, 2, raw = TRUE), data = outlier_removed_data3pn)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4604  -1.7614  -0.1674   1.4346  15.9444
##
## Coefficients: (1 not defined because of singularities)
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -29.870791   8.486351  -3.520 0.000436 ***
## age                         0.917821   0.306000   2.999 0.002720 **
## height                      0.140271   0.041299   3.396 0.000689 ***
## factor(d_round)2            0.385216   0.248093   1.553 0.120565
## factor(d_round)U            1.207586   0.328451   3.677 0.000239 ***
## gp                          0.037192   0.004717   7.885 3.95e-15 ***
## reb                         9.003769   0.685729  13.130  < 2e-16 ***
```

```
## ast                         -1.784756   0.937595  -1.904 0.057035 .
## net_rating                  -0.031790   0.009566  -3.323 0.000898 ***
## poly(ast, 2, raw = TRUE)1         NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2  -0.183352   0.016198 -11.320  < 2e-16 ***
## age:height                  -0.004595   0.001485  -3.094 0.001989 **
## age:ast                     -0.034682   0.008312  -4.172 3.07e-05 ***
## height:reb                  -0.037403   0.003312 -11.292  < 2e-16 ***
## height:ast                   0.028594   0.004289   6.667 2.95e-11 ***
## factor(d_round)2:reb        -0.222278   0.056599  -3.927 8.72e-05 ***
## factor(d_round)U:reb        -0.340669   0.092859  -3.669 0.000247 ***
## factor(d_round)2:ast        -0.236133   0.089624  -2.635 0.008450 **
## factor(d_round)U:ast        -0.887044   0.119510  -7.422 1.38e-13 ***
## gp:reb                      -0.003262   0.001228  -2.658 0.007899 **
## ast:net_rating               0.039601   0.004589   8.630  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.075 on 4366 degrees of freedom
## Multiple R-squared:  0.7086, Adjusted R-squared:  0.7074
## F-statistic: 558.8 on 19 and 4366 DF,  p-value: < 2.2e-16
```

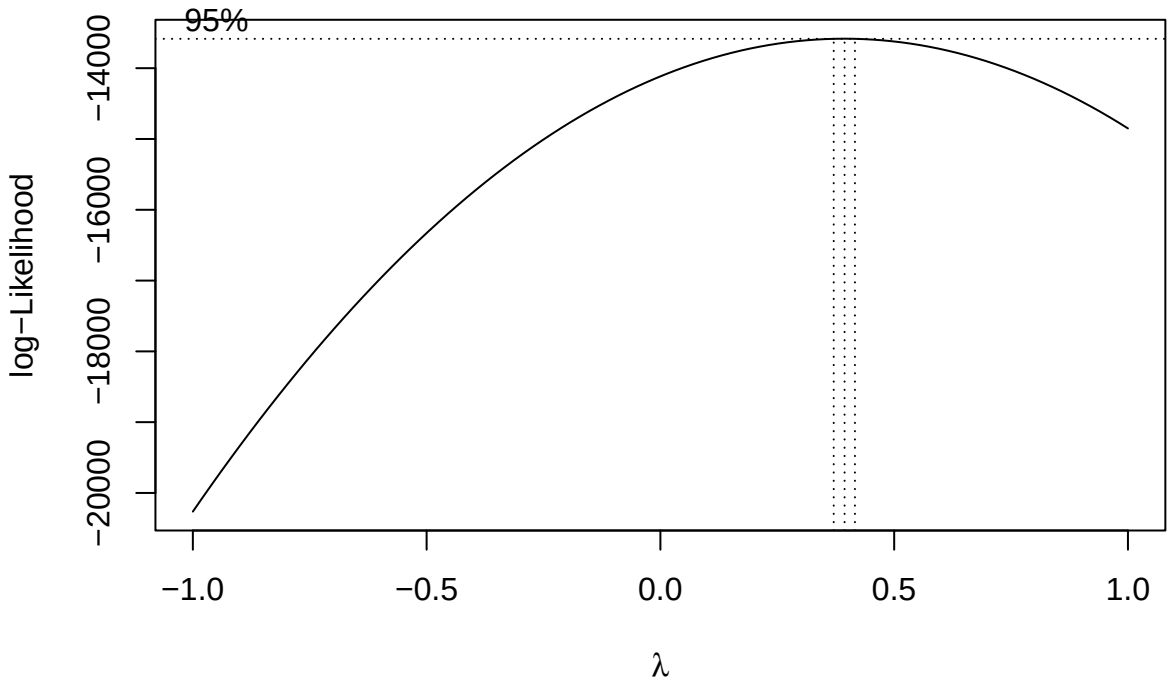
```
anova(reducedmodel7, reducedmodel6)
```

```
## Analysis of Variance Table
##
## Model 1: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:height + age:ast + height:reb + height:ast + factor(d_round):reb +
##     factor(d_round):ast + gp:reb + ast:net_rating + poly(ast,
##     2, raw = TRUE)
## Model 2: pts ~ age + height + factor(d_round) + gp + reb + ast + net_rating +
##     age:height + age:ast + height:factor(d_round) + height:reb +
##     height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + ast:net_rating + poly(ast, 2, raw = TRUE)
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1   4366 41280
## 2   4364 41242  2    38.692 2.0471 0.1292
```

```
# Partial F test on reducedmodel6 and reducedmodel7 (reducedmodel6 with
# height:factor(d_round) interaction removed)

# Ho: Beta_reb:height:factor(d_round) = 0
# Ha: Beta_reb:height:factor(d_round) != 0

# The Partial F test gives a p-value = 0.1292   < alpha = 0.05, therefore we
# fail to reject Ho and conclude that we should have dropped the
# height:factor(d_round) interaction variable from the model.

# we will use reducedmodel7

bestmodelsofar = reducedmodel7
```

Box-Cox Transformation

```
# Box-Cox
bc=boxcox(bestmodelsofar,lambda=seq(-1,1))
```




```
#extract best lambda
bestlambda=bc$x[which(bc$y==max(bc$y))]
print(bestlambda)
```

```
## [1] 0.3939394
```

```
bcmodel=lm((((pts^bestlambda)-1)/bestlambda)~age+height+factor(d_round)
           +gp+reb+ast+net_rating
           +age:height+age:ast
           +height:reb+height:ast
           +factor(d_round):reb+factor(d_round):ast
           +gp:reb+ast:net_rating
           +poly(ast,2,raw=TRUE)
           ,data = outlier_removed_data3pn)

summary(bcmodel)
```

```
##
```

```
## Call:
## lm(formula = (((pts^bestlambda) - 1)/bestlambda) ~ age + height +
##     factor(d_round) + gp + reb + ast + net_rating + age:height +
##     age:ast + height:reb + height:ast + factor(d_round):reb +
##     factor(d_round):ast + gp:reb + ast:net_rating + poly(ast,
##     2, raw = TRUE), data = outlier_removed_data3pn)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7094 -0.5038 -0.0023  0.4959  2.8354
##
## Coefficients: (1 not defined because of singularities)
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -6.9465320  2.1320363  -3.258 0.001130 **
## age                         0.2771059  0.0768767   3.605 0.000316 ***
## height                      0.0344253  0.0103755   3.318 0.000914 ***
## factor(d_round)2           -0.1579856  0.0623287  -2.535 0.011288 *
## factor(d_round)U            0.0227970  0.0825172   0.276 0.782354
## gp                          0.0223975  0.0011850  18.900  < 2e-16 ***
## reb                         2.2663077  0.1722766  13.155  < 2e-16 ***
## ast                        -0.2606644  0.2355531  -1.107 0.268525
## net_rating                  0.0006082  0.0024034   0.253 0.800221
## poly(ast, 2, raw = TRUE)1          NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2 -0.0576781  0.0040693 -14.174  < 2e-16 ***
## age:height                 -0.0014276  0.0003732  -3.826 0.000132 ***
## age:ast                    -0.0055966  0.0020883  -2.680 0.007390 **
## height:reb                 -0.0086028  0.0008322 -10.338  < 2e-16 ***
## height:ast                  0.0059891  0.0010776   5.558 2.89e-08 ***
## factor(d_round)2:reb       -0.0030727  0.0142194  -0.216 0.828924
## factor(d_round)U:reb       -0.0204795  0.0233290  -0.878 0.380070
## factor(d_round)2:ast       -0.0242163  0.0225163  -1.076 0.282210
## factor(d_round)U:ast       -0.1735665  0.0300246  -5.781 7.95e-09 ***
## gp:reb                     -0.0032767  0.0003084 -10.624  < 2e-16 ***
## ast:net_rating              0.0036978  0.0011528   3.208 0.001348 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7725 on 4366 degrees of freedom
## Multiple R-squared:  0.7454, Adjusted R-squared:  0.7443
## F-statistic: 672.9 on 19 and 4366 DF,  p-value: < 2.2e-16
```

```
# bcmodel has the highest r-squared adjusted and lowest RMSE, it is the best model so far

# factor(d_round):reb becomes insignificant as shown in the individual t test,
# we take that out and do f-test to make sure
reducedbcmodel=lm((((pts^bestlambda)-1)/bestlambda)~age+height+factor(d_round)
                  +gp+reb+ast+net_rating
                  +age:height+age:ast
                  +height:reb+height:ast
                  +factor(d_round):ast
                  +gp:reb+ast:net_rating
                  +poly(ast,2,raw=TRUE)
                  ,data = outlier_removed_data3pn)
summary(reducedbcmodel)
```

```
##
## Call:
## lm(formula = (((pts^bestlambda) - 1)/bestlambda) ~ age + height +
##     factor(d_round) + gp + reb + ast + net_rating + age:height +
##     age:ast + height:reb + height:ast + factor(d_round):ast +
##     gp:reb + ast:net_rating + poly(ast, 2, raw = TRUE), data = outlier_removed_data3pn)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6971 -0.5034 -0.0026  0.4952  2.8364
##
## Coefficients: (1 not defined because of singularities)
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -6.8869261  2.1293357  -3.234 0.001228 **
## age                         0.2807468  0.0767522   3.658 0.000257 ***
## height                      0.0342418  0.0103685   3.302 0.000966 ***
## factor(d_round)2           -0.1696891  0.0464185  -3.656 0.000260 ***
## factor(d_round)U           -0.0317125  0.0544026  -0.583 0.559976
## gp                          0.0221654  0.0011515  19.250  < 2e-16 ***
## reb                         2.2314618  0.1668954  13.370  < 2e-16 ***
## ast                        -0.2792760  0.2344750  -1.191 0.233691
## net_rating                  0.0005540  0.0024022   0.231 0.817612
## poly(ast, 2, raw = TRUE)1          NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2 -0.0575766  0.0040671 -14.157  < 2e-16 ***
## age:height                -0.0014460  0.0003725  -3.882 0.000105 ***
## age:ast                   -0.0056161  0.0020878  -2.690 0.007174 **
## height:reb                -0.0084595  0.0008137 -10.396  < 2e-16 ***
## height:ast                 0.0060871  0.0010711   5.683 1.41e-08 ***
## factor(d_round)2:ast      -0.0245812  0.0220997  -1.112 0.266076
## factor(d_round)U:ast      -0.1742997  0.0300087  -5.808 6.76e-09 ***
## gp:reb                    -0.0032200  0.0003012 -10.690  < 2e-16 ***
## ast:net_rating             0.0037385  0.0011514   3.247 0.001176 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7724 on 4368 degrees of freedom
## Multiple R-squared:  0.7454, Adjusted R-squared:  0.7444
## F-statistic: 752.3 on 17 and 4368 DF,  p-value: < 2.2e-16
```

```
anova(reducedbcmodel, bcmodel)
```

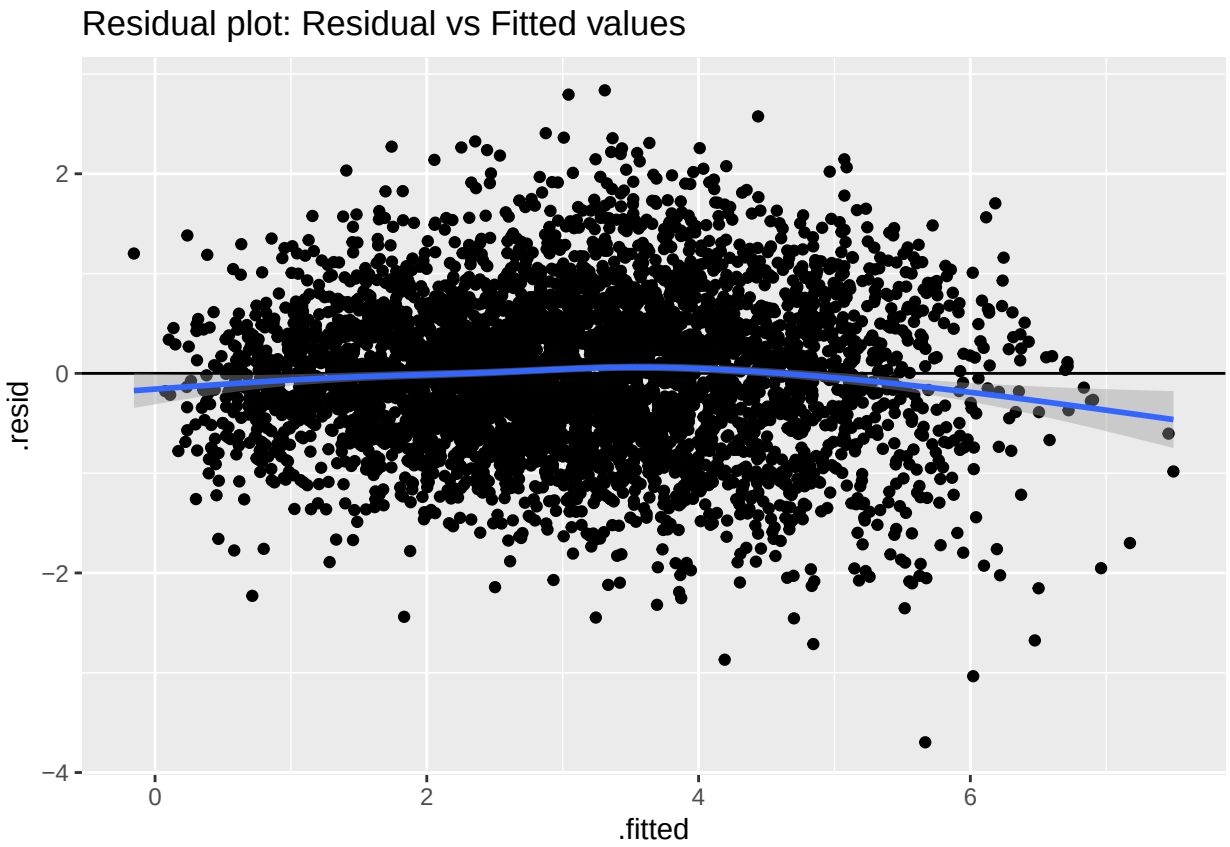
```
## Analysis of Variance Table
##
## Model 1: (((pts^bestlambda) - 1)/bestlambda) ~ age + height + factor(d_round) +
##     gp + reb + ast + net_rating + age:height + age:ast + height:reb +
##     height:ast + factor(d_round):ast + gp:reb + ast:net_rating +
##     poly(ast, 2, raw = TRUE)
## Model 2: (((pts^bestlambda) - 1)/bestlambda) ~ age + height + factor(d_round) +
##     gp + reb + ast + net_rating + age:height + age:ast + height:reb +
##     height:ast + factor(d_round):reb + factor(d_round):ast +
##     gp:reb + ast:net_rating + poly(ast, 2, raw = TRUE)
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1   4368 2605.9
## 2   4366 2605.5  2   0.46111 0.3863 0.6796
```

```
# f test shows that it's good to remove that interaction

# Partial F test on bcmodel (Box-Cox transformed model) and
# reducedbcmodel (Box-Cox transformed model with factor(d_round):reb interaction dropped)

# Ho: Beta_factor(d_round):reb = 0
# Ha: Beta_factor(d_round):reb != 0

# The Partial F test gives a p-value = 0.6796 > alpha = 0.05, therefore we fail
# to reject Ho and conclude all that we should have dropped the factor(d_round):reb
# interaction from the model.
# all good with taking that out

bestmodelsofar = reducedbcmodel
```

Model Diagnostics on Box-Cox Transform model

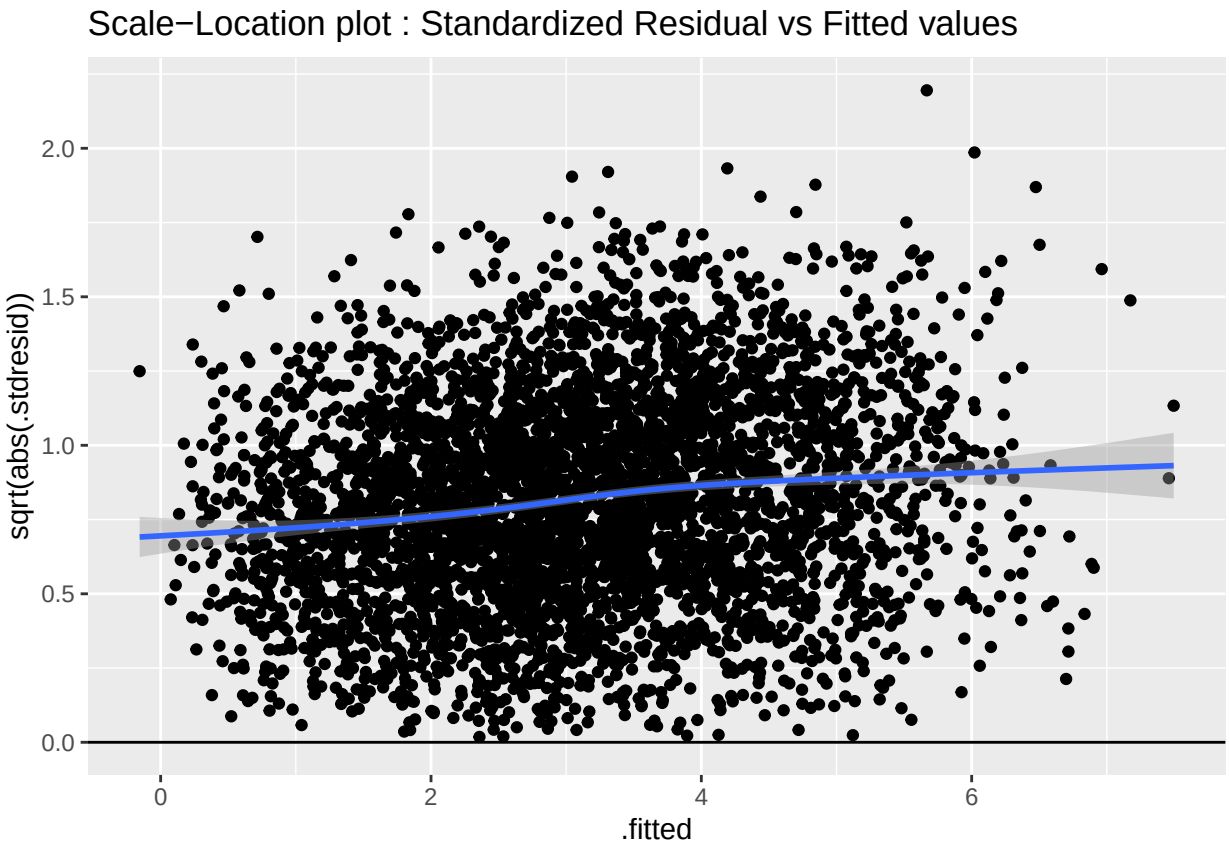
```
# Linearity Assumption check
ggplot(bestmodelsofar, aes(x=.fitted, y=.resid)) +
geom_point() +
geom_hline(yintercept = 0) +
geom_smooth()+
ggtitle("Residual plot: Residual vs Fitted values")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




```
# Linearity Assumption Conclusion:
# The Residual vs. Fitted plot shows a straight line, meaning
# the linearity assumption is met.

# Equal Variance Assumption Check
ggplot(bestmodelsofar, aes(x=.fitted, y=sqrt(abs(.stdresid)))) +
geom_point() +
geom_hline(yintercept = 0) +
geom_smooth()+
ggtitle("Scale-Location plot : Standardized Residual vs Fitted values")
```

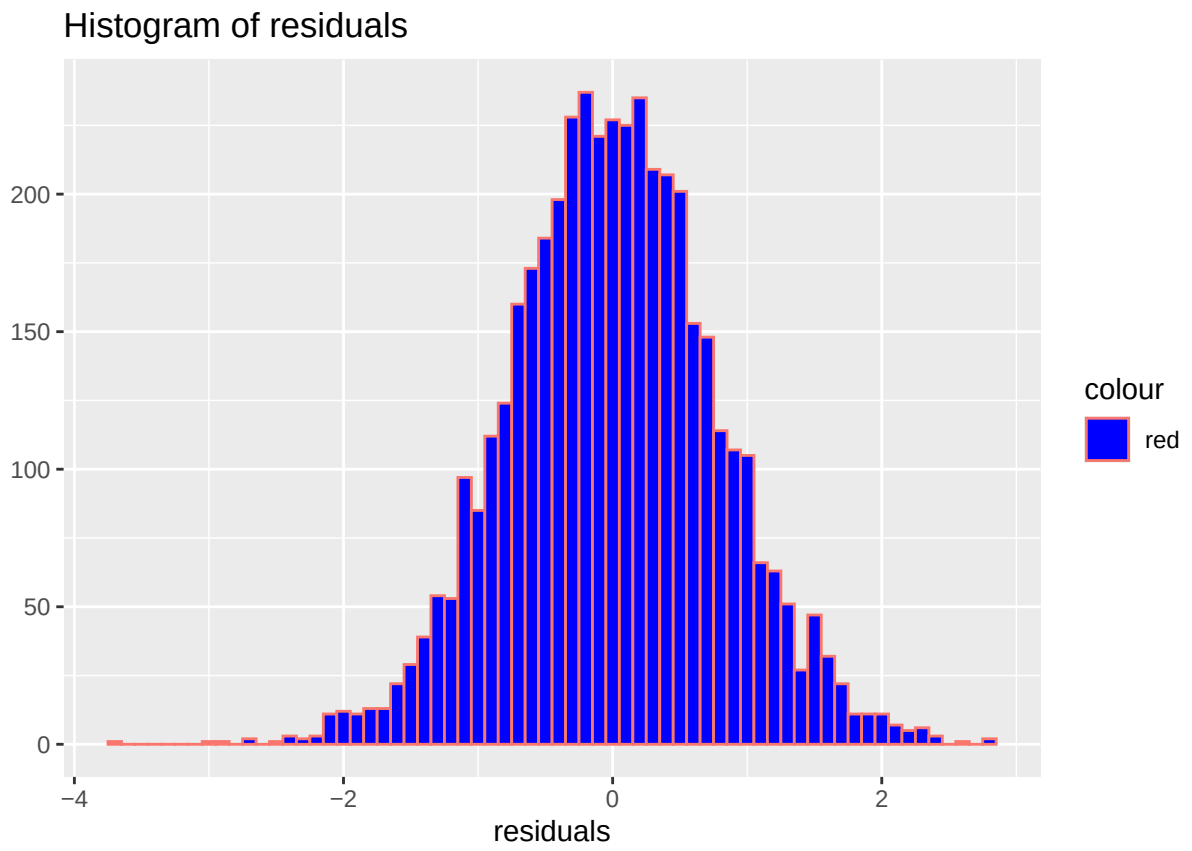
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



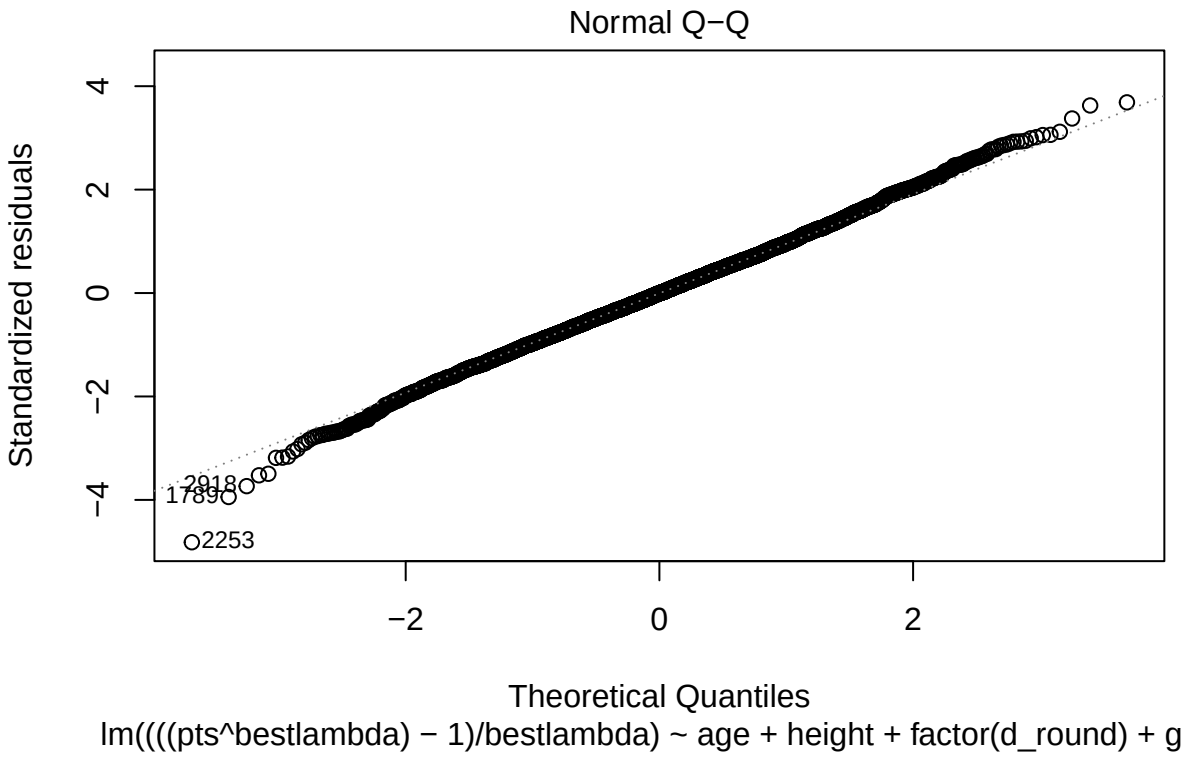

```
# Breusch-Pagan test (for equal variance)
# Ho: heteroscedasticity is not present (homoscedasticity)
# Ha: heteroscedasticity is present
bptest(bestmodelsofar)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  bestmodelsofar
## BP = 253.71, df = 17, p-value < 2.2e-16
```

```
# Equal Variance Assumption Conclusion:
# The Residual vs. Fitted and Scale-Location plot both don't show any
# evidence of a pattern, hinting that the Equal Variance assumption is met.
# However, the Breusch-Pagan test returns a p-value = 2.2e-16 < alpha = 0.05,
# strongly concluding that the Equal Variance assumption is not met.

# Check Normality Assumption - histogram and qq plot and Shapiro wilk
qplot(residuals(bestmodelsofar),
geom="histogram",
binwidth = 0.1,
main = "Histogram of residuals",
xlab = "residuals", color="red",
fill=I("blue"))
```




```
plot(bestmodelsofar, which=2)
```




```
#test normality with Shapiro-Wilk test
#Ho: the sample data are significantly normally distributed
#Ha: the sample data are not significantly normally distributed

shapiro.test(residuals(bestmodelsofar))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(bestmodelsofar)
## W = 0.99873, p-value = 0.001681
```

```
# Normality Assumption Conclusion:
# The histogram does hint the normality assumption is met and definitely shows
# improvement over the untransformed model.

# The Normal Q-Q plot shows significant improvement that strongly lines up
# along the diagonal line with no hint of S-shape or bow-shape, hinting that
# normality is met.

# The Shapiro-Wilks test returns a p-value = 0.001681 < alpha 0.05,concluding
# that the Normality assumption is not met. It is important to note that the
# Box-Cox transform we applied did significantly improve the Shapiro-Wilk
# test p-value, increasing from 2.2e-16 (untransformed model)
# to 0.001681 (Box-Cox transformed model)
```

Final Model & Predictive Tests

```r
# final models
finalmodel = bestmodelsofar
summary(finalmodel)
```

```
##
## Call:
## lm(formula = (((pts^bestlambda) - 1)/bestlambda) ~ age + height +
##     factor(d_round) + gp + reb + ast + net_rating + age:height +
##     age:ast + height:reb + height:ast + factor(d_round):ast +
##     gp:reb + ast:net_rating + poly(ast, 2, raw = TRUE), data = outlier_removed_data3pn)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6971 -0.5034 -0.0026  0.4952  2.8364
##
## Coefficients: (1 not defined because of singularities)
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 -6.8869261  2.1293357  -3.234 0.001228 **
## age                          0.2807468  0.0767522   3.658 0.000257 ***
## height                       0.0342418  0.0103685   3.302 0.000966 ***
## factor(d_round)2            -0.1696891  0.0464185  -3.656 0.000260 ***
## factor(d_round)U            -0.0317125  0.0544026  -0.583 0.559976
## gp                           0.0221654  0.0011515  19.250  < 2e-16 ***
## reb                          2.2314618  0.1668954  13.370  < 2e-16 ***
## ast                         -0.2792760  0.2344750  -1.191 0.233691
## net_rating                   0.0005540  0.0024022   0.231 0.817612
## poly(ast, 2, raw = TRUE)1           NA         NA      NA       NA
## poly(ast, 2, raw = TRUE)2 -0.0575766  0.0040671 -14.157  < 2e-16 ***
## age:height                -0.0014460  0.0003725  -3.882 0.000105 ***
## age:ast                   -0.0056161  0.0020878  -2.690 0.007174 **
## height:reb                -0.0084595  0.0008137 -10.396  < 2e-16 ***
## height:ast                 0.0060871  0.0010711   5.683 1.41e-08 ***
## factor(d_round)2:ast      -0.0245812  0.0220997  -1.112 0.266076
## factor(d_round)U:ast      -0.1742997  0.0300087  -5.808 6.76e-09 ***
## gp:reb                    -0.0032200  0.0003012 -10.690  < 2e-16 ***
## ast:net_rating             0.0037385  0.0011514   3.247 0.001176 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7724 on 4368 degrees of freedom
## Multiple R-squared:  0.7454, Adjusted R-squared:  0.7444
## F-statistic: 752.3 on 17 and 4368 DF,  p-value: < 2.2e-16
```

```r
# function to invert Box-Cox transformation
# reference: https://stat.ethz.ch/pipermail/r-help/2007-June/134480.html
invBoxCox <- function(x, lambda)
            if (lambda == 0) exp(x) else (lambda*x + 1)^(1/lambda)


# try to use the model to predict 3 players's performence in the last season (2020-21 season)
```

```
# because we did not use that season's data for training the model, we will use it to test

# Tests
# 1st player is LeBron James, an older player, very good at every aspect of the game
# 2nd player is Stephen Curry, the player with the most points per game in the 2020-21 season
# 3rd player is Fred VanVleet, a player that showed a lot of improvement in the 2020-21 season
# 4th player is James Wiseman, new player, big man, was the 2nd overall pick in the 2020 draft
# 5th player is Tristan Thompson, a Canadian player, average in scoring points, good rebounder

# attempt to predict LeBron James's Points Score Per Game for the 2020-21 season
player_interested = 'LeBron James'
season_interested = '2020-21'
queriedvalues = filter((filter(bb_raw, season==season_interested)),
                               player_name==player_interested)
pred=predict.lm(finalmodel,
                newdata=queriedvalues,
                level=0.95, interval="prediction")
```

```
## Warning in predict.lm(finalmodel, newdata = queriedvalues, level = 0.95, :
## prediction from a rank-deficient fit may be misleading
```

```
point_est = invBoxCox(pred[1], bestlambda)
low_est= invBoxCox(pred[2], bestlambda)
high_est= invBoxCox(pred[3], bestlambda)

cat('Model Testing: \nPlayer Name:', player_interested,'\nSeason:',season_interested,
    '\nPoints Per Game, 95% prediction interval:(',
    round(low_est, digits = 4), ', ', round(high_est, digits = 4), ') ',
    '\nPoints Per Game, point prediction:', round(point_est, digits = 4),
    '\nPoints Per Game, actual:', round(queriedvalues$pts, digits = 4), '\n')
```

```
## Model Testing:
## Player Name: LeBron James
## Season: 2020-21
## Points Per Game, 95% prediction interval:( 13.06 ,  33.246 )
## Points Per Game, point prediction: 21.7474
## Points Per Game, actual: 25
```

```
# attempt to predict Stephen Curry's Points Score Per Game for the 2020-2021 season
player_interested = 'Stephen Curry'
season_interested = '2020-21'
queriedvalues = filter((filter(bb_raw, season==season_interested)),
                               player_name==player_interested)
pred=predict.lm(finalmodel,
                newdata=queriedvalues,
                level=0.95, interval="prediction")
```

```
## Warning in predict.lm(finalmodel, newdata = queriedvalues, level = 0.95, :
## prediction from a rank-deficient fit may be misleading
```

```
point_est = invBoxCox(pred[1], bestlambda)
low_est= invBoxCox(pred[2], bestlambda)
high_est= invBoxCox(pred[3], bestlambda)

cat('Model Testing: \nPlayer Name:', player_interested,'\nSeason:',season_interested,
    '\nPoints Per Game, 95% prediction interval:(',
    round(low_est, digits = 4), ', ', round(high_est, digits = 4), ') ',
    '\nPoints Per Game, point prediction:', round(point_est, digits = 4),
    '\nPoints Per Game, actual:', round(queriedvalues$pts, digits = 4), '\n')
```

```
## Model Testing:
## Player Name: Stephen Curry
## Season: 2020-21
## Points Per Game, 95% prediction interval:( 11.8191 ,  30.4926 )
## Points Per Game, point prediction: 19.8376
## Points Per Game, actual: 32
```

```
# attempt to predict Fred VanVleet's Points Score Per Game for the 2020-2021 season
player_interested = 'Fred VanVleet'
season_interested = '2020-21'
queriedvalues = filter((filter(bb_raw, season==season_interested)),
                               player_name==player_interested)
pred=predict.lm(finalmodel,
                newdata=queriedvalues,
                level=0.95, interval="prediction")
```

```
## Warning in predict.lm(finalmodel, newdata = queriedvalues, level = 0.95, :
## prediction from a rank-deficient fit may be misleading
```

```
point_est = invBoxCox(pred[1], bestlambda)
low_est= invBoxCox(pred[2], bestlambda)
high_est= invBoxCox(pred[3], bestlambda)

cat('Model Testing: \nPlayer Name:', player_interested,'\nSeason:',season_interested,
    '\nPoints Per Game, 95% prediction interval:(',
    round(low_est, digits = 4), ', ', round(high_est, digits = 4), ') ',
    '\nPoints Per Game, point prediction:', round(point_est, digits = 4),
    '\nPoints Per Game, actual:', round(queriedvalues$pts, digits = 4), '\n')
```

```
## Model Testing:
## Player Name: Fred VanVleet
## Season: 2020-21
## Points Per Game, 95% prediction interval:( 5.8556 ,  19.491 )
## Points Per Game, point prediction: 11.4642
## Points Per Game, actual: 19.6
```

```
# attempt to predict James Wiseman's Points Score Per Game for the 2020-2021 season
player_interested = 'James Wiseman'
season_interested = '2020-21'
queriedvalues = dplyr::filter((dplyr::filter(bb_raw, season==season_interested)),
                               player_name==player_interested)
```

```
pred=predict.lm(finalmodel,
                newdata=queriedvalues,
                level=0.95, interval="prediction")
```

```
## Warning in predict.lm(finalmodel, newdata = queriedvalues, level = 0.95, :
## prediction from a rank-deficient fit may be misleading
```

```
point_est = invBoxCox(pred[1], bestlambda)
low_est= invBoxCox(pred[2], bestlambda)
high_est= invBoxCox(pred[3], bestlambda)

cat('Model Testing: \nPlayer Name:', player_interested,'\nSeason:',season_interested,
    '\nPoints Per Game, 95% prediction interval:(',
    round(low_est, digits = 4), ', ', round(high_est, digits = 4), ') ',
    '\nPoints Per Game, point prediction:', round(point_est, digits = 4),
    '\nPoints Per Game, actual:', round(queriedvalues$pts, digits = 4), '\n')
```

```
## Model Testing:
## Player Name: James Wiseman
## Season: 2020-21
## Points Per Game, 95% prediction interval:( 3.3211 ,  13.6544 )
## Points Per Game, point prediction: 7.4213
## Points Per Game, actual: 11.5
```

```
# attempt to predict Tristan Thompson's Points Score Per Game for the 2020-2021 season
player_interested = 'Tristan Thompson'
season_interested = '2020-21'
queriedvalues = filter((filter(bb_raw, season==season_interested)),
                               player_name==player_interested)
pred=predict.lm(finalmodel,
                newdata=queriedvalues,
                level=0.95, interval="prediction")
```

```
## Warning in predict.lm(finalmodel, newdata = queriedvalues, level = 0.95, :
## prediction from a rank-deficient fit may be misleading
```

```
point_est = invBoxCox(pred[1], bestlambda)
low_est= invBoxCox(pred[2], bestlambda)
high_est= invBoxCox(pred[3], bestlambda)

cat('Model Testing: \nPlayer Name:', player_interested,'\nSeason:',season_interested,
    '\nPoints Per Game, 95% prediction interval:(',
    round(low_est, digits = 4), ', ', round(high_est, digits = 4), ') ',
    '\nPoints Per Game, point prediction:', round(point_est, digits = 4),
    '\nPoints Per Game, actual:', round(queriedvalues$pts, digits = 4), '\n')
```

```
## Model Testing:
## Player Name: Tristan Thompson
## Season: 2020-21
## Points Per Game, 95% prediction interval:( 6.5764 ,  20.6291 )
## Points Per Game, point prediction: 12.4151
## Points Per Game, actual: 7.6
```

Chapter 4: Conclusion

Through the above analysis, the best fit model based on the data set is the model we tested at the very end, reducedbcmodel, where the response variable has had a Box-Cox transformation applied, and the interactions and higher order terms have been added for our independent variables, with all insignificant variables taken out. Plugging in all the coefficients, the model equation is as follows:

$$\hat{pts}^{0.3939394} = -6.8869261 + 0.2807468 * age + 0.0342418 * height + 0.0221654 * gp$$
$$+2.2314618 * reb - 0.2792760 * ast - 0.0575766 * ast^2 + 0.0005540 * netRating$$
$$-0.0014460 * age * height - 0.0056161 * age * ast - 0.0084595 * height * reb$$
$$+0.0060871 * height * ast - 0.0032200 * gp * reb + 0.0037385 * ast * netRating +$$
$$\begin{cases} 0 & if\ dRound = 1 \\ -0.1696891 - 0.0245812 * ast & if\ dRound = 2 \\ -0.0317125 - 0.1742997 * ast & if\ dRound = U \end{cases}$$

Please note that we have the option to simplify this equation further, however, since that will be confusing to read as there will be several lines of equation for each dRound, we will leave the equation as it is for clarity purposes.

Here is a table showing all models we tested in our code and their R^2_{adj}, and RMSE values.

Model Name in Code	R^2_{adj}	RMSE	Note
fullmodel	0.6627	3.349	8 Variables, contains insignificant term
stepw, backmodel, formodel, reducedmodel	0.6626	3.350	7 Variables, reduced from fullmodel, p+1 closest to Cp
interacmodel	0.6954	3.183	full interaciton model based on reducedmodel, contains insignificant term
reducedmodel2	0.6941	3.190	reduced from interacmodel, no good, too much reduction, took out significant term
reducedmodel3	0.6948	3.186	reduced from interacmodel, with insignificant terms taken out
highermodel	0.7084	3.114	based on reducedmodel3, contains higher order terms
highermodel2	0.7083	3.115	based on highermodel, with an insignificant higher order term taken out
reducedmodel4	0.7082	3.116	based on highermodel2, with an insignificant interaction term taken out
highermodel3	0.7082	3.115	based on reducedmodel4, try even higher order for "ast", no good, insignificant
highermodel4	0.7084	3.114	based on reducedmodel4, try even higher order for "reb", no good, insignificant
outlier_removed_model1	0.7078	3.072	based on reducedmodel4, with influential points $h_i > \frac{3p}{n}$ removed, used (some influential points taken out proved to be bad data)
outlier_removed_model2	0.7030	3.063	based on reducedmodel4, with influential points $h_i > \frac{2p}{n}$ removed, not used (the additional points taken out were good data)
reducedmodel5	0.7077	3.073	based on outlier_removed_model1, with insignificant terms taken out
reducedmodel6	0.7075	3.074	based on reducedmodel5, with more insignificant terms taken out
reducedmodel7	0.7074	3.075	based on reducedmodel6, with more significant terms taken out

Model Name in Code	R^2_{adj}	RMSE	Note
bcmodel	0.7443	0.773	based on reducedmodel7, with box-cox transformation done using $\lambda = 0.3939394$
reducedbcmodel	0.7444	0.772	based on bcmodel, with insignificant interaction term taken out, this is best fit model

Here is a summary of why the model we chose is the best fit model for our data:

- Highest R^2_{adj} and lowest RMSE that had no insignificant terms
- Variables chosen gives p+1 closest to Cp, which indicates low bias

Independent Variable Interpretation or Significance

As for the interpretation of the effects of our independent variables present in the best fit model, we will not do individual interpretations as response variable has gone through a Box-Cox transformation and interpreting it post transformation is beyond the scope of our current knowledge, so we will list below the independent variables significant to our response variable.

Independent variables significant to our response variable are:

- age - the age of the player at the start of the season in question, in years
- height - the height of the player at the start of the season in question, in centimeters
- d_round - the round that the player was drafted in the NBA draft
- gp - the number of games the player has played in for the season in question
- reb - the number of rebounds per game for the player for the season in question
- ast - the number of assists per game for the player for the season in question
- net_rating - the team's point differential per 100 possessions while the player is on the court for the season in question

As an added clarification, independent variables that was insignificant to our response variable and was therefor not included in our best fit model is:

- weight - the weight of the player at the start of the season in question, in kilograms

R^2_{adj}, and RMSE interpretation for the best fit model

From the best fit MLR model selected, here is an interpretation of: R^2_{adj}, and RMSE: • $R^2_{adj} = 0.7444$, hence 74.44% of the variation of the points per game in a season for an NBA player who has played 10 or more games in the season is explained by the model. • An RMSE = 0.772 means that the standard deviation of the unexplained variance by the Box-Cox transformed model is 0.772

General discussion of the results

Now, to discuss the results (expected vs. actual), and answers to our research questions:

1. We expected individual attributes such as age, height, and weight may influence a players points per game stat. Result:

- age and height both proved to be significant,
- However it seems weight was not a significant variable.

2. We expected that box score statistics such as number of ast, reb, net_rating, gp may influence a player's points per game stat. Result:

- ast, reb, net_rating, gp all proved to be significant.

3. We also expected that the round the player was drafted in the NBA draft may influence a players points per game stat. Result:

- draft round proved to be significant.

Overall the results are in line with what we expected from the introduction section.

Possible improvements/changes

As for possible improvements/changes to the best model selected, here are some improvements we may be able to do:

1. Try to test and see if there are significant logarithmic terms,
2. Apart from Box-Cox transformation, perhaps we could try other transformations to have the model meet Normality and Equal Variance assumptions.
3. The biggest improvement that we could do is to more closely examine our data next time and catch any mistakes before building our model. For this project, during the step where we examine for outliers and influential points, we caught some erroneous data in our "draft round"(d_round) variable. We saw that in the data used for modeling, there were 5 possible values for this qualitative variable, which were "0", "1", "2", "4", and "Undrafted". Through further research, we realized it's not possible for the the variable to carry the "0" and "4" Value as the NBA draft only had 2 rounds. the "0" value was there because the dataset used that to represent "Undrafted" players on several occasions, and the "4" value was there for unknown reasons, possibly a typo. We were able to take those data out for the next modeling steps after that, but should we have caught these errors in the data before we started modeling, we could have started with clean data which is definitely preferred.

Overall, this project was a great learning experience for both us, as we learned from our mistakes, and at the same time, gained a lot of valuable experience for any future sports analytics modeling we do in the future, and we look forward to learning more modeling techniques and statistical methods in the future.

References

NBA Rulebook (NBA, 2021). Retrieved December 6, 2021, from https://official.nba.com/rulebook/

NBA Players Biometric, biographic and basic box score features from 1996 to 2021 season. (Kaggle, 2021). Retrieved December 6, 2021, from https://www.kaggle.com/justinas/nba-players-data