Math 487/Stat 442 Final Project

Ashley Akamine

2023-12-13

```
library('wehoop')
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.2.3
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
box_scores <- wehoop::load_wbb_player_box()
```

Research Question

How do different player statistics (such as minutes played, field goal percentage, and rebounds) influence a player's scoring efficiency in basketball games?

1. About the data set

a. My research question of interest is "How do different player statistics (such as minutes played, field goal percentage, and rebounds) influence a player's scoring efficiency (points per shot attempt) in basketball games?" I think it is worthwhile to study because I have played basketball for most of my life and how efficiently a player scores is a key to winning games. I also wanted to take this final project as an opportunity to explore sports analysis and use Women's Basketball data as I don't see a lot of sports analysis high lightening women's sports.

b. I saw a Tiktok created by Maddy wnba (@wnbadata), that used the 'wehoop' R package to model the frequency of NCAA D1 Women's basketball jersey numbers containing digits 6-9, as the NCAA is now allowing players to have jerseys 0-99. After narrowing down the kind of research question and demographic I wanted to explore, I came across the NCAA D1 Women's Basketball Play by Play data set within this R package.

c. This data set is part of the wehoop package in R. The box_scores data set from the wehoop package includes detailed player-level statistics from women's college basketball games. These statistics often encompass various performance metrics such as points scored, assists, rebounds, minutes played, field goal percentages, free throw percentages, three-point percentages, turnovers, and more.

d. https://cran.r-project.org/web/packages/wehoop/wehoop.pdf (https://cran.r-project.org/web/packages/wehoop/wehoop.pdf) on page 62.

e. Some ethical implications about using this data to model a player's scoring efficiency is the interpretation of the model's results may portray some players in a particular way depending on how "good" or "bad" their scoring efficiency is. However it shouldn't be used to make definitive judgement about a player's overall value and capabilities. Additionally, such a model's results might inadvertently lead to labeling or categorizing players in ways that could impact their career opportunities or public perception. It's also crucial to remember that box score data lacks contextual elements like team dynamics, game situations, and player roles, which can significantly influence performance metrics

2. Candidate Models

a. Log Scoring Efficiency Main Effects Model By Free Throw Percentage (ftm_main_mod):
$$\text{Log Scoring Efficiency} = \beta_0 + \beta_1 \times \text{free\_throw\_percentage} + \epsilon$$

Log Scoring Efficiency Main Effects Model By Athlete Position (condensed_pos_mod):
$$\text{log\_scoring\_efficiency} = \beta_0 + \beta_1 \times \text{Ipg} + \beta_2 \times \text{Ig} + \beta_3 \times \text{If} + \beta_4 \times \text{Ic} + \epsilon$$

Log Scoring Efficiency Interaction Model Athlete Position, Free Throw Percentage, and their interaction (fs_int_mod):
$$\text{log\_scoring\_efficiency} = \beta_0 + \beta_1 \times \text{Ipg} + \beta_2 \times \text{Ig} + \beta_3 \times \text{If} + \beta_4 \times \text{free\_throw\_percentage} + \beta_5 \times \text{Ipg} \times \text{free\_throw\_percentage} + \beta_6 \times$$

Log Scoring Efficiency Interaction Effects Model By IatHome, Free Throw Percentage, and their interaction (mh_int_mod):
$$\text{log\_scoring\_efficiency} = \beta_0 + \beta_1 \times \text{IatHome} + \beta_2 \times \text{free\_throw\_percentage} + \beta_3 \times \text{IatHome} \times \text{free\_throw\_percentagefree\_throw\_perc}$$

Log Scoring Efficiency Interaction Effects Model By Minutes and Athlete Position (mtp_main_mod):
$$\text{log\_scoring\_efficiency} = \beta_0 + \beta_1 \times \text{Minutes} + \beta_3 \times \text{free\_throw\_percentage} \times \text{Minutes} + \beta_4 \times \text{Ipg} + \beta_5 \times \text{Ig} + \beta_6 \times \text{If} + \epsilon$$

b. Log Scoring Efficiency: Dependent variable, quantitative. It's the natural logarithm of scoring efficiency, which is also know as the points per shot attempt.

ftm_main_mod

free_throws_percentage (FT): Independent variable, quantitative. Represents the percentage of free throws made by a player.

condensed_pos_mod

athlete_position_name: Independent variable, categorical. Represents the position of the athlete (Center, Forward, Guard, Point Guard).

fs_int_mod

athlete_position_name: Independent variable, categorical. Represents the position of the athlete (Center, Power Forward, Small Forward, Forward, Gurad, Shooting Guard, Point Guard). I am using Center as the reference group.

free_throw_percentage: Independent variable, quantitative. Represents the percentage of free throws made by a player.

mh_int_mod

IatHome: Independent variable, binary. Indicates whether the game is at home (1) or away (0).

free_throw_percentage: Independent variable, quantitative. Represents the percentage of free throws made by a player.

IatHome:free_throw_percentage: Interaction term, quantitative. Represents the interaction between the game being at home and the percentage of free throws made.

mtp_main_mod

Minutes: Independent variable, quantitative.

athlete_position_name: Independent variable, categorical. Represents the position of the athlete (Center, Power Forward, Small Forward, Forward, Gurad, Shooting Guard, Point Guard). I am using Center as the reference group.

c. tm_main_mod (Main Effects Model with Free Throw Percentage)

Reasoning: This model assesses the impact of free throw percentage. Free throws are often regarded as high-probability scoring opportunities in basketball. Players take these shots without any defensive pressure, which typically leads to a higher success rate compared to field goals. Additionally, for every free throw attempted/made, it is not counted toward field goals attempted. So it will directly raise a player's shooting efficiency, if they take/make a lot of free throws.

condensed_pos_mod (Main Effects Model with Athlete Position)

Reasoning: This model examines how scoring efficiency differs by player position. Different positions in basketball often have distinct roles and scoring opportunities, so it's valuable to understand how position affects scoring efficiency. Including free throws made as a covariate ensures that the analysis accounts for one aspect of scoring that might be consistent across positions. I condensed the "Shooting Guard" position into "Guard" and "Small Forward" & "Power Forward" into "Forward." Most coaches/players tend to omit these specializations and it makes interpretation easier.

fs_int_mod (Interaction Model with Athlete Position, Free Throw Percentage, and their Interaction)

Reasoning: This model looks at the combined effect of athlete position, free throws percentage, and their interaction. Including the interaction term between player position and free throws made allows the model to explore whether the impact of free throw percentage on scoring efficiency varies by player position.

mh_int_mod (Interaction Effects Model by IatHome, Free Throw Percentage, and their Interaction)

Reasoning: The purpose here is to explore how scoring efficiency is influenced by the game location (home vs. away) and free throw percentage, along with their interaction. The interaction term can reveal if the advantage of playing at home interacts with a player's ability to make free throws, which might indicate differing pressures or comfort levels in different venues.

mtp_main_mod (Interaction Effects Model by Minutes and Athlete Position)

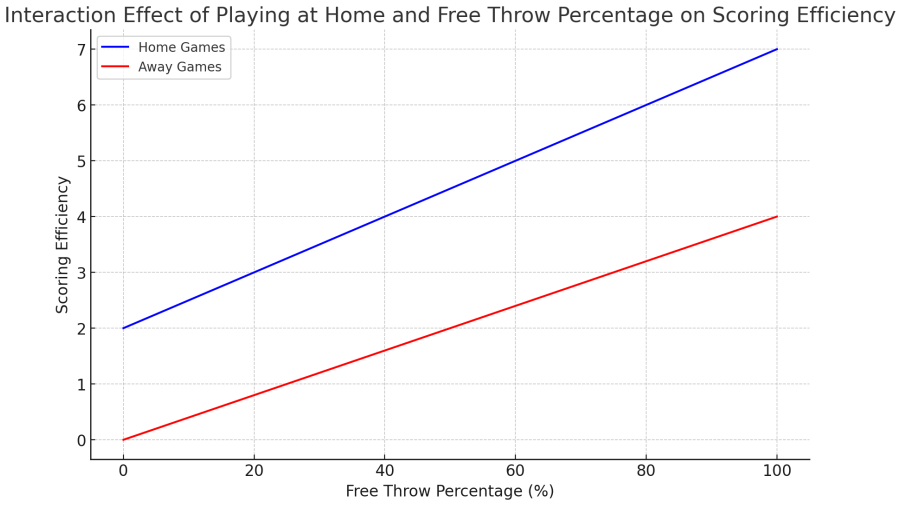
Reasoning: This model investigates how scoring efficiency is affected by both minutes played, and athlete position. Minutes played can indicate the athlete's endurance and coach's trust.

d. mh_int_mod (Interaction Effects Model by IatHome, Free Throw Percentage, and their Interaction)

The coefficient for IatHome is what the model predicts would be the difference in log scoring efficiency for a player when a game is played at home and a game played away.

The coefficient for Free Throw Percentage is what the model predicts would be the change log scoring efficiency for each unit increase in free throw percentage, holding game location constant.

The coefficient for the interaction term is what the model predicts the effect of "Free Throws Percentage" is, specifically for home games. This coefficient tells us how much more (or less) influential free throw percentage is on the log scoring efficiency when the game is at home, compared to when it's away.



Final Plot

Both lines trend upwards, showing that an increase in free throw percentage is associated with increased scoring efficiency. However, the slope of the blue line (home games) is steeper, indicating a more pronounced effect of free throw percentage on scoring efficiency in home games. This steeper slope for home games aligns with the positive interaction coefficient, suggesting that players may perform better in free throws, contributing more to their overall scoring efficiency, when they are in a familiar home environment.

3. Data Prep & Cleaning

```
library(dplyr)
box_scores <- wehoop::load_wbb_player_box()
```

```
# Trimming white space to filter inputs in these fields
box_scores$starter <- trimws(box_scores$starter)
box_scores$home_away <- trimws(box_scores$home_away)

# format indicator variables
box_scores$IatHome <- ifelse(box_scores$home_away == "home", 1 ,0 )
box_scores$Istarter <- ifelse(box_scores$starter == "TRUE", 1 ,0 )

# calculate scoring_efficiency & log_scoring_efficiency
box_scores$scoring_efficiency <- ifelse(box_scores$field_goals_attempted > 0, box_scores$points / box_scores$field_goals_att
empted, NA)
box_scores$log_scoring_efficiency <- log(box_scores$scoring_efficiency)

# calculate point percentage
box_scores$three_point_percentage <- with(box_scores, three_point_field_goals_made / three_point_field_goals_attempted)
box_scores$log_three_point_percentage <- log(box_scores$three_point_percentage + 0.0001)

box_scores$free_throw_percentage <- with(box_scores, free_throws_made / free_throws_attempted)
box_scores$log_free_throw_percentage <- log(box_scores$free_throw_percentage + 0.0001)


# removing rows with that had unspecified position names
box_scores <- box_scores[!box_scores$athlete_position_abbreviation == "NA", ]
box_scores <- box_scores[!box_scores$athlete_position_name == "Athlete", ]

# removing rows with players who didn't shoot
box_scores <- box_scores[!is.na(box_scores$scoring_efficiency), ]
box_scores <- box_scores[!is.infinite(box_scores$log_scoring_efficiency), ]


# Recode the athlete_position_name into broader categories
box_scores$condensed_position <- with(box_scores, ifelse(athlete_position_name %in% c("Shooting Guard", "Guard"), "Guard",
                                  ifelse(athlete_position_name %in% c("Power Forward", "Forward", "Small Forward"), "Forwa
rd",
                                  ifelse(athlete_position_name == "Point Guard", "Point Guard", "Center"))))
# Relevel so 'Center' is the reference group
box_scores$condensed_position <- factor(box_scores$condensed_position)
box_scores$condensed_position <- relevel(box_scores$condensed_position, ref = "Center")
```

```
# Main Effects Model with Free Throw Percentage
ftm_main_mod <- lm(log_scoring_efficiency ~ free_throw_percentage, data = box_scores)
```

```
# Main Effects Model with Athlete Position
condensed_pos_mod <- lm(log_scoring_efficiency ~ condensed_position, data = box_scores)
```

```
# Interaction Model with Athlete Position, Free Throw Percentage, and their Interaction
fs_int_mod <- lm(log_scoring_efficiency ~ free_throw_percentage * condensed_position, data = box_scores)
```

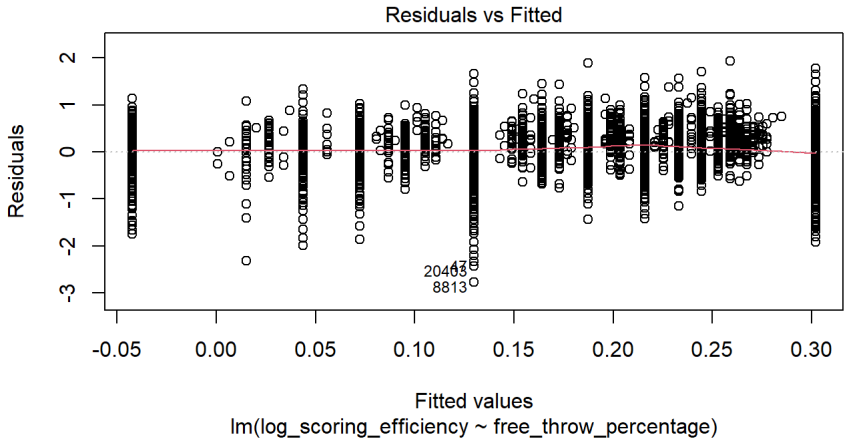
```
# Interaction Effects Model by IatHome, Free Throw Percentage, and their Interaction
mh_int_mod <- lm(log_scoring_efficiency ~ free_throw_percentage * IatHome, data = box_scores)
```

```
# Main Effects Model by Minutes, Free Throw Percentage, and their Interaction
mtp_main_mod <- lm(log_scoring_efficiency ~ minutes + condensed_position, data = box_scores)
```

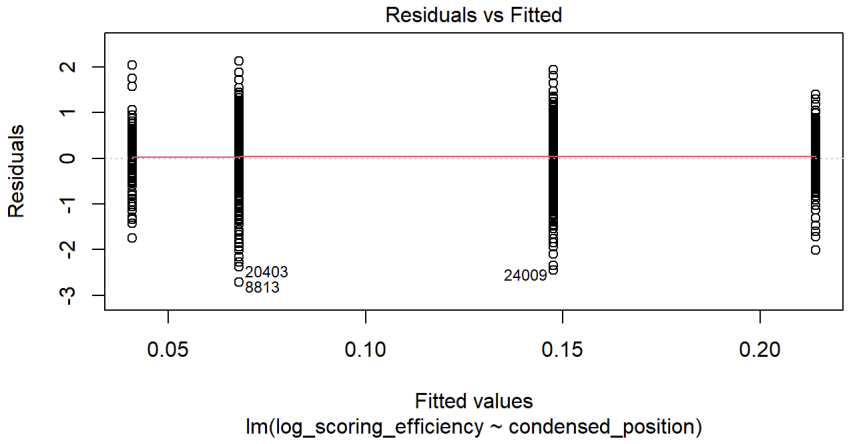
```
pos_ftp_main_mod <-  lm(log_scoring_efficiency ~ condensed_position + free_throw_percentage, data = box_scores)
```

4. Evaluating Model Assumptions

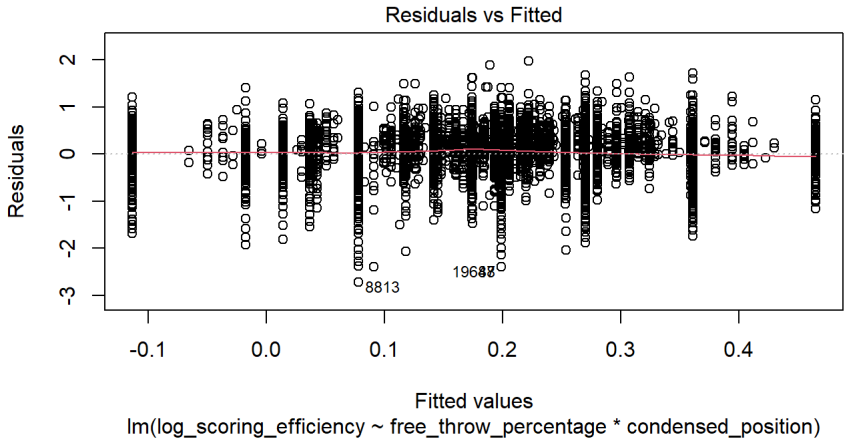
```
plot(ftm_main_mod, 1)  # 5
```



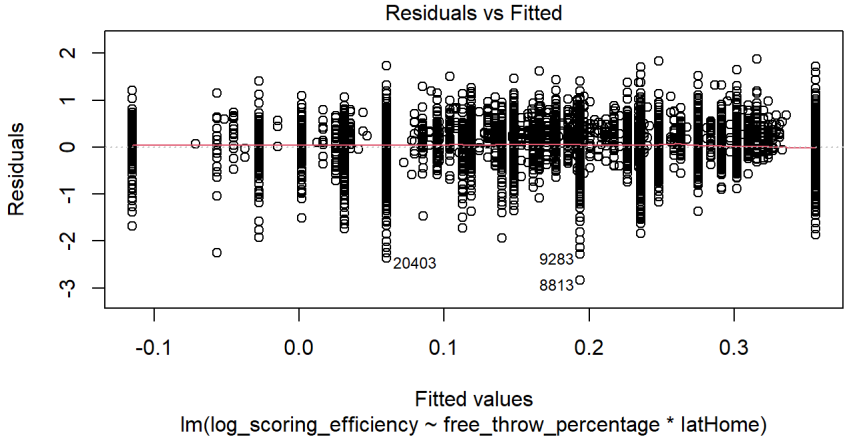

```
plot(condensed_pos_mod, 1) # 1
```



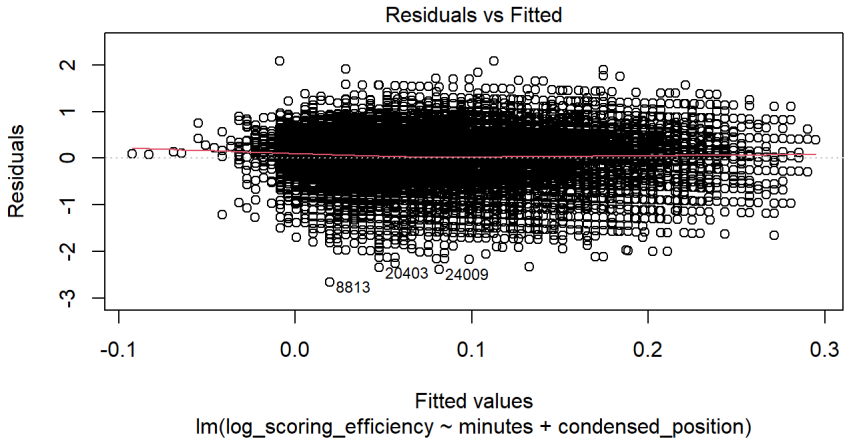

```
plot(fs_int_mod, 1)   # 3
```



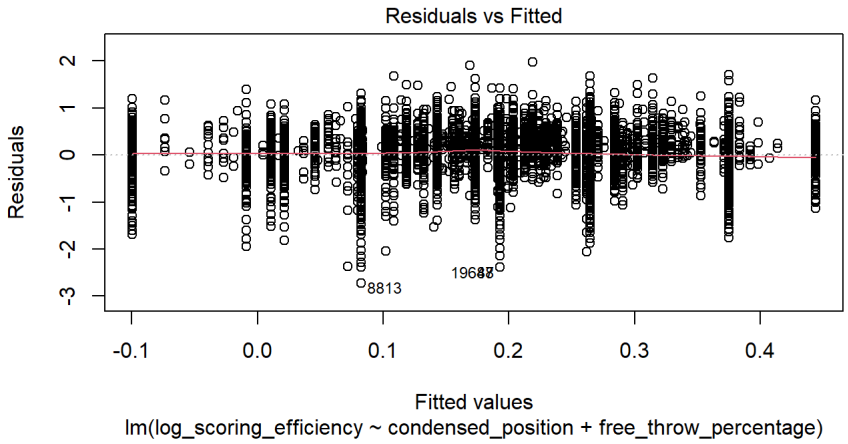

```
plot(mh_int_mod, 1)   # 2
```



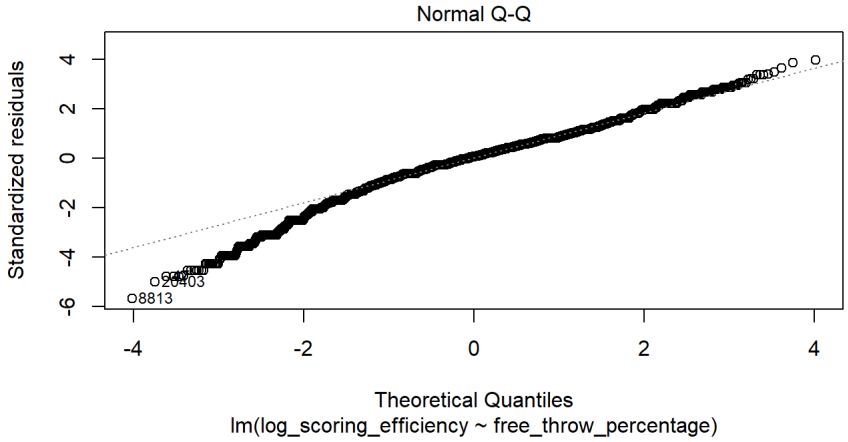

```
plot(mtp_main_mod, 1)  # 5
```



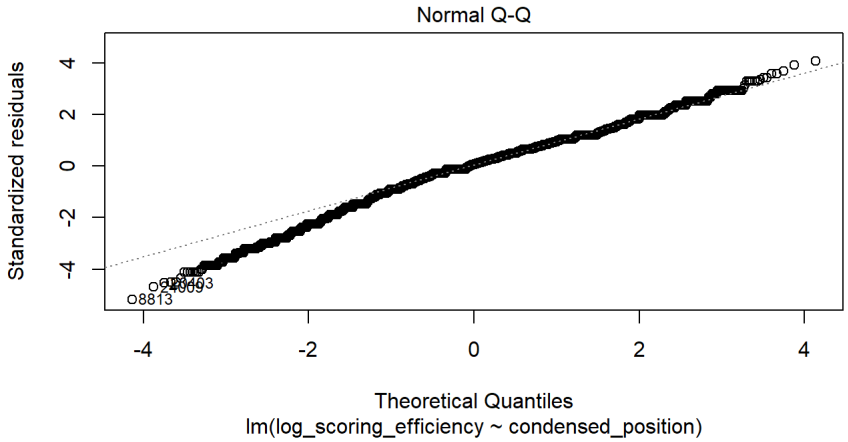

```
plot(pos_ftp_main_mod, 1) # 2
```



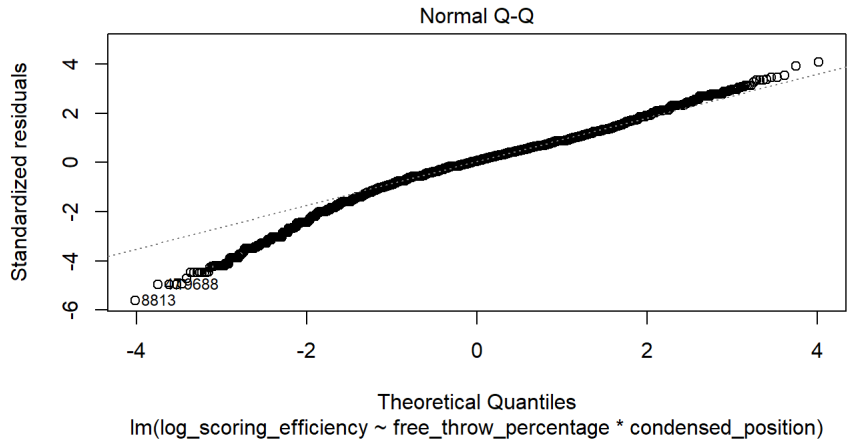

```
plot(ftm_main_mod, 2)   # 3
```



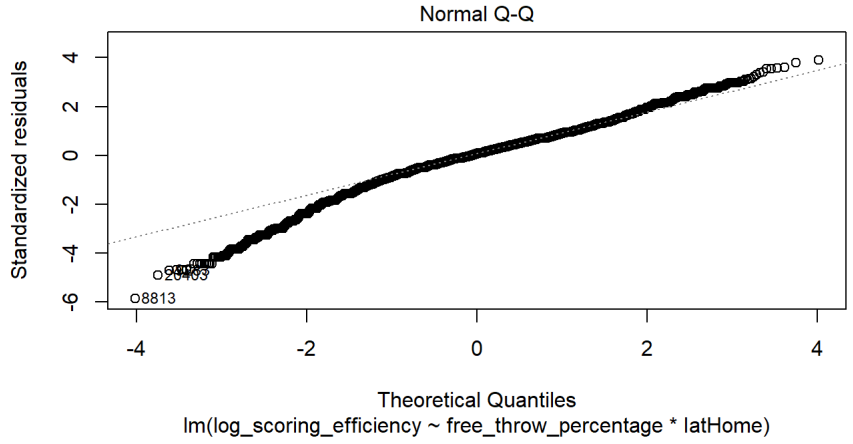

```
plot(condensed_pos_mod, 2)  # 1
```



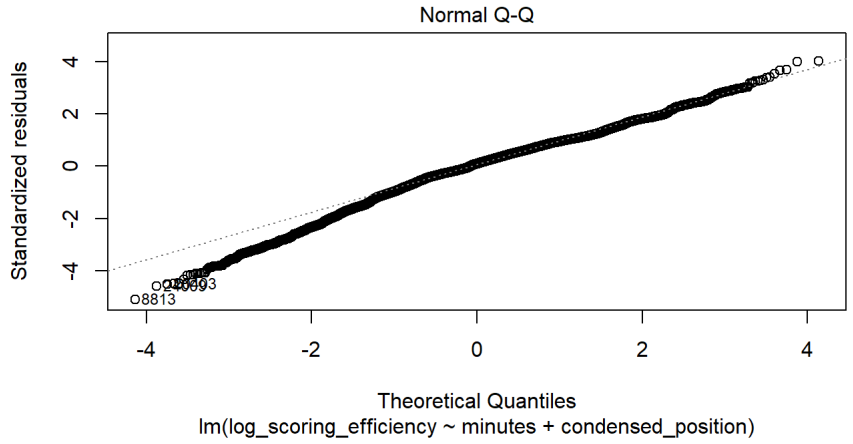

```
plot(fs_int_mod, 2)    # 3
```



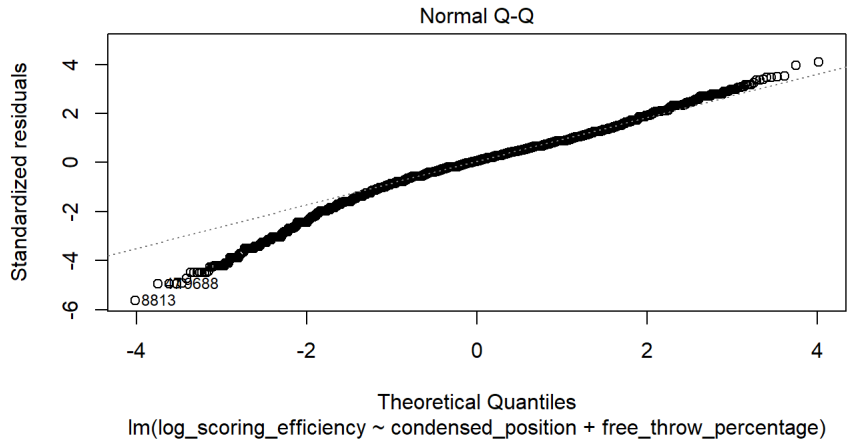

```
plot(mh_int_mod, 2)    # 5
```



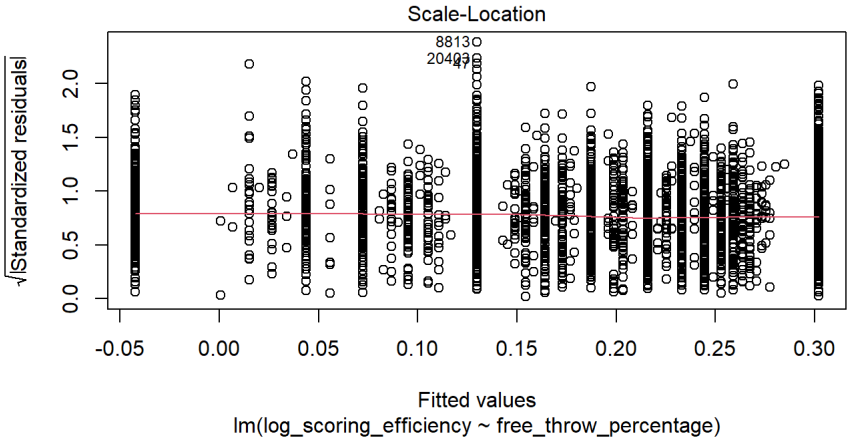

```
plot(mtp_main_mod, 2)   # 2
```



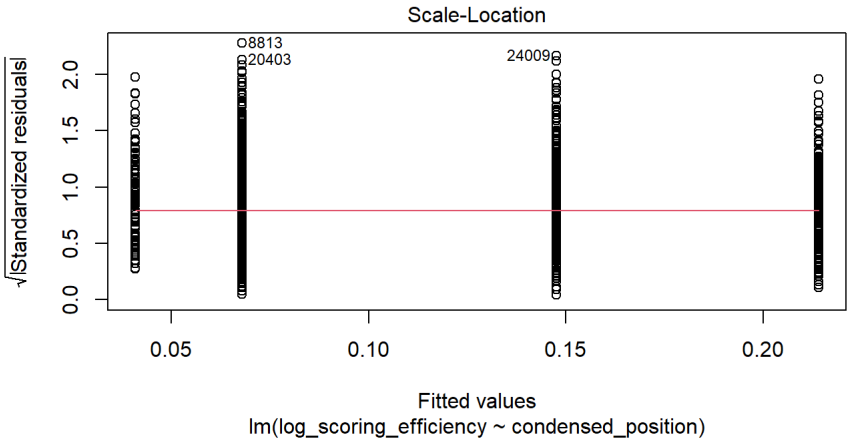

```
plot(pos_ftp_main_mod, 2) # 2
```



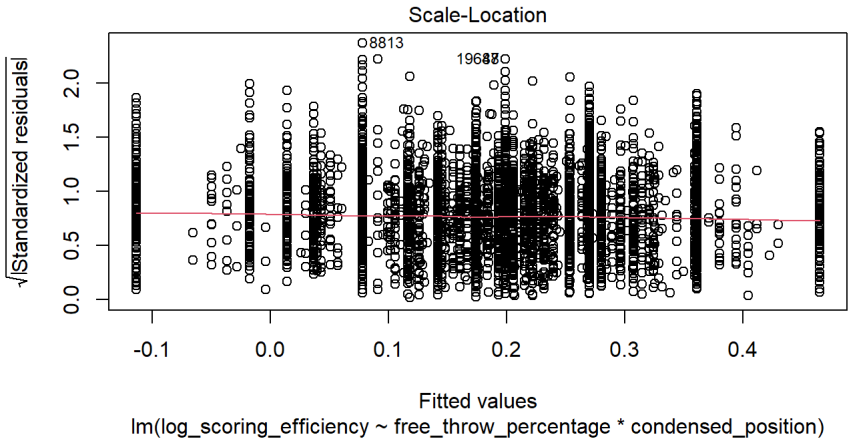

```
plot(ftm_main_mod, 3)  # 5
```



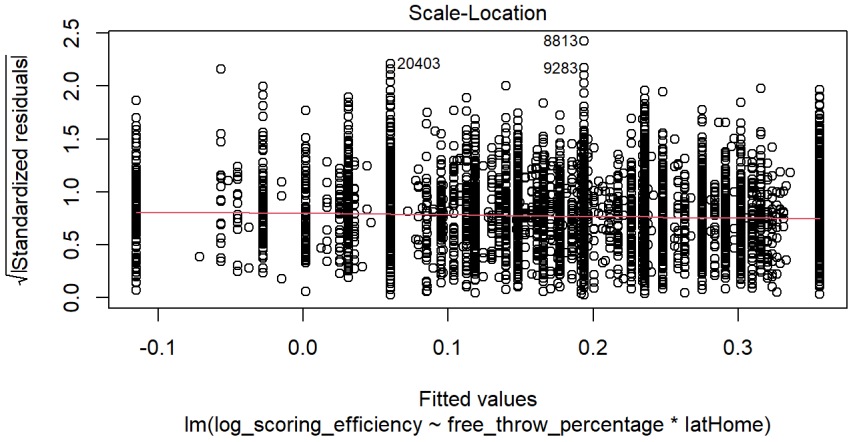

```
plot(condensed_pos_mod, 3) # 1
```



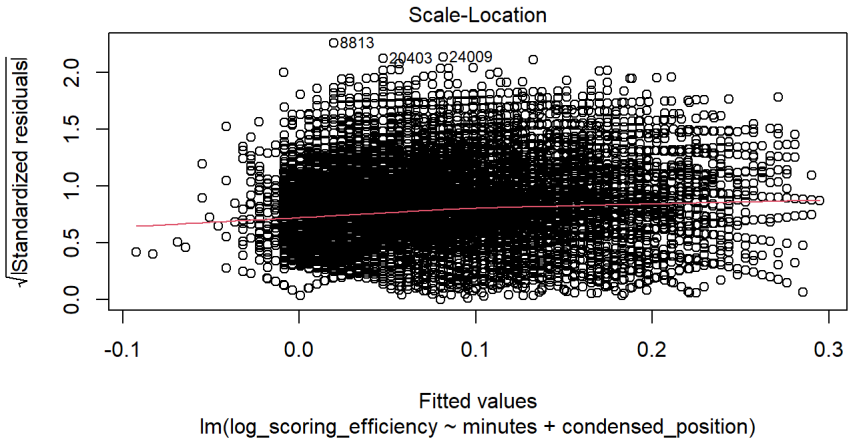

```
plot(fs_int_mod, 3)   # 2
```



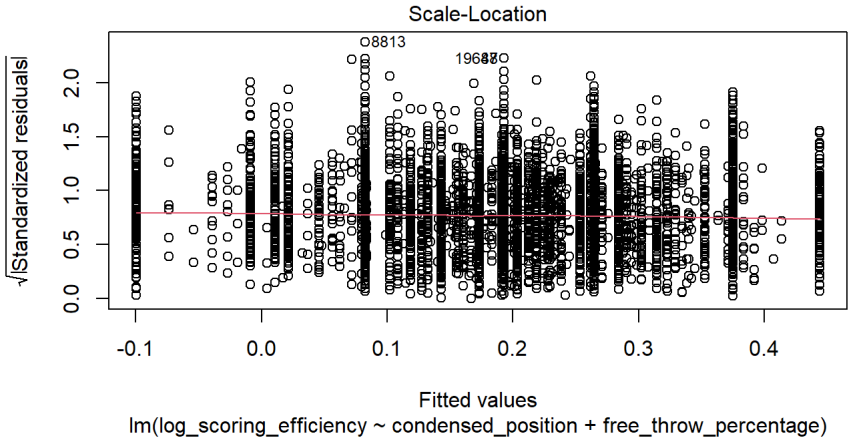

```
plot(mh_int_mod, 3)   # 3
```




```
plot(mtp_main_mod, 3)  # 5
```




```
plot(pos_ftp_main_mod, 3) # 2
```



a. Model Rankings:
   1. condensed_pos_mod
   2. pos_ftp_main_mod
   3. fs_int_mod
   4. mh_int_mod
   5. ftm_main_mod
   6. mtp_main_mod

b. I used a log transformation on the response variable, scoring efficiency to address some skew. I initially started working with “Free Throws Made”, “Field Goals Made” and other just plain shot data. I tried using Free Throws Made as a covariate, it was a good start but I found using Free Throw Percentage was even better for meeting the linearity assumptions. I also condensed the initial positions to just “Point Guard, Guard, Forward, Center.” From personal experience, I know many teams and coaches tend to omit the specializations within the Guard and Forward classes.

5. Model Fit

a. I will evaluate the model fit by considering multiple different metrics such as the R Squared/Adjusted R Squared Value, the Residual Standard Error, and F-Statistic.

b. I will test each model according to these measure:

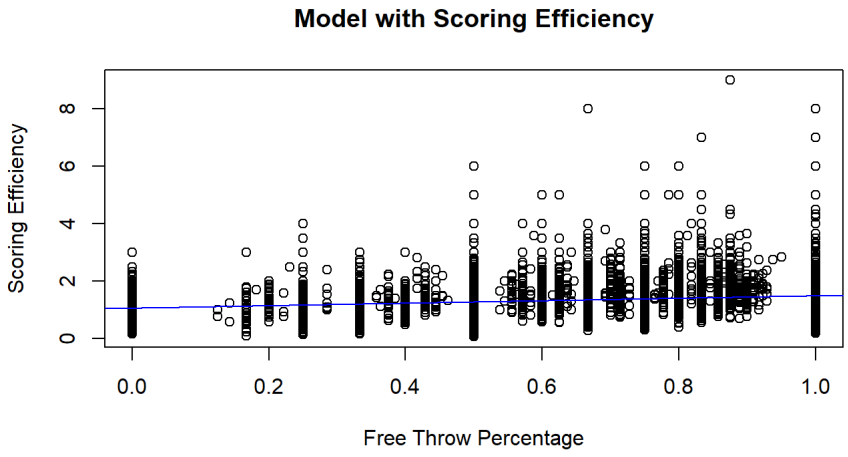
1. R-squared / Adjusted R-squared:
   - Advantage: It’s a measure that indicates the portion of variance in the dependent variable that’s explained by the independent variable. The Adjusted R-Square also takes into the number of predictors, penalizing more complexity.
   - Disadvantage: High R-squared values can be misleading if the independent variables don’t actually cause the changes in the dependent variable (Correlation does not equal causation)
2. Residual Standard Error (RSE):
   - Advantage: This measure tells you the average difference that the observed values fall from the regression line. Basically this allows us to see how much error the model is making in its predictions.
   - Disadvantage: It does not account for the complexity of the model. For example, a complex model with many predictors may have a lower RSE because it’s over fitting the data. (Again, correlation does not equal causation)
3. F-Statistic
   - Advantage: Useful for models where you have nested versions, as it can show whether the complexity added by including more variables significantly improves the model’s fit.
   - Disadvantage: It’s not as informative when comparing models that are not nested or when models have a single predictor, and it doesn’t address the potential violation of underlying model assumptions.

c. If my best fit results is not the model with the best assumption conditions, I would first need to re-evaluate my best assumptions to ensure I didn’t make a mistake processing the data or misinterpreting results. I would then consider additional model transformations. I would lastly consider how I could balance the importance of fit and assumptions. I would likely want to choose my “best” model as the model that is good at both at these. I would not choose the model that had the worst assumption conditions yet the best fit model.

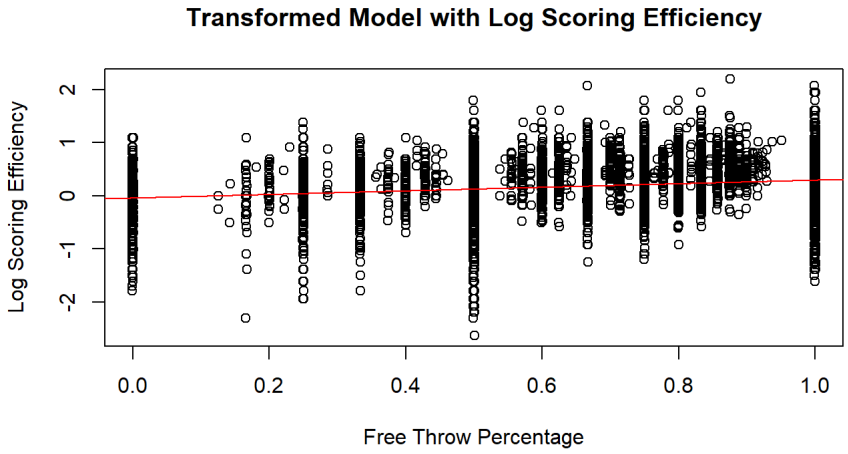
6. Fit the Models

Log Scoring Efficiency Main Effects Model By Free Throw Percentage (ftm_main_mod): $$\widehat{\text{Log Scoring Efficiency}} = -0.04585 + 0.34816 \times \text{free\_throw\_percentage}$$

```
# untransformed plot
nt_ftm_main_mod <- lm(scoring_efficiency ~ free_throw_percentage, data = box_scores)
plot(y = box_scores$scoring_efficiency, x = box_scores$free_throw_percentage, main = "Model with Scoring Efficiency", xlab =
"Free Throw Percentage", ylab = "Scoring Efficiency")
abline(nt_ftm_main_mod, col = "blue")
```




```
# transformed plot
plot(y = box_scores$log_scoring_efficiency, x=jitter(box_scores$free_throw_percentage), main = "Transformed Model with Log S
coring Efficiency", xlab = "Free Throw Percentage", ylab = "Log Scoring Efficiency")
abline(ftm_main_mod, col = "red")
```




```
# summary(ftm_main_mod)
```

Relevant Features & Issues

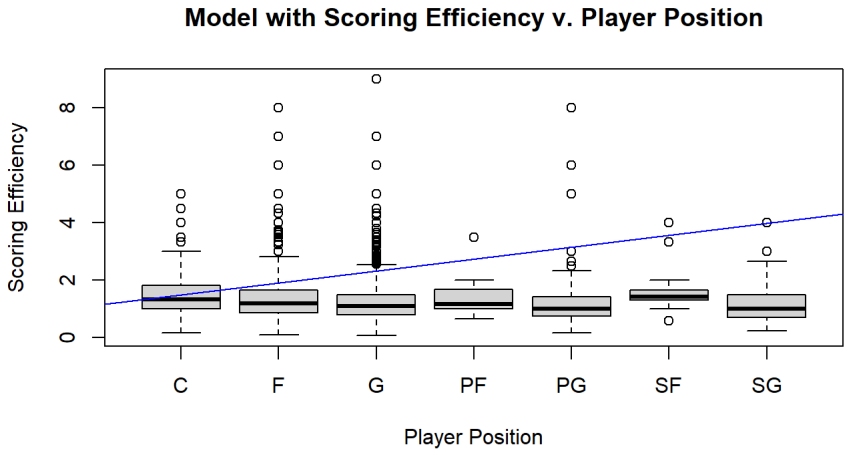
The relationship between free_throw_percentage and log_scoring_efficiency appears linear. It feels balanced as I can visually see that as free throw percentage increases/decreases, the log scoring efficiency also tends to increase/decrease.

I also notice that it seems like for players who shoot above 50% from the line seems to have a Log scoring efficiency above 0. It looks like for players that shoot at 50% seems to have the most variation in terms of log scoring efficiency. An issue I had to deal with when working in this model was that many of the outliers in this log scoring efficiency model where situations where players shot & made many free throws, yet did not take many field goals. Free throws attempted are not counted as field goals attempted and therefore would dramatically increase a players scoring efficiency because it is a measure of points per field goal attempts. This was an interesting issue as I did not want to omit free throws because this shot is vital to the game of basketball and limiting it's influence felt wrong. I dealt with this by turning to free throw percentage and it seemed to control for the outliers better.

Log Scoring Efficiency Main Effects Model By Athlete Position (condensed_pos_mod):

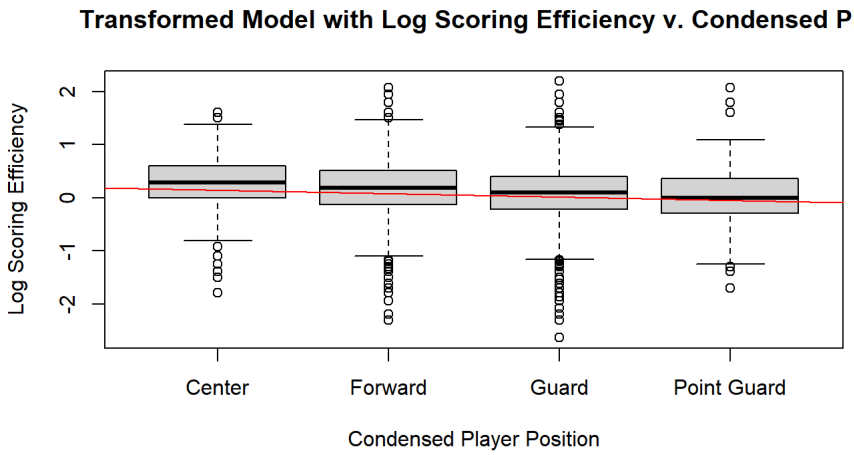
$$\widehat{\text{log\_scoring\_efficiency}} = 0.21382 + -0.16712 \times \text{Ipg} + -0.14450 \times \text{Ig} + -0.06686 \times \text{If} + \beta_4 \times \text{Ic}$$

```
# untransformed plot
box_scores$athlete_position_abbreviation <- factor(box_scores$athlete_position_abbreviation)
nt_pos_mod <- lm(scoring_efficiency ~ athlete_position_abbreviation, data = box_scores)
plot(y=box_scores$scoring_efficiency, x=box_scores$athlete_position_abbreviation, main = "Model with Scoring Efficiency v. P
layer Position", xlab = "Player Position", ylab = "Scoring Efficiency")
abline(nt_ftm_main_mod, col = "blue")
```




```
# transformed plot
plot(y=box_scores$log_scoring_efficiency, x=box_scores$condensed_position, main = "Transformed Model with Log Scoring Effici
ency v. Condensed PP", xlab = "Condensed Player Position", ylab = "Log Scoring Efficiency")
abline(condensed_pos_mod, col = "red")
```

```
## Warning in abline(condensed_pos_mod, col = "red"): only using the first two of 4
## regression coefficients
```




```
# summary(condensed_pos_mod)
```

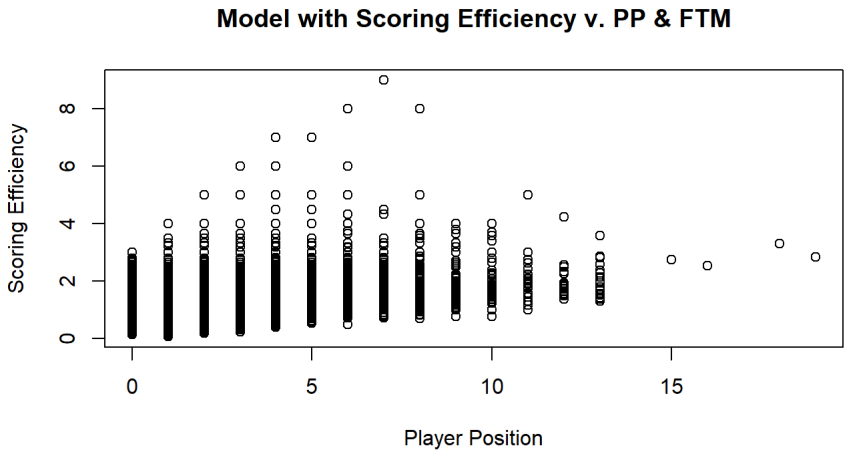
Relevant Features & Issues

There are significant differences in the median log_scoring_efficiency between positions suggest that position affects scoring efficiency. The spread (variability) within each position gives insight into how consistent the positions are regarding scoring efficiency. I would expect the Guard position to have the most variability since the Guard and Forward position has a much wider range of playing styles that may specialize in different areas of basketball. An issue I ran into with my initial model was it was difficult navigating the specialization of these positions. As a collegiate basketball player, I do not often see coaches categorizing their players into these specializations like Shooting Guard, Small Forward, and Power Forward. So using these distinguished positions made me feel skeptical in who actually made up these categories. Additionally, in some model exploration, I found positions like Forward, Small Forward, and Power Forward and then Guard and Shooting Guard were not statistically discernible from each other, which furthered my drive to condense these groups. I dealt with issue by condensing the Guard and Forward positions because it was more traditional and the interpretation of the model's coefficients became simpler.

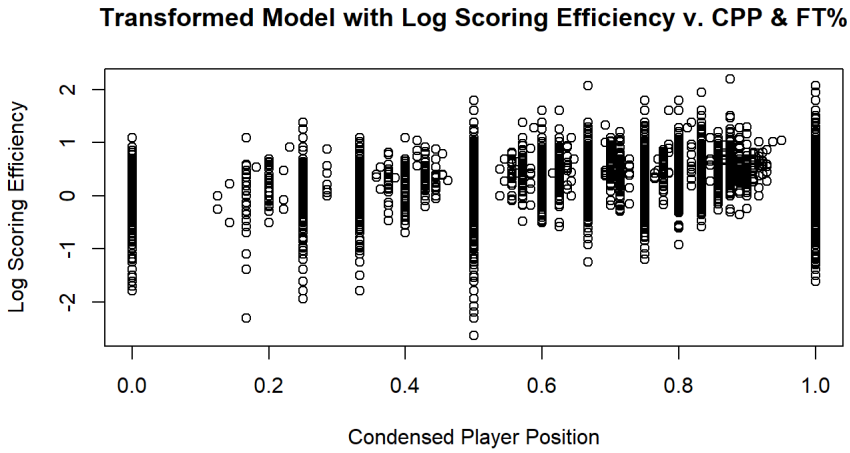
Log Scoring Efficiency Interaction Model Athlete Position, Free Throw Percentage, and their interaction (fs_int_mod):

$$\widehat{\text{log\_scoring\_efficiency}} = 0.05082 + 0.07075 \times \text{Ipg} + -0.16820 \times \text{Ig} + -0.01802 \times \text{If} + 0.05082 \times \text{free\_throw\_percentage} + -0.32074 \times \text{Ipg} \times$$

```
# untransformed plot
nt_fs_int_mod <- lm(scoring_efficiency ~ athlete_position_name * free_throws_made, data = box_scores)
plot(y=box_scores$scoring_efficiency, x=box_scores$free_throws_made, main = "Model with Scoring Efficiency v. PP & FTM", xla
b = "Player Position", ylab = "Scoring Efficiency")
```




```
# transformed plot
plot(y=box_scores$log_scoring_efficiency, x=box_scores$free_throw_percentage, main = "Transformed Model with Log Scoring Eff
iciency v. CPP & FT%", xlab = "Condensed Player Position", ylab = "Log Scoring Efficiency")
```



Relevant Features & Issues

The scatter plot of log scoring efficiency against free throw percentage (FT%) and condensed player position (CPP) indicates a positive relationship between FT% and scoring efficiency, but with considerable variability. The lack of clear patterns among player positions suggests that player position may not significantly differentiate scoring efficiency when considering FT%. The model's complexity and the non-significant interaction terms, except for a marginal effect in Point Guards, imply that the combined effect of position and FT% on scoring efficiency is not pronounced. Given these findings, the model could potentially benefit from simplification by removing non-significant terms to enhance interpretability and focus on the most influential factors. Given the complexity and the non-significance of many terms, I omitted adding the fit lines and this analysis leads me to consider a simpler model. Below I create a model featuring the same terms, without the interaction and it performs better.

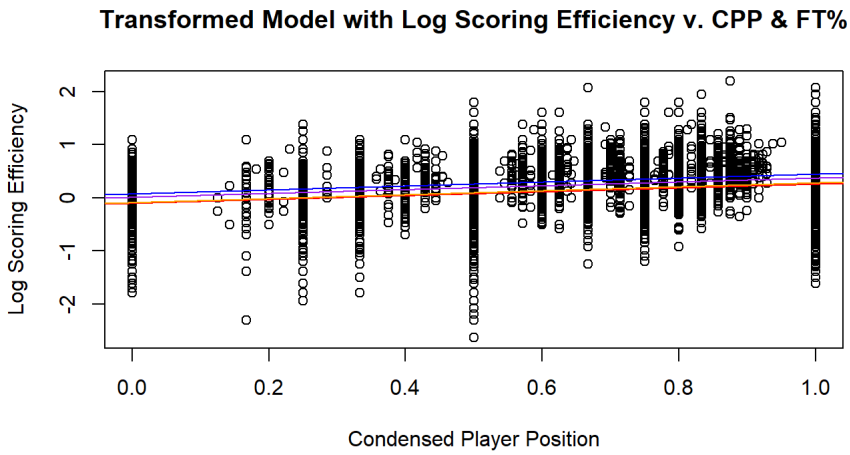
Log Scoring Efficiency Main Effects Model Athlete Position and Free Throw Percentage (pos_ftp_main_mod):

$$\widehat{\text{log\_scoring\_efficiency}} = 0.08067 + -0.16445 \times \text{Ipg} + -0.18323 \times \text{Ig} + -0.07397 \times \text{If} + 0.36756 \times \text{free\_throw\_percentage}$$

```
summary(pos_ftp_main_mod)
```

```
##
## Call:
## lm(formula = log_scoring_efficiency ~ condensed_position + free_throw_percentage,
##     data = box_scores)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.72147 -0.26442  0.03009  0.31646  1.97831
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   0.07999    0.01825   4.383 1.18e-05 ***
## condensed_positionForward    -0.06894    0.01780  -3.872 0.000108 ***
## condensed_positionGuard      -0.17959    0.01717 -10.460  < 2e-16 ***
## condensed_positionPoint Guard -0.15352    0.04580  -3.352 0.000804 ***
## free_throw_percentage         0.36402    0.01226  29.683  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4832 on 16763 degrees of freedom
##   (11469 observations deleted due to missingness)
## Multiple R-squared:  0.05887,    Adjusted R-squared:  0.05864
## F-statistic: 262.1 on 4 and 16763 DF,  p-value: < 2.2e-16
```

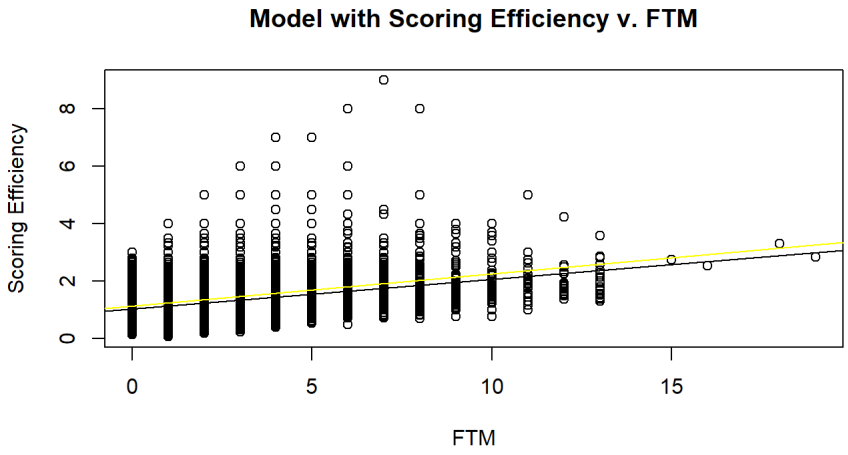
```
# transformed plot
plot(y=box_scores$log_scoring_efficiency, x=box_scores$free_throw_percentage, main = "Transformed Model with Log Scoring Eff
iciency v. CPP & FT%", xlab = "Condensed Player Position", ylab = "Log Scoring Efficiency")
abline(b =  0.36756 , a =  0.08067, col="blue")
abline(b = 0.36756 , a =  0.08067 -0.07397 , col="purple")
abline(b = 0.36756 , a =  0.08067 -0.18323, col="red")
abline(b =  0.36756 , a =  0.08067 -0.16445 , col="orange")
```



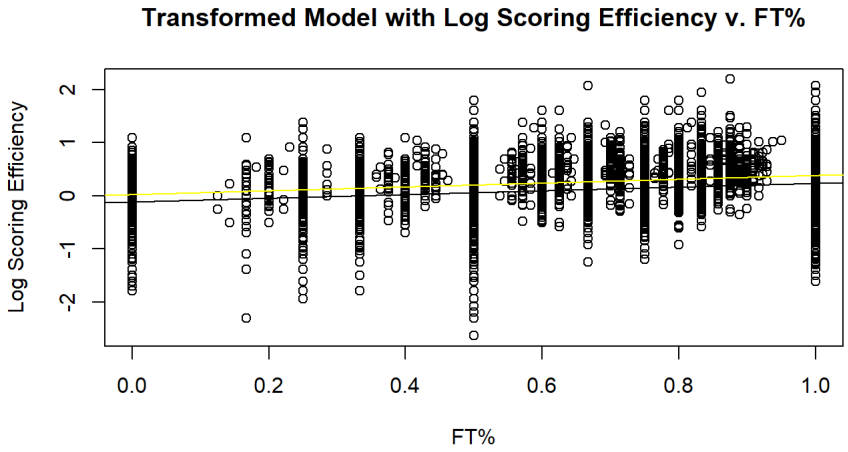
Log Scoring Efficiency Interaction Effects Model By IatHome, Free Throw Percentage, and their interaction (mh_int_mod):

$$\widehat{\text{log\_scoring\_efficiency}} = -0.11743 + 0.35613 \times \text{IatHome} + 0.14534 \times \text{free\_throw\_percentage} + -0.02986 \times \text{IatHome} \times \text{free\_throw\_percenta}$$

```
# untransformed plot
nt_mh_int_mod <- lm(scoring_efficiency ~ IatHome * free_throws_made, data = box_scores)
plot(y=box_scores$scoring_efficiency, x=box_scores$free_throws_made, main = "Model with Scoring Efficiency v. FTM", xlab =
"FTM", ylab = "Scoring Efficiency")
abline(b = 0.102903, a =  1.046266, col="black") # (0) = Away
abline(b =  0.111846, a =  1.151529, col="yellow") # (1) = Home
```




```
# transformed plot
plot(y=box_scores$log_scoring_efficiency, x=box_scores$free_throw_percentage, main = "Transformed Model with Log Scoring Eff
iciency v. FT%", xlab = "FT%", ylab = "Log Scoring Efficiency")
abline(b = 0.35613 , a =  -0.11743, col="black") # (0) = Away
abline(b =  0.35613 , a =  0.02791, col="yellow") # (1) = Home
```



Relevant Features & Issues

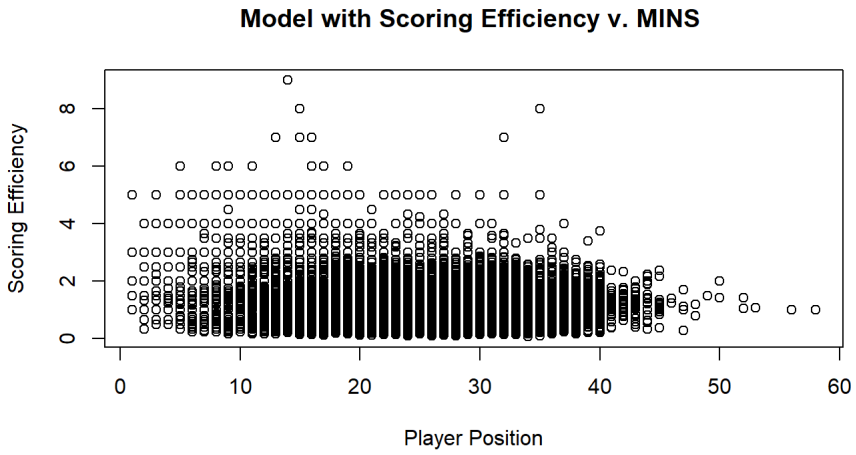
In my initial model, I did not seem to follow linearity patterns because of many extreme outliers. These outliers are the same from previous models where a player will not shoot many field goals will make/shot free throws which dramatically affected their scoring efficiency. I fixed this by transforming the scoring efficiency with log and turned to free throw percentage.

I chose to omit the interaction term in my final model because the summary output indicated a high p-value for this term. This high p-value suggests that there is not statistically significant difference in the percentage of free throws made between games played at home and those played away, according to my model's predictions. I would expect this from Division 1 Women's Basketball players as these individuals are extremely talented, but it is interesting to see in my initial mode, the difference in quantity of free throws made at home vs away seems to be statistically discernible.

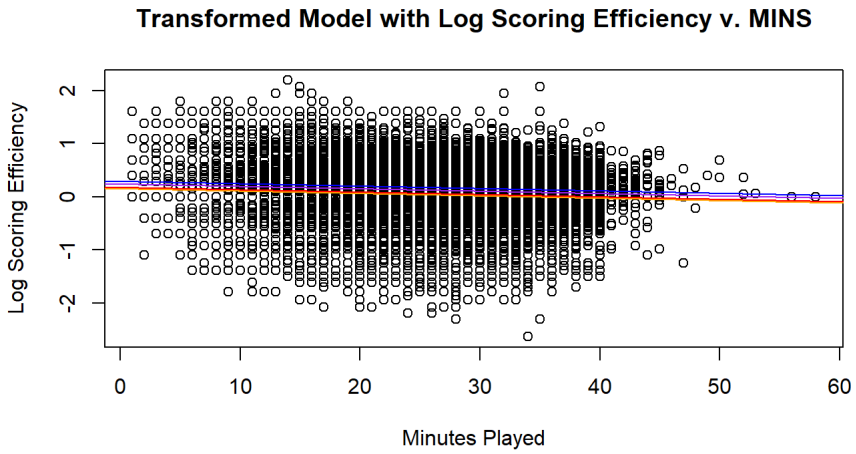
Log Scoring Efficiency Interaction Effects Model By Minutes and Athlete Position (mtp_int_mod):

$$\widehat{\text{log\_scoring\_efficiency}} = 0.1842022 + -0.0053057 \times \text{Minutes} + -0.0621920 \times \text{free\_throw\_percentage} + 0.07075 \times \text{Ipg} + -0.16820 \times \text{Ig} + -$$

```
# untransformed plot
nt_mtp_main_mod <- lm(scoring_efficiency ~ athlete_position_name + minutes, data = box_scores)
plot(y=box_scores$scoring_efficiency, x=box_scores$minutes, main = "Model with Scoring Efficiency v. MINS ", xlab = "Player
Position", ylab = "Scoring Efficiency")
```




```
# transformed plot
plot(box_scores$minutes, box_scores$log_scoring_efficiency, main = "Transformed Model with Log Scoring Efficiency v. MINS",
xlab = "Minutes Played", ylab = "Log Scoring Efficiency")
abline(b = -0.0045437 , a =  0.2970554, col="blue")
abline(b = -0.0045437 , a =  0.2970554 -0.0555350, col="purple")
abline(b = -0.0045437 , a =  0.2970554 -0.1204884, col="red")
abline(b = -0.0045437 , a =  0.2970554-0.1398051, col="orange")
```



Relevant Features & Issues

We can see that as minutes played increases, scoring efficiency slightly decreases. Athlete positions typically have a lower scoring efficiency compared to the Center position, with Guards exhibiting the most substantial negative impact. However, the models explain a relatively small portion of the variance in scoring efficiency, suggesting other factors not included in the models also play a significant role. Despite the low R-squared values, the negative relationship between minutes played and scoring efficiency is a robust finding.

7. Evaluating Models

```
summary(ftm_main_mod)
```

```
##
## Call:
## lm(formula = log_scoring_efficiency ~ free_throw_percentage,
##     data = box_scores)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.76883 -0.28997  0.03545  0.30555  1.93831
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -0.04242    0.00932  -4.551 5.37e-06 ***
## free_throw_percentage  0.34438    0.01229  28.023  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4868 on 16766 degrees of freedom
##   (11469 observations deleted due to missingness)
## Multiple R-squared:  0.04474,    Adjusted R-squared:  0.04469
## F-statistic: 785.3 on 1 and 16766 DF,  p-value: < 2.2e-16
```

```
summary(condensed_pos_mod)
```

```
##
## Call:
## lm(formula = log_scoring_efficiency ~ condensed_position, data = box_scores)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.70692 -0.29101  0.03471  0.33760  2.12936
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    0.21403    0.01388  15.419  < 2e-16 ***
## condensed_positionForward     -0.06642    0.01499  -4.432 9.37e-06 ***
## condensed_positionGuard       -0.14616    0.01441 -10.141  < 2e-16 ***
## condensed_positionPoint Guard -0.17317    0.03861  -4.485 7.32e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5221 on 28233 degrees of freedom
## Multiple R-squared:  0.007369,   Adjusted R-squared:  0.007263
## F-statistic: 69.86 on 3 and 28233 DF,  p-value: < 2.2e-16
```

```
summary(fs_int_mod)
```

```
##
## Call:
## lm(formula = log_scoring_efficiency ~ free_throw_percentage *
##     condensed_position, data = box_scores)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.71721 -0.26998  0.02721  0.31165  1.97520
##
## Coefficients:
##                                                 Estimate Std. Error
## (Intercept)                                     0.042687   0.037318
## free_throw_percentage                           0.422495   0.052481
## condensed_positionForward                      -0.005782   0.040391
## condensed_positionGuard                        -0.156369   0.039272
## condensed_positionPoint Guard                   0.086427   0.121801
## free_throw_percentage:condensed_positionForward -0.097954   0.056620
## free_throw_percentage:condensed_positionGuard   -0.038832   0.054795
## free_throw_percentage:condensed_positionPoint Guard -0.331743   0.154545
##                                                 t value Pr(>|t|)
## (Intercept)                                       1.144   0.2527
## free_throw_percentage                             8.050 8.80e-16 ***
## condensed_positionForward                        -0.143   0.8862
## condensed_positionGuard                          -3.982 6.87e-05 ***
## condensed_positionPoint Guard                     0.710   0.4780
## free_throw_percentage:condensed_positionForward  -1.730   0.0836 .
## free_throw_percentage:condensed_positionGuard    -0.709   0.4785
## free_throw_percentage:condensed_positionPoint Guard  -2.147   0.0318 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4831 on 16760 degrees of freedom
##   (11469 observations deleted due to missingness)
## Multiple R-squared:  0.05942,    Adjusted R-squared:  0.05902
## F-statistic: 151.2 on 7 and 16760 DF,  p-value: < 2.2e-16
```

```
summary(pos_ftp_main_mod)
```

```
##
## Call:
## lm(formula = log_scoring_efficiency ~ condensed_position + free_throw_percentage,
##     data = box_scores)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.72147 -0.26442  0.03009  0.31646  1.97831
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    0.07999    0.01825   4.383 1.18e-05 ***
## condensed_positionForward     -0.06894    0.01780  -3.872 0.000108 ***
## condensed_positionGuard       -0.17959    0.01717 -10.460  < 2e-16 ***
## condensed_positionPoint Guard -0.15352    0.04580  -3.352 0.000804 ***
## free_throw_percentage          0.36402    0.01226  29.683  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4832 on 16763 degrees of freedom
##   (11469 observations deleted due to missingness)
## Multiple R-squared:  0.05887,    Adjusted R-squared:  0.05864
## F-statistic: 262.1 on 4 and 16763 DF,  p-value: < 2.2e-16
```

```
summary(mh_int_mod)
```

```
##
## Call:
## lm(formula = log_scoring_efficiency ~ free_throw_percentage *
##     IatHome, data = box_scores)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.83289 -0.24261  0.02931  0.31166  1.88161
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  -0.11516    0.01315  -8.756  < 2e-16 ***
## free_throw_percentage         0.35088    0.01752  20.024  < 2e-16 ***
## IatHome                       0.14662    0.01848   7.933 2.27e-15 ***
## free_throw_percentage:IatHome -0.02614    0.02439  -1.072    0.284
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4826 on 16764 degrees of freedom
##   (11469 observations deleted due to missingness)
## Multiple R-squared:  0.06136,    Adjusted R-squared:  0.0612
## F-statistic: 365.3 on 3 and 16764 DF,  p-value: < 2.2e-16
```

```
summary(mtp_main_mod)
```

```
##
## Call:
## lm(formula = log_scoring_efficiency ~ minutes + condensed_position,
##     data = box_scores)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.65851 -0.28223  0.04954  0.35619  2.08833
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   0.2995082  0.0152914  19.587  < 2e-16 ***
## minutes                      -0.0046613  0.0003548 -13.139  < 2e-16 ***
## condensed_positionForward    -0.0548985  0.0149660  -3.668 0.000245 ***
## condensed_positionGuard      -0.1215665  0.0144914  -8.389  < 2e-16 ***
## condensed_positionPoint Guard -0.1452522  0.0385532  -3.768 0.000165 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5205 on 28232 degrees of freedom
## Multiple R-squared:  0.0134, Adjusted R-squared:  0.01326
## F-statistic: 95.87 on 4 and 28232 DF,  p-value: < 2.2e-16
```

R Squared & Adjusted R Squared:

1. mh_int_mod – Multiple R-squared: 0.06139, Adjusted R-squared: 0.06121
2. fs_int_mod – Multiple R-squared: 0.06036, Adjusted R-squared: 0.05994
3. pos_ftp_main_mod – Multiple R-squared: 0.05983, Adjusted R-squared: 0.05959
4. ftm_main_mod – Multiple R-squared: 0.04571, Adjusted R-squared: 0.04565
5. mtp_main_mod – Multiple R-squared: 0.01281, Adjusted R-squared: 0.01266
6. condensed_pos_mod – Multiple R-squared: 0.007077, Adjusted R-squared: 0.006965

Basketball is a dynamic and complex sport where performance is influenced by a multitude of factors, both quantifiable and un-quantifiable. Variables like player fatigue, psychological factors, defensive pressure, and game situations play significant roles in shooting efficiency but are often difficult to quantify accurately. Many of my models are extremely simple in the large scope of what may impact shooting efficiency, so I wouldn't expect large R-squared values.

Residual Standard Error (RSE):

1. mh_int_mod – Residual standard error: 0.4824 on 15781 degrees of freedom
2. fs_int_mod – Residual standard error: 0.4827 on 15777 degrees of freedom
3. pos_ftp_main_mod – Residual standard error: 0.4828 on 15780 degrees of freedom
4. ftm_main_mod – Residual standard error: 0.4863 on 15783 degrees of freedom
5. mtp_main_mod – Residual standard error: 0.5198 on 26568 degrees of freedom
6. condensed_pos_mod – Residual standard error: 0.5213 on 26569 degrees of freedom

The RSE reflects the average prediction error, with a lower RSE indicating better predictive accuracy. The models mh_int_mod, fs_int_mod, and pos_ftp_main_mod demonstrate the lowest RSEs, suggesting they are comparatively more accurate in predicting log scoring efficiency in basketball.

F-Statistic:

```
anova(pos_ftp_main_mod, ftm_main_mod)
```

```
## Analysis of Variance Table
##
## Model 1: log_scoring_efficiency ~ condensed_position + free_throw_percentage
## Model 2: log_scoring_efficiency ~ free_throw_percentage
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  16763 3914.2
## 2  16766 3973.0 -3   -58.743 83.857 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This result suggests that adding condensed_position to the model with free_throw_percentage significantly improves the model's fit. In other words, condensed_position contributes to explaining the variance in log_scoring_efficiency above and beyond what is explained by free_throw_percentage alone.

```
anova(ftm_main_mod, fs_int_mod)
```

```
## Analysis of Variance Table
##
## Model 1: log_scoring_efficiency ~ free_throw_percentage
## Model 2: log_scoring_efficiency ~ free_throw_percentage * condensed_position
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  16766 3973.0
## 2  16760 3911.9  6    61.026 43.576 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

A significant F-statistic of 40.984 with a very small p-value (< 2.2e-16). This means that the additional predictors in Model 2 significantly improve the model's fit compared to Model 1. However, I recognize many of the intercepts in the interaction model are statistically significant.

```
anova(mtp_main_mod, condensed_pos_mod)
```

```
## Analysis of Variance Table
##
## Model 1: log_scoring_efficiency ~ minutes + condensed_position
## Model 2: log_scoring_efficiency ~ condensed_position
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  28232 7650.0
## 2  28233 7696.8 -1   -46.776 172.62 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

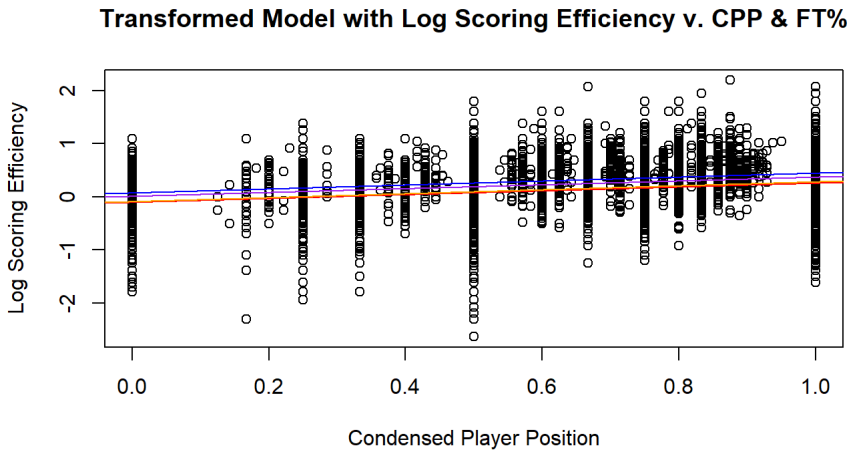
These results indicate that 'minutes' is a significant predictor for log scoring efficiency, and its inclusion in the model alongside 'condensed_position' provides a significantly better fit compared to using 'condensed_position' alone.

8. Final Model

a. Log Scoring Efficiency Main Effects Model Athlete Position and Free Throw Percentage (pos_ftp_main_mod):

$$\widehat{\text{log\_scoring\_efficiency}} = 0.08067 + -0.16445 \times \text{Ipg} + -0.18323 \times \text{Ig} + -0.07397 \times \text{If} + 0.36756 \times \text{free\_throw\_percentage}$$

```
# transformed plot
plot(y=box_scores$log_scoring_efficiency, x=box_scores$free_throw_percentage, main = "Transformed Model with Log Scoring Eff
iciency v. CPP & FT%", xlab = "Condensed Player Position", ylab = "Log Scoring Efficiency")
abline(b =  0.36756 , a =  0.08067, col="blue")
abline(b = 0.36756 , a =   0.08067 -0.07397 , col="purple")
abline(b = 0.36756 , a =   0.08067 -0.18323, col="red")
abline(b =  0.36756 , a =  0.08067 -0.16445 , col="orange")
```



I believe this model is the best choice as my final model for these reasons: - Statistical Significance: All predictors in the model are statistically significant, as indicated by their p-values.

- Practical Relevance: The model includes important factors that are intuitively relevant to scoring efficiency in basketball - player position and free throw accuracy.
- Balance Between Complexity and Interpretability: The model is complex enough to include key factors but not so complex as to lose interpretability.
- Overall Performance in Fit & Assumptions: I have models that would excel in one Fit but do the poorest in meeting linearity assumptions. I found this to be the best model because it was okay in both assumptions and fit.

b. Interpretation for a Non-Statistical Audience:

- Intercept (0.08067): This is the baseline level of log scoring efficiency when all other variables are zero. In this context, it represents the base level of efficiency for a Center, with a free throw percentage of zero.
- condensed_position Forward (-0.07397): Being a Forward is associated with a decrease in log scoring efficiency compared to the a Center. The negative sign indicates that Forwards, on average, have lower scoring efficiency than a Center.
- condensed_position Guard (-0.18323): Guards have the most significant decrease in scoring efficiency when compared to the Center position. This suggests that Guards, on average, tend to have lower efficiency in scoring than a Center.
- condensed_position Point Guard (-0.16445): Point Guards also show a decrease in efficiency compared to the Center position, though the impact is slightly less than that for Guards.
- free_throw_percentage (0.36756): This positive coefficient indicates that an increase in free throw percentage is associated with an increase in scoring efficiency. Essentially, better free throw shooters tend to have higher overall scoring efficiency.

d. Caveats/Limitations

- Low Explanatory Power: It's noteworthy that the model accounts for approximately 6% of the variance in log scoring efficiency. This indicates that a substantial portion of the variance is driven by factors not included in the model. Hence, conclusions drawn from this model should be made with an understanding of its limited explanatory scope.
- Association, Not Causation: The findings from this model should be interpreted as associations rather than causal relationships. The nature of statistical modeling used here does not allow us to infer causation, and thus, any changes or recommendations based on this study should be considered with this limitation in mind.
- Data Biases: Potential biases within the data set, such as an over representation of certain player positions or specific skills, might have influenced the model's predictions. These biases can skew the understanding of which factors are most important in determining scoring efficiency.

e. Recommendations for Future Research

- Enriching the Data set: I choose to do this data analysis on box scores because this data is kept across all NCAA divisions and I knew if I wanted to translate these models to use game data for my personal use, I would be able to do that. However since box scores only show a portion of what actually influences scoring efficiency, I would be interested in looking into other predictors.

Future studies should aim to incorporate a more diverse range of data points. This could include variables like player fatigue, defensive assignments, and team strategies, which are likely to have a significant impact on scoring efficiency but were not captured in the current data set.

- Exploring Advanced Modeling Techniques: There is potential in exploring more complex modeling approaches. Non-linear models might be more adept at capturing the nuanced relationships between different factors influencing scoring efficiency.