Predicting NFL QB's Fantasy Points Using Tracking Data

Introduction

In the United States, football is considered the king of sports, as year after year viewing metrics for the NFL dwarfs their fellow competitors. According to viewing metrics on statista.com, the most watched sports league in the USA in 2023 was the NFL, with over 947 billion minutes watched. The next closest was the MLB with a over 329 billion minutes watched in 2023, nearly one third of the NFL. As the NFL continues its stranglehold over TV viewers, it's no surprise that fantasy football has become quite popular. According to Fantasy Sports & Gaming Association (FSGA), as of 2023, there are approximately 41 million people that participate in fantasy football. Perhaps more than ever participants are looking for any competitive advantage they can find and with advancements in technology, there may be an edge to be had by evaluating new data to potentially determine what drives a player's performance on the field. In this paper, we will be looking at Next Generation Stats, or NGS, in attempt to determine if there's any useful variables to predict a quarterback's fantasy points total.

NGS is "NFL player tracking, also known as Next Gen Stats, is the capture of real time location data, speed and acceleration for every player, every play on every inch of the field. Sensors throughout the stadium track tags placed on players' shoulder pads, charting individual movements within inches." For this study, we are going to specifically look at 10 different metrics that measures certain aspects of how the QB is performing on the field. These metrics include (courtesy of Nextgenstats.com):

"Time To Throw (TT)

Time to Throw measures the average amount of time elapsed from the time of snap to throw on every pass attempt for a passer (sacks excluded).

Average Completed Air Yards (CAY) and Average Intended Air Yards (IAY)

Air Yards is the vertical yards on a pass attempt at the moment the ball is caught in relation to the line of scrimmage. CAY shows the average Air Yards a passer throws on completions, and IAY shows the average Air Yards a passer throws on all attempts. This metric shows how far the ball is being thrown 'downfield'. Air Yards is recorded as a negative value when the pass is behind the Line of Scrimmage. Additionally Air Yards is calculated into the back of the end zone to better evaluate the true depth of the pass.

Average Air Yards Differential (AYD)

Air Yards Differential is calculated by subtracting the passer's average Intended Air Yards from his average Completed Air Yards. This stat indicates if he is on average attempting deep passes than he on average completes.

Longest Completed Air Distance (LCAD)

Air Distance is the amount of yards the ball has traveled on a pass, from the point of release to the point of reception (as the crow flies). Unlike Air Yards, Air Distance measures the actual distance the passer throws the ball.

Aggressiveness (AGG%)

Aggressiveness tracks the amount of passing attempts a quarterback makes that are into tight coverage, where there is a defender within 1 yard or less of the receiver at the time of completion or incompletion. AGG is shown as a % of attempts into tight windows over all passing attempts.

Air Yards to the Sticks (AYTS)

Air Yards to the Sticks shows the amount of Air Yards ahead or behind the first down marker on all attempts for a passer. The metric indicates if the passer is attempting his passes past the 1st down marker, or if he is relying on his skill position players to make yards after catch.

Completion Probability

The probability of a pass completion, based on numerous factors such as receiver separation from the nearest defender, where the receiver is on the field, the separation the passer had at time of throw from the nearest pass rusher, and more.

Expected Completion Percentage (xCOMP)

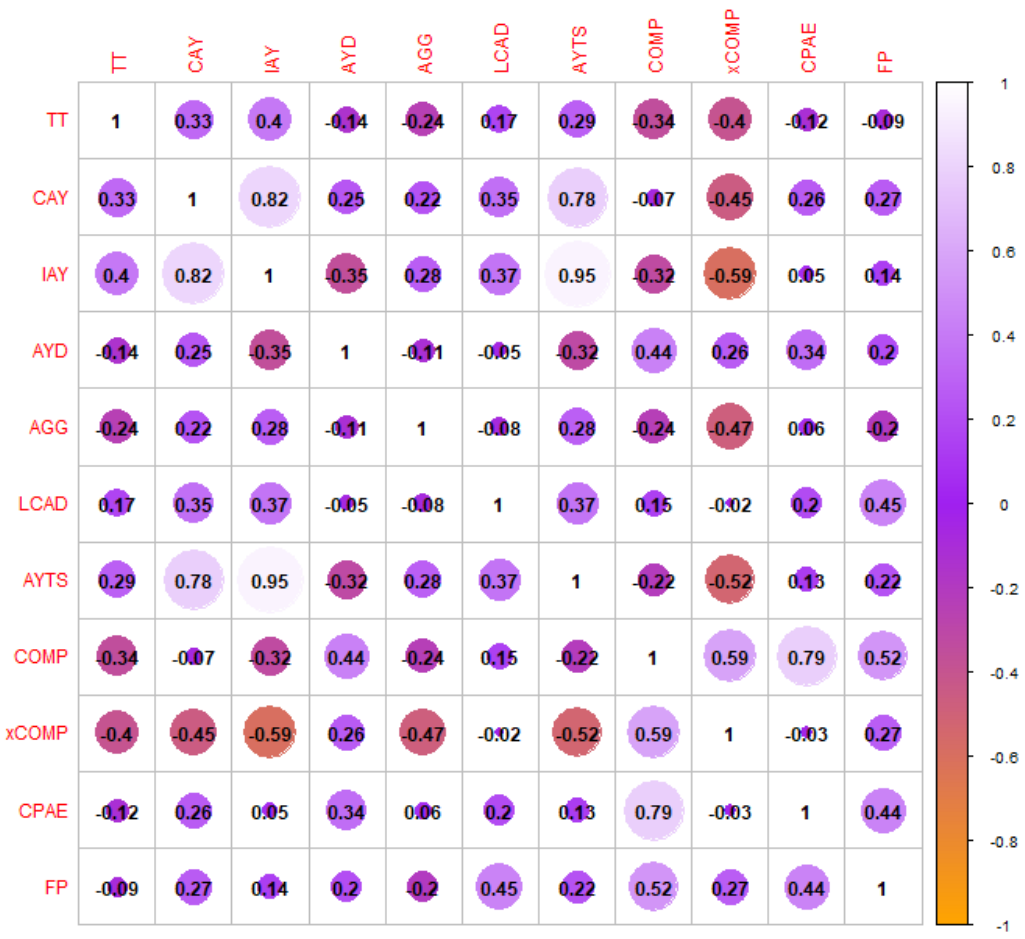
Using a passer's Completion Probability on every play, determine what a passer's completion percentage is expected to be.

Completion Percentage Above Expectation (+/-)

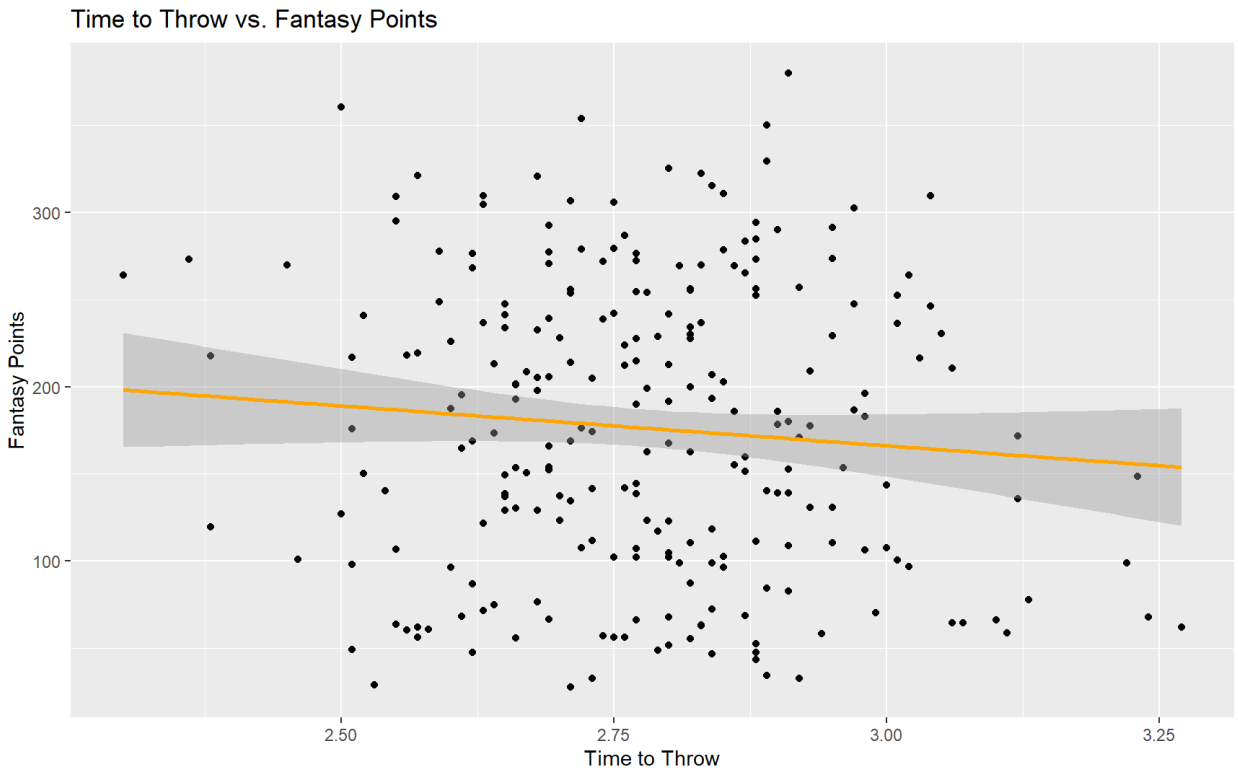
A passer's actual completion percentage compared to their Expected Completion Percentage."

Exploratory Data Analysis

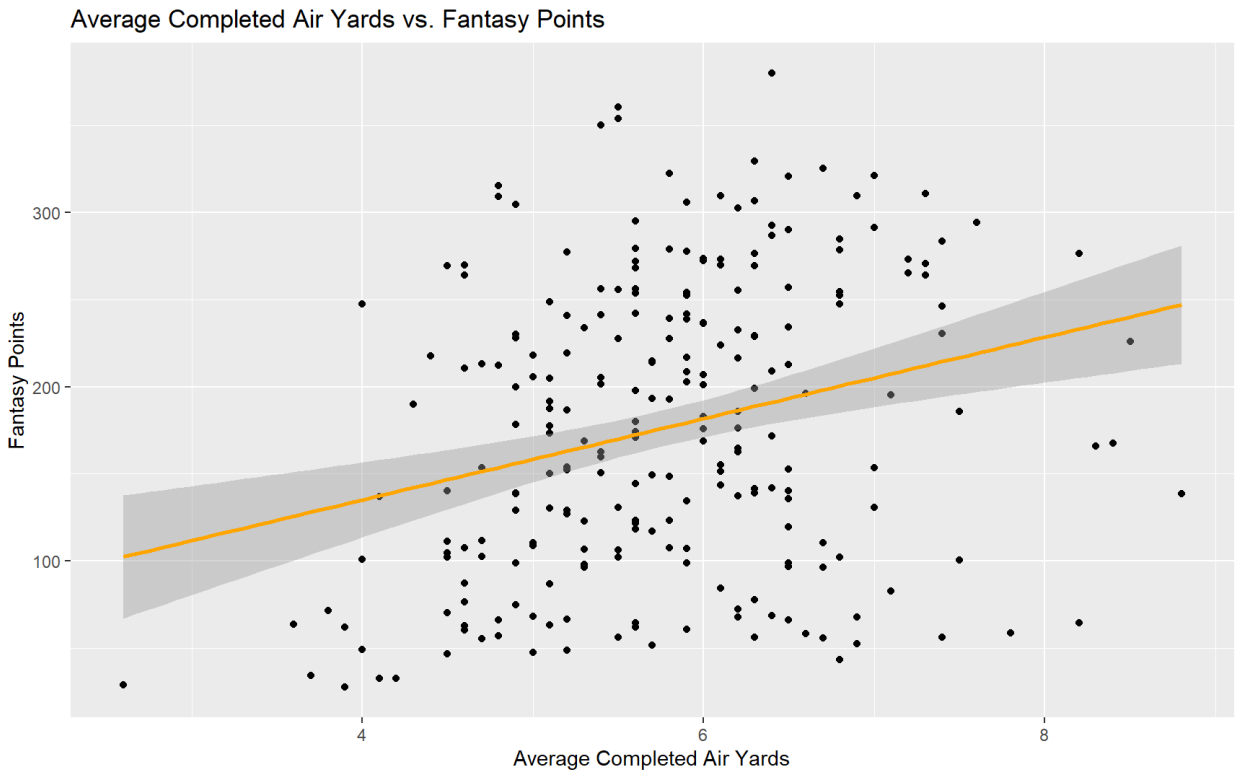
To begin, since we are attempting to see which metrics could be useful in predicting a QB's fantasy points total, it would be a good idea to produce a correlation matrix between each potential predictor variable and the response variable.



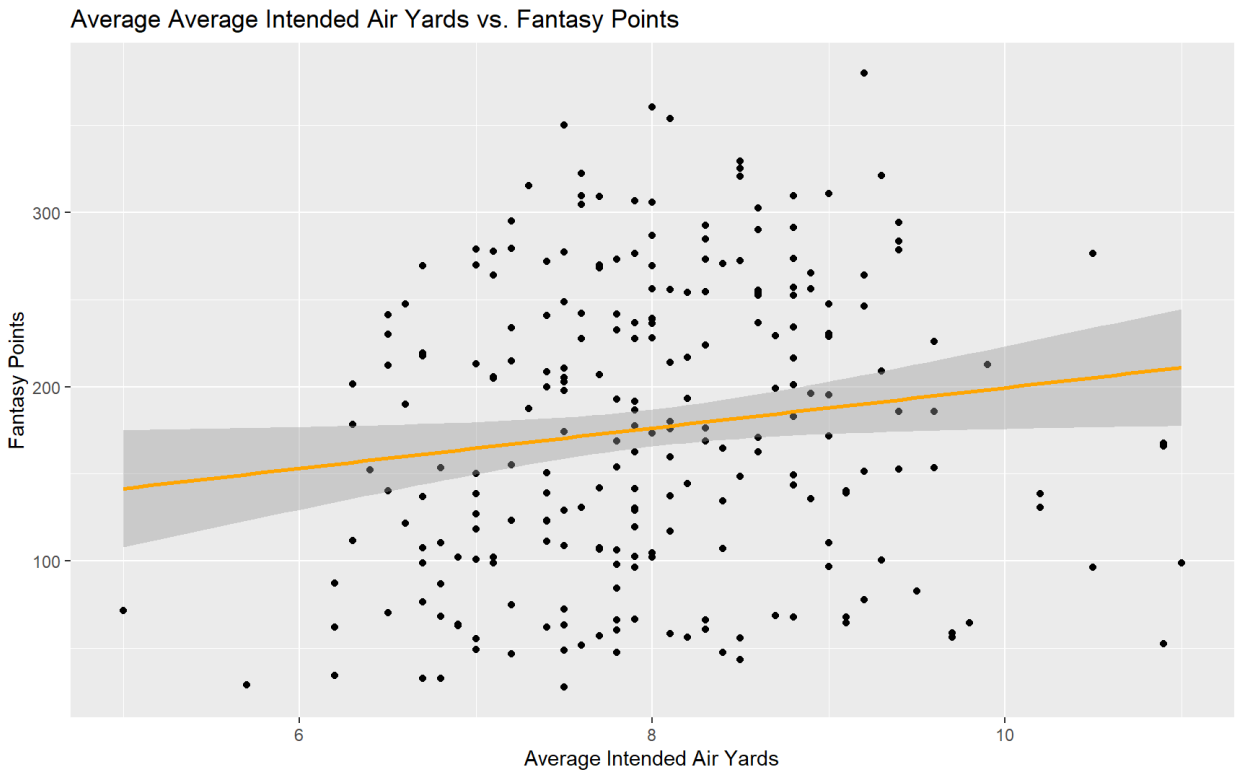
Based on the correlation matrix above, looking at the “FP” column, there doesn’t appear to be any strong correlations, either negative or positive, but a few metrics show promise, in particular COMP, LCAD, and CPAE have roughly a moderate correlation with a QB’s fantasy points. To examine a closer look, let’s see some scatterplots of each potential predictor value and our response variable.



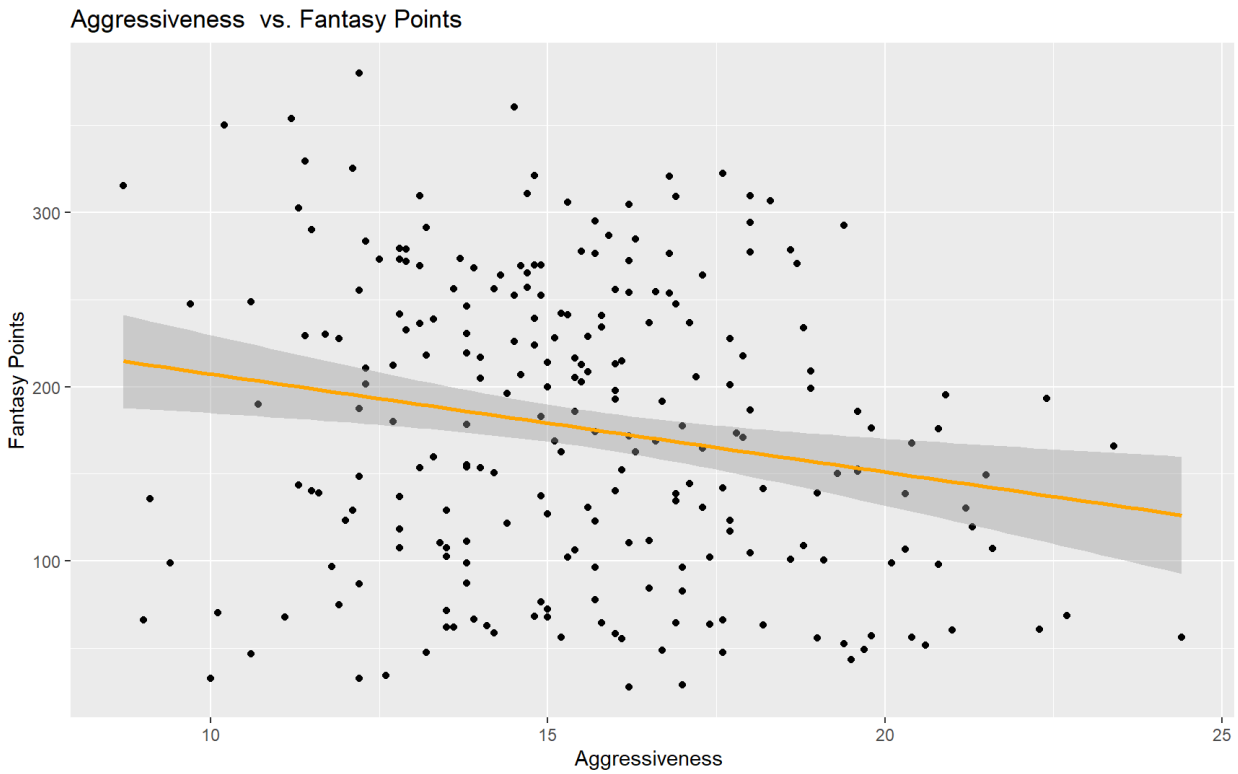
The time to throw versus fantasy points shows a weak, negative relationship. Even though the relationship is weak, the negative relationship is what we sort of expect. Typically, the longer the QB holds onto the ball, the easier it will be for a defensive line or blitzer from the back seven will be to beat their opponent and pressure or sack the QB. Also, getting rid of the ball quickly can be indicative of well a QB can read a defense and process quickly where to go with the ball. When a QB gets roughly past 3.15 with time to throw, we notice four data points below the average, perhaps confirming that QBs who hold the ball too long will struggle to consistently produce points.



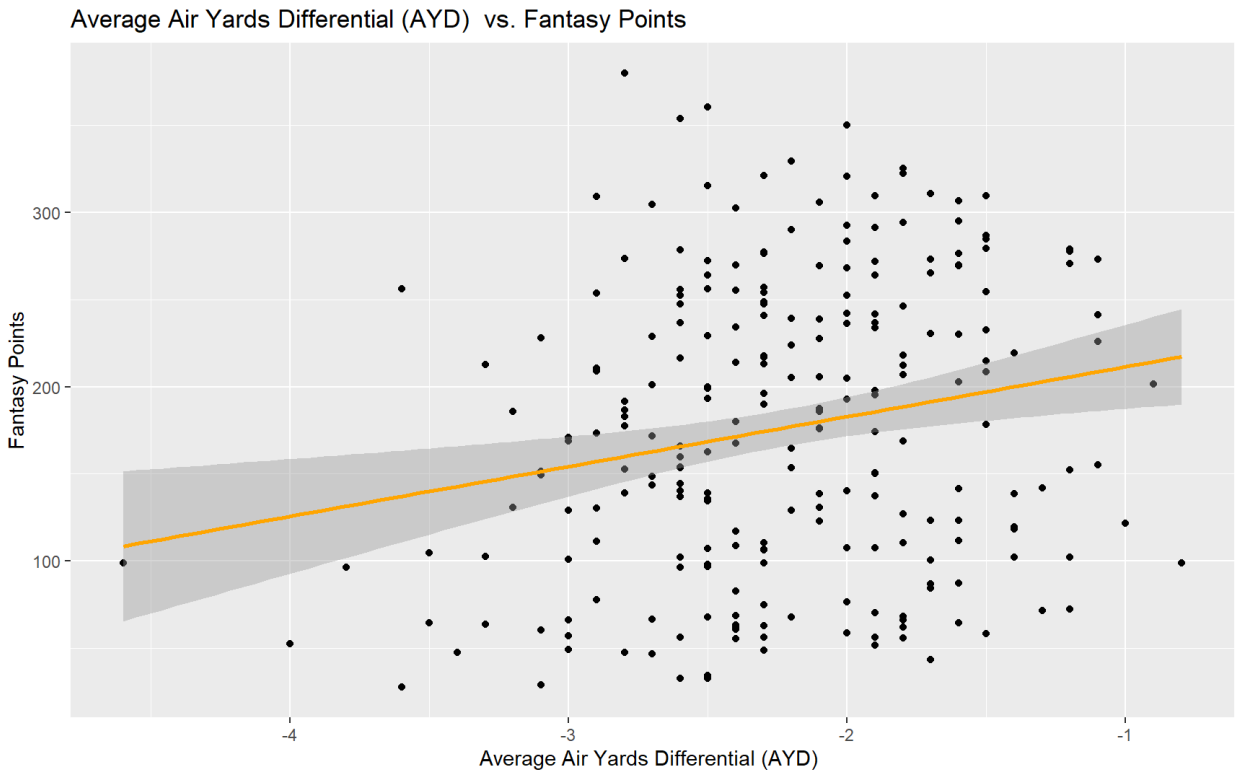
Average completed air yards versus fantasy points seems to indicate a somewhat moderate, positive linear relationship. Similar to time to throw, even though the relationship isn't strong, it's a pattern that we expect. When a QB consistently completes passes further the field, they will likely generate more yards than a QB that checks down and take the safe few yards, but throwing the ball downfield is often rewarded with significantly more yards and thus points.



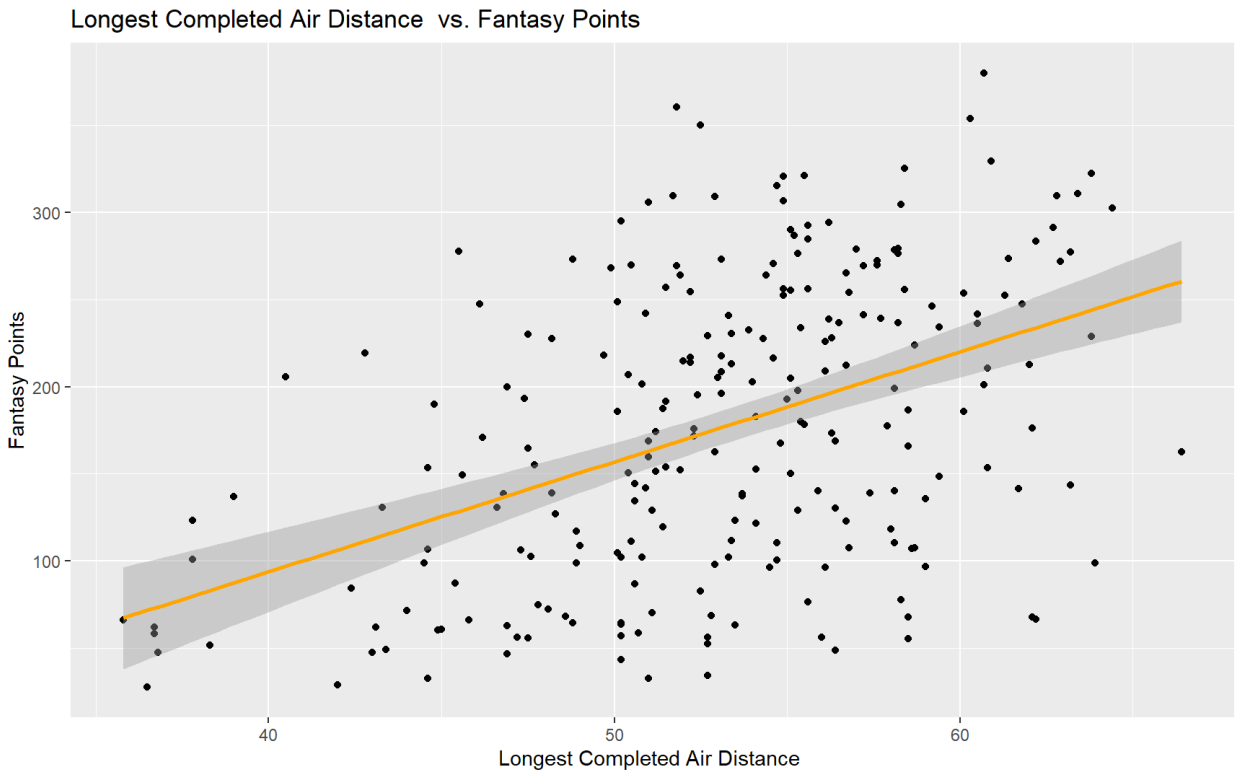
Average intended air yards has a weak, but positive linear relationship with fantasy points. This is somewhat a surprise as with CAY we would expect a pretty strong relationship with QBs that are more willing to throw further down the field for more yards, but perhaps passers that tend to really stretch the field, roughly past 10 yards, may struggle with accuracy and an inability to generate points off of other less risky and rewarding passes. A good example of this may be Jameis Winston, who can rack up a ton of yardage by throwing the ball deep, but struggles mightily with throwing interceptions and passes short to intermediate, which will usually lead to him being benched for several games.



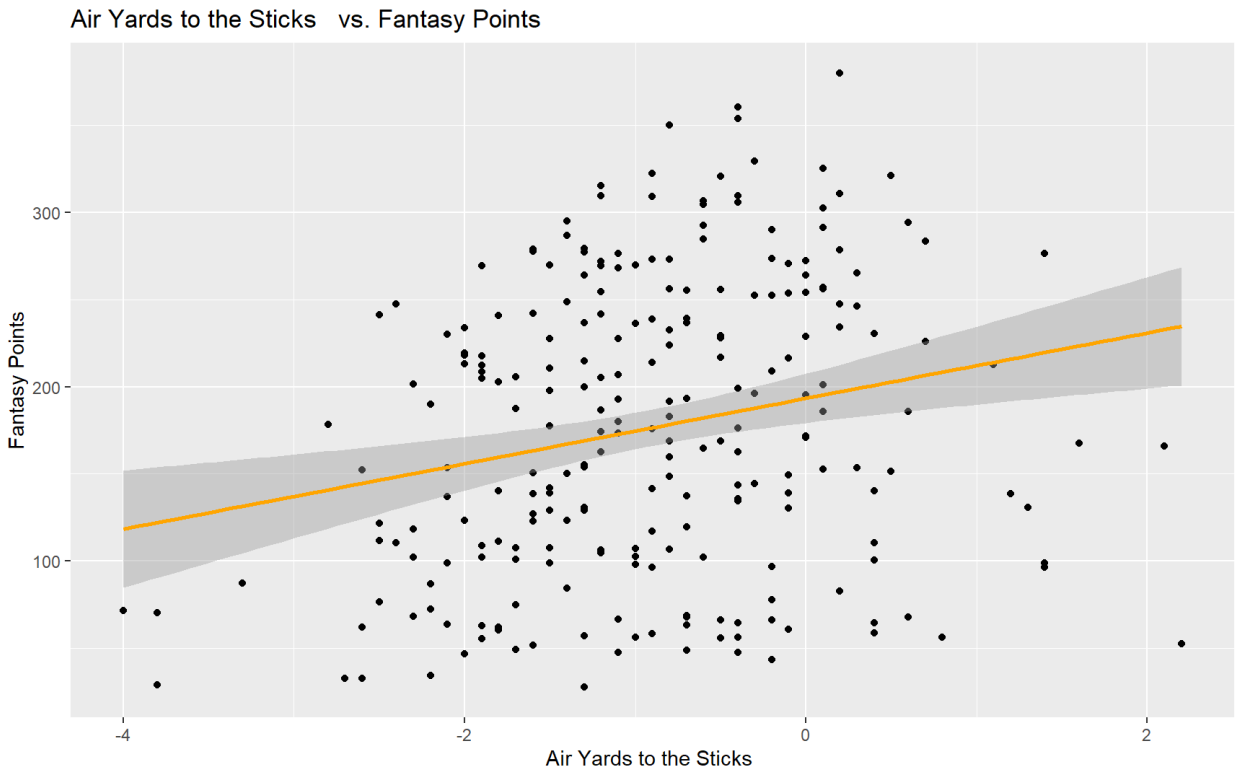
Aggressiveness versus fantasy points has a somewhat moderate, weak relationship. This is pretty surprising since one could argue that a QB that is able to fit passes into tight windows would generally be a good fantasy QB that accumulates a bountiful amount of yards and touchdowns, but perhaps a good fantasy QB is one that doesn't need to take as many chances because they can properly diagnose a defense pre or post snap to get the ball to an open receiver.



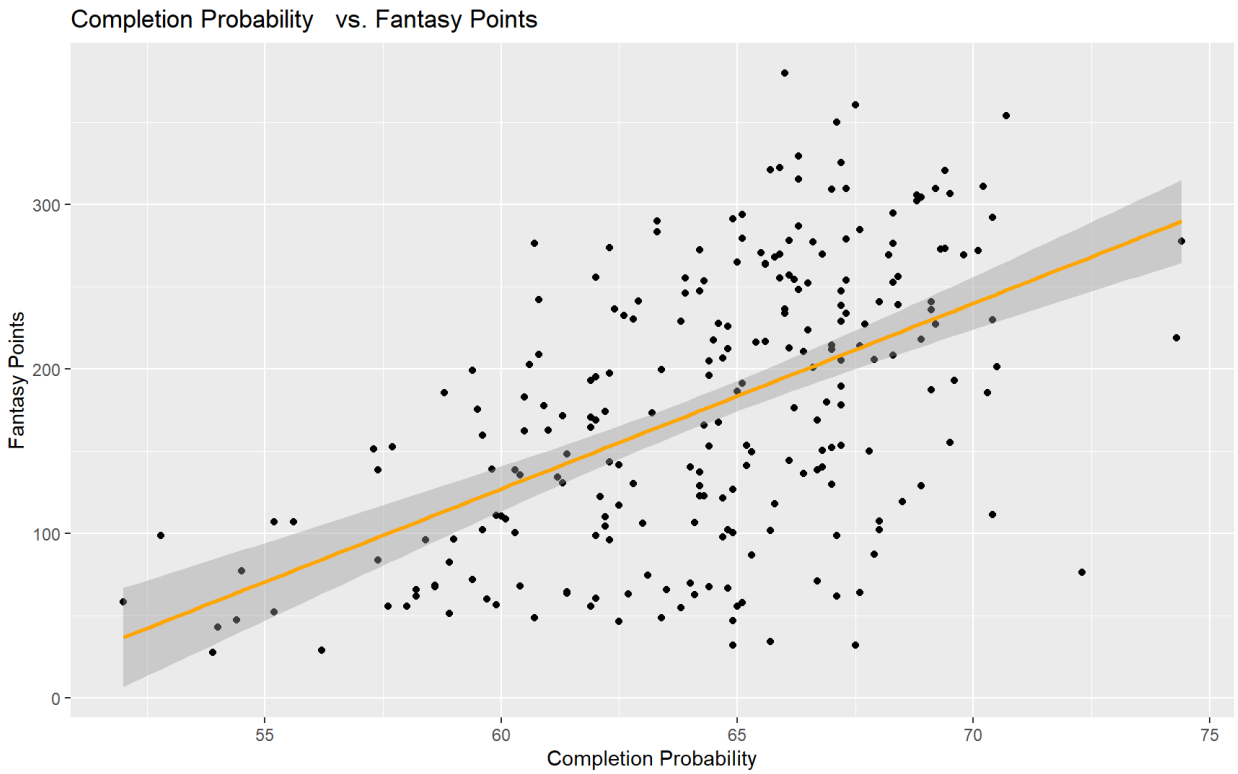
Average air yards differential shows a somewhat moderate, positive linear relationship. With this being the difference in AIY and CAY, it makes sense that it would show a similar pattern to those variables. Though, it is a bit surprising that CAY actually shows a stronger correlation than AYD as one would expect efficiency to play a big factor into the QB fantasy success, but the actual result seems to be more aligned with our response variable.



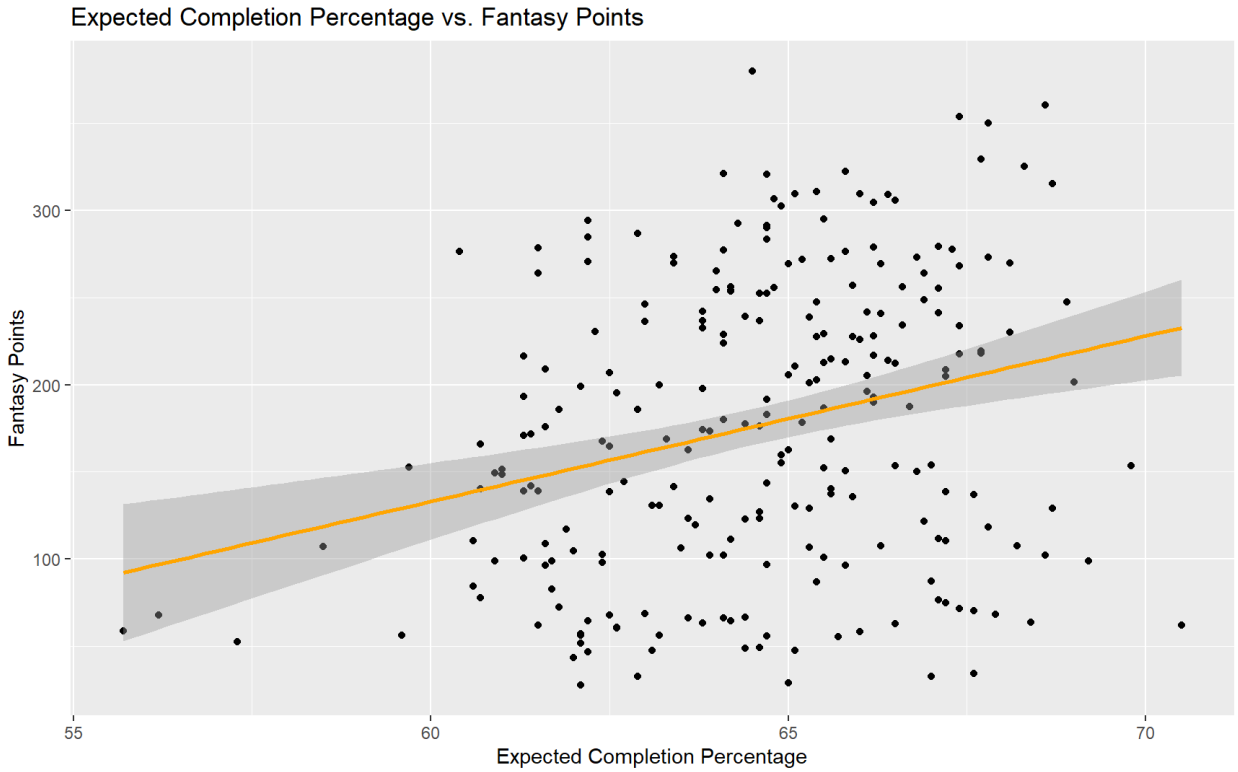
Longest completed air distance versus fantasy points shows a moderate, positive linear relationship. This is a bit of a surprising result because LCAD is merely a measurement of a QB's longest throw in the air that is completed, which doesn't seem to suggest much predictive power. Since several QB's throw roughly 500 to 650 times a year, perhaps it isn't too surprising that those QB's are going to have a good chance of completing a deep pass due to sheer volume.



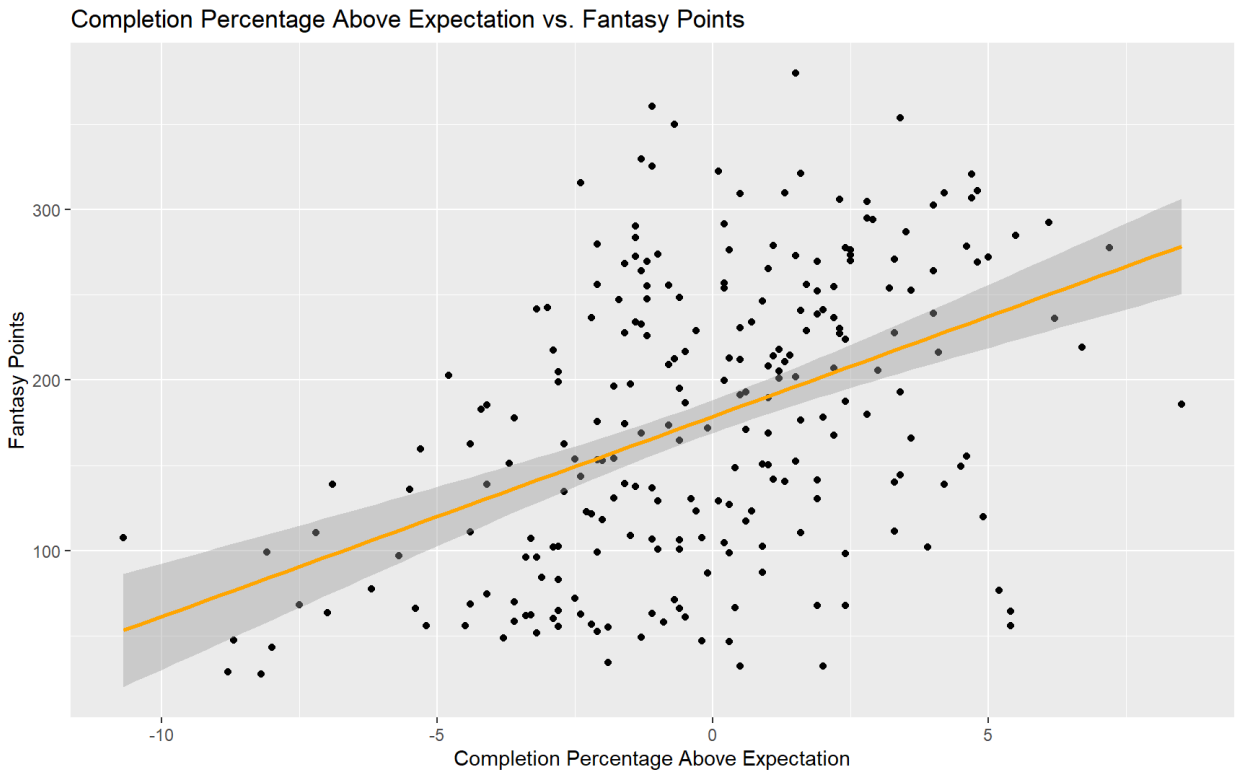
Air yards to the sticks versus fantasy points shows a weak, positive linear relationship. This is a tricky one as we might expect a QB that is aggressively attacking the field for first downs to keep the drive going, but as we saw with the aggressive percentage maybe there's a diminishing return on attempting too many throws past the first down mark as defenses are generally prepared for those type of passes in obvious situations.



Completion probability has a moderate, linear relationship with fantasy points. CP is a pretty unique tracking stat as it measures several different factors of the on field play to generate an estimate of a player completing a pass. Based off of this metric, we see how much potential tracking data has when evaluating a player's performance.



Expected completion percentage versus fantasy points shows a slightly moderate, positive linear relationship with fantasy points. While CP had a moderate to pretty strong relationship, it's a bit odd that xCP doesn't have quite the same strength. Maybe CP does a better job of assessing a player's ability since xCP is somewhat trying to regress CP, perhaps too harshly.



CPAE shows a moderate, positive linear relationship with fantasy points. CPAE having a moderate, linear relationship is to be expected, as similar to CP, CPAE can showcase a QB's skillset better than traditional metrics.

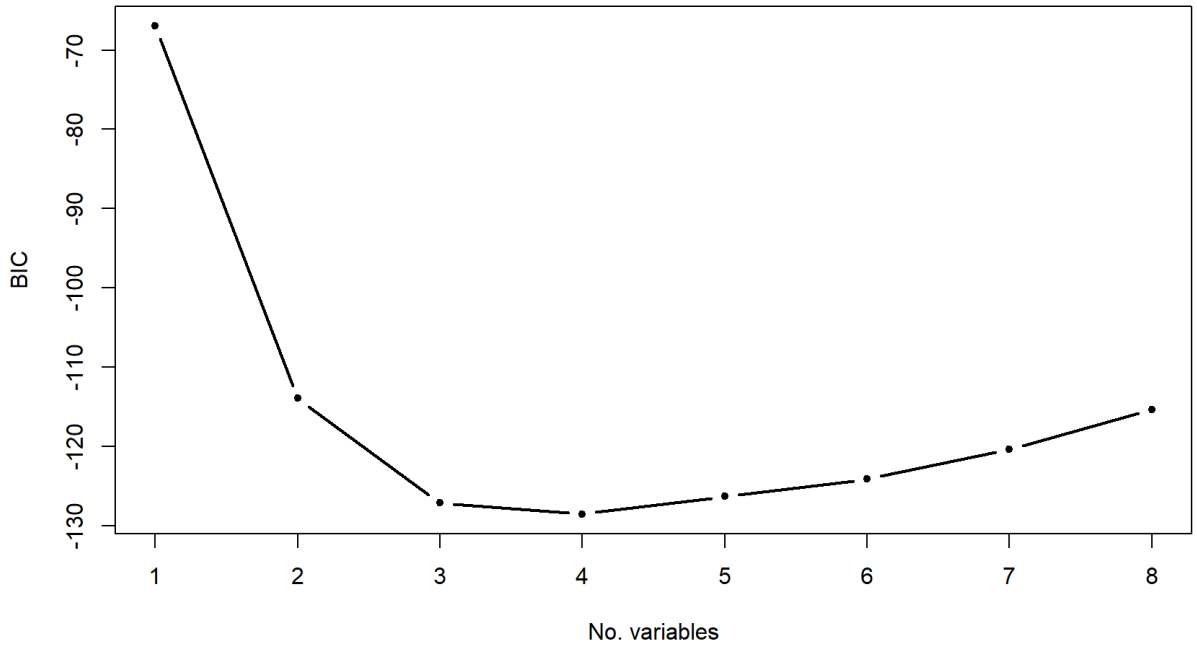
Outline of Analysis

For this report we will look at a multiple linear regression to determine which predictor variables, if any, are useful in predicting a QB's fantasy points through a whole season. Since we are dealing with a continuous, non-discrete response variable, first we'll start off with a multiple linear regression, then attempt regularization techniques such as ridge and lasso regression.

Analysis

```
Call:
lm(formula = FP ~ ., data = ngs)

Residuals:
     Min       1Q   Median       3Q      Max
-148.394  -43.681    4.849   46.205  128.436

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -515.5584   271.5157  -1.899  0.05883 .
TT           -41.5304    33.7945  -1.229  0.22036
CAY           67.6623    78.3517   0.864  0.38872
IAY          -72.3911    77.2948  -0.937  0.34996
AYD          -53.2438    78.4868  -0.678  0.49821
AGG           -4.1452     1.8444  -2.247  0.02556 *
LCAD           3.7582     0.7593   4.950 1.43e-06 ***
AYTS          34.3877    13.1723   2.611  0.00963 **
COMP          10.9368    80.9469   0.135  0.89264
xCOMP          1.0723    81.0934   0.013  0.98946
CPAE          -2.4604    80.7716  -0.030  0.97573
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 61.26 on 231 degrees of freedom
Multiple R-squared:  0.4949,   Adjusted R-squared:  0.473
F-statistic: 22.63 on 10 and 231 DF,  p-value: < 2.2e-16
```

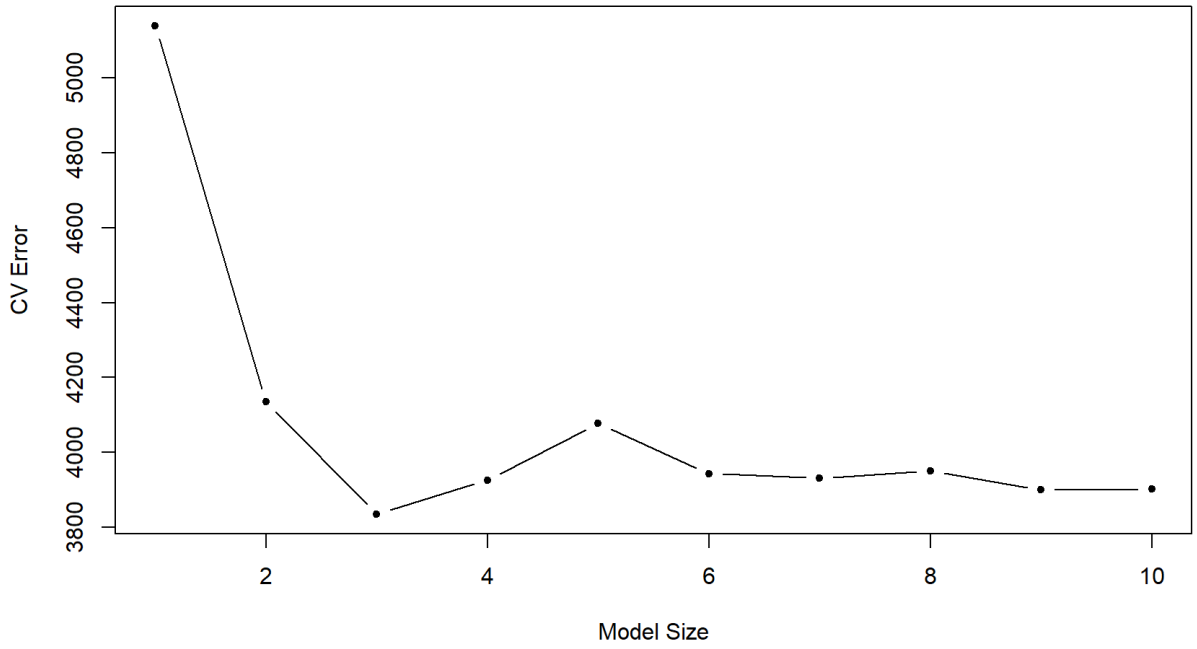
We start with a multiple regression between fantasy points and each predictor variable. To no surprise, we see the p-value is small and we can reject the null hypothesis and conclude that at least one of these predictors is useful in predicting a QB’s fantasy points. Now, let’s see what a potential best fit model might be, using BIC as our benchmark.



From the BIC chart, we expect somewhere between three and six variables may the best model. Now, we'll perform a stepwise regression to gain more information about how many and which predictor variables would be best to use.

```
lm(formula = FP ~ CAY + IAY + AGG + LCAD + AYTS + COMP + CPAE,
    data = ngs)

Coefficients:
(Intercept)          CAY          IAY          AGG         LCAD         AYTS
COMP         CPAE
   -679.056       13.881      -24.792       -2.911        3.723       38.902
13.299       -4.683
```

A stepwise selection process suggest that seven predictor variables may be the best model fit, choosing CAY, IAY, AGG, LCAD, AYTS, CP, and CPAE as the predictor variables, a bit higher than the suggest four variables from the BIC approach. Next, we'll run a cross validation to gain more information.



Based off of the CV error, somewhere between three and six variables may be best, though five seems to be a tad high. With all these metrics in mind, perhaps three or four variables will be appropriate. Now, for the sake of producing a regression, we'll pick the three 'best' predictor variables as the difference between three and four in BIC isn't too large, but in CV error it seems like there's a notable difference between three and four with three having a lower CV error.

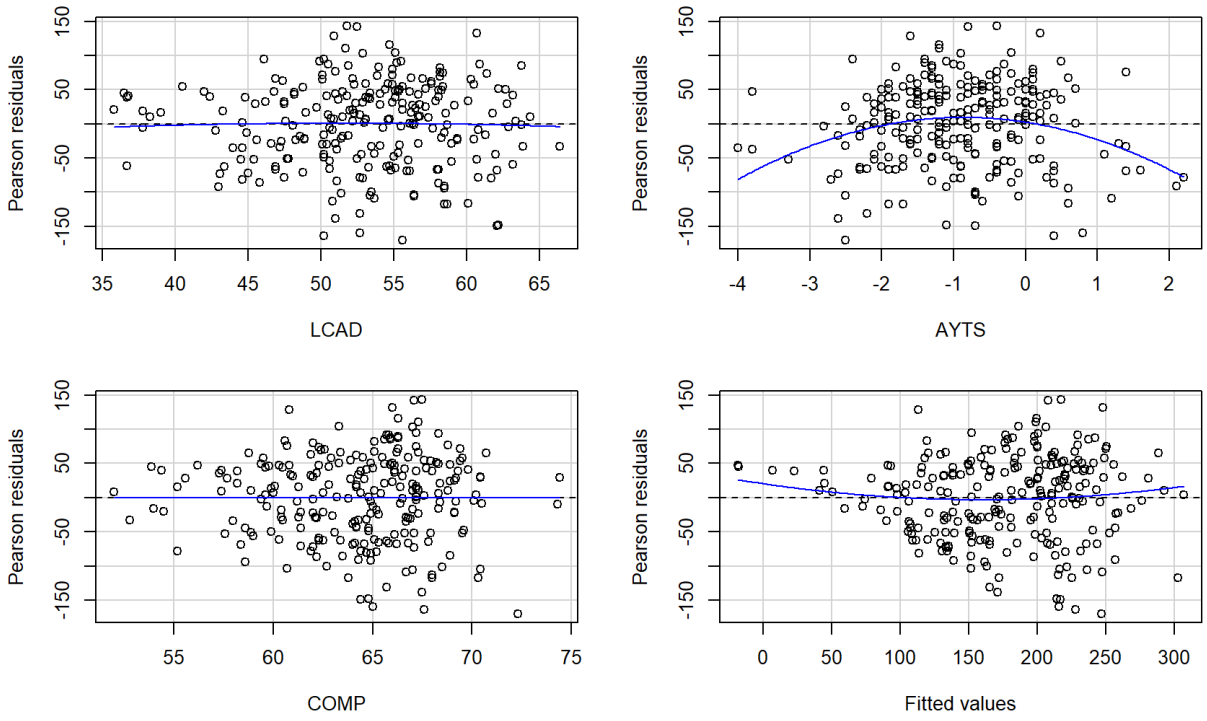
```
lm(formula = FP ~ LCAD + AYTS + COMP, data = ngs)

Residuals:
     Min       1Q   Median       3Q      Max
-170.222  -44.633    7.011   45.726  142.970

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -757.3464    71.1145 -10.650  < 2e-16 ***
LCAD           3.9841     0.7473   5.331 2.26e-07 ***
AYTS          19.6597     4.4981   4.371 1.85e-05 ***
COMP          11.5038     1.0847  10.605  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 62.4 on 238 degrees of freedom
Multiple R-squared:  0.4599,   Adjusted R-squared:  0.4531
F-statistic: 67.56 on 3 and 238 DF,  p-value: < 2.2e-16
```

Our regression model includes LCAD, AYTS, and CP as our predictor variables. Each predictor variable has a low p-value and under the general 0.5 significant level, suggesting this model and each variable are useful in predicting a QB's fantasy point total at the end of a season. Our adjusted R-squared is moderate at 0.45, but not as strong as we'd like. However, after using BIC and CV error, it isn't a deterrent for this model and we'll proceed to performing diagnostics.

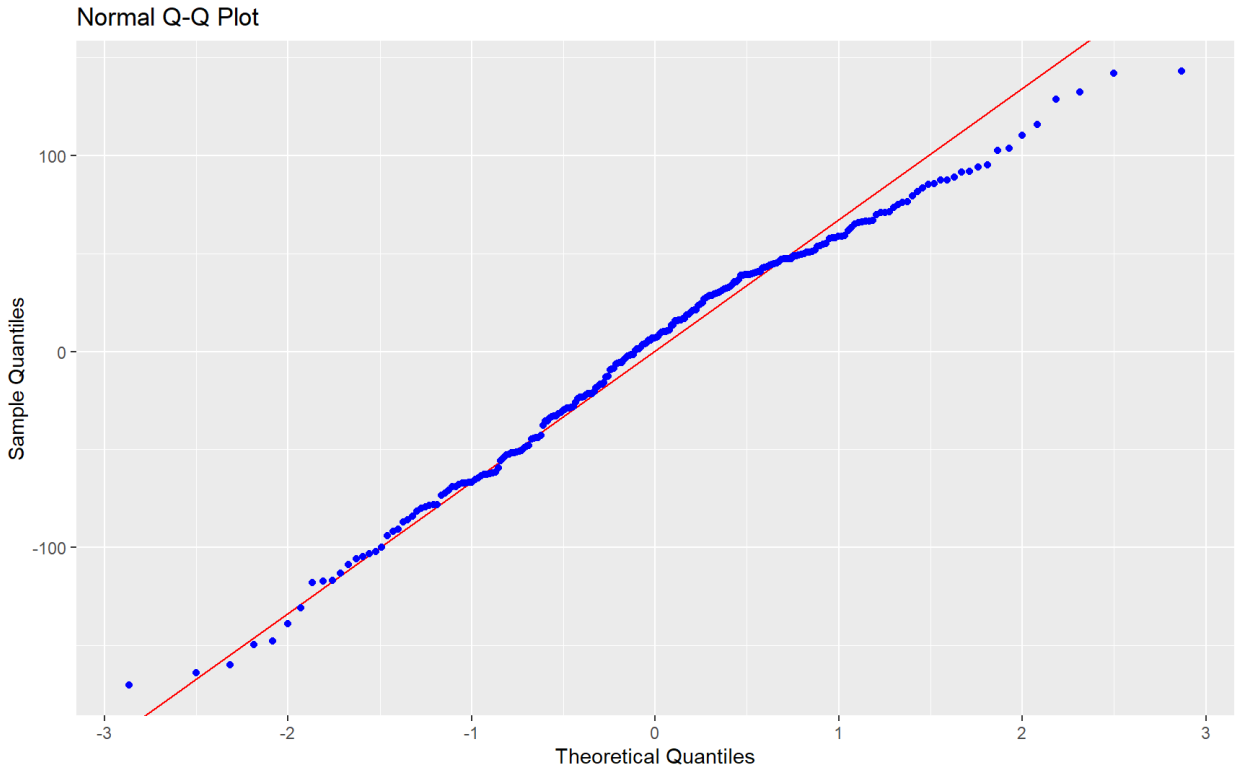
We'll start with checking to see if we have any collinearity by calculating the VIF for each variable.

```
    LCAD     AYTS     COMP
1.239548 1.273413 1.119712
```

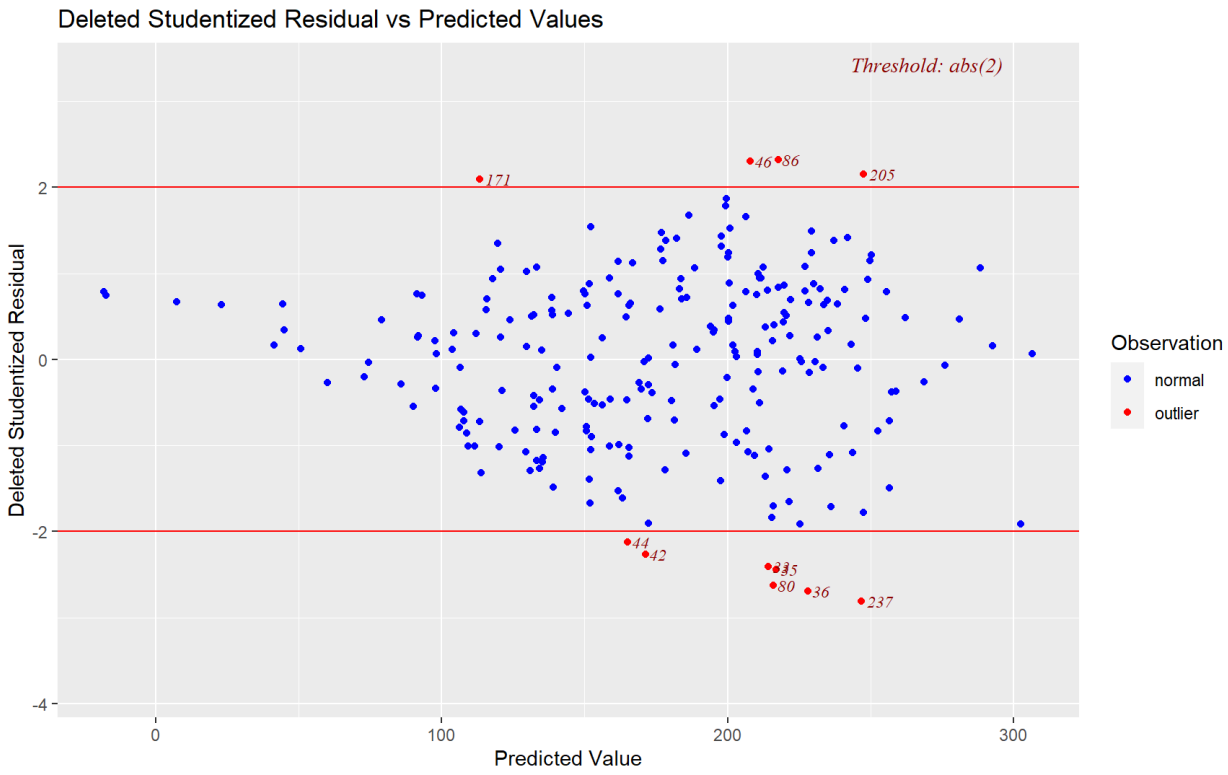
We generally want each VIF to be below 10 and each VIF for our predictors is less than two, so we can proceed to our next step of residual plots.



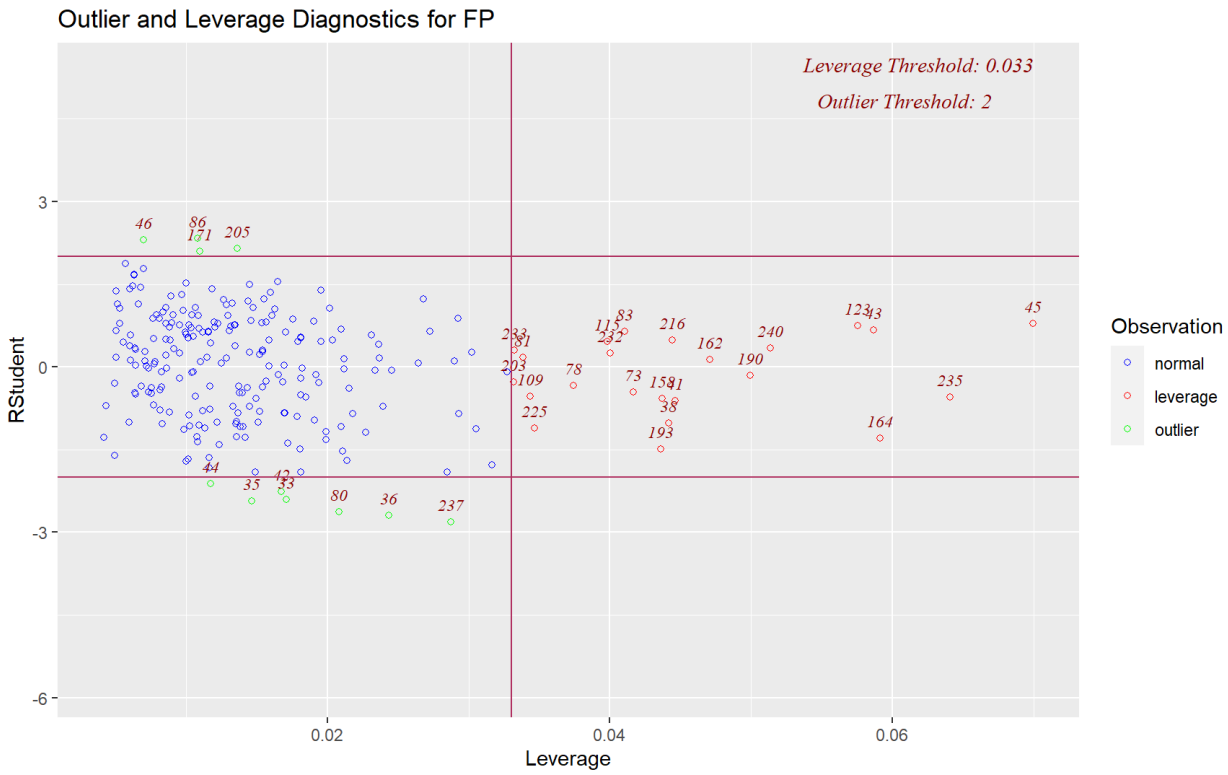
We see no discernable patterns in each predictor variable plot. For the fitted values, we generally want a square size, but there doesn't appear to be any problematic patterns so we'll continue with our diagnostics by looking at a Q-Q plot of residuals.



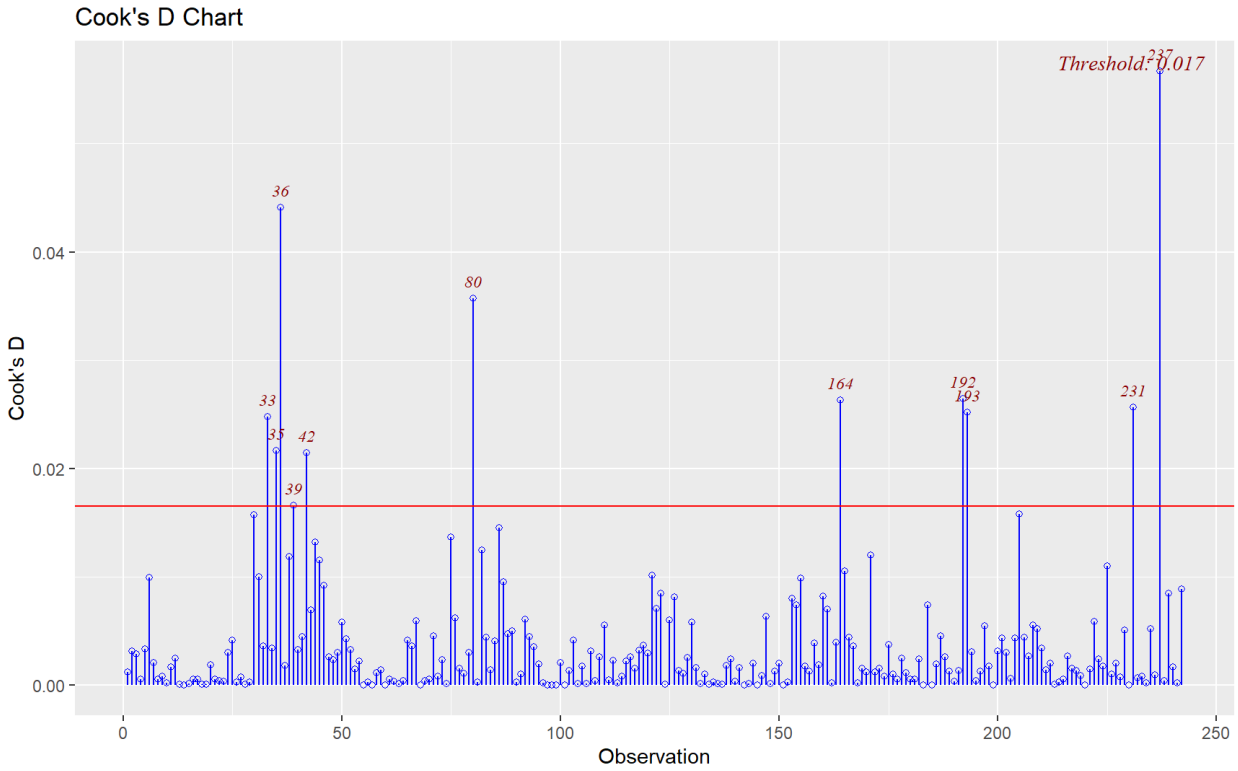
We want our residuals to follow the diagonal line as close as possible. The residuals seem to follow very closely but as we get further down the quantiles, we do notice a bit of an aberration. However, we're rarely dealing with perfect data and these residuals don't appear to be highly influential, so we'll now check for outliers.



There are 11 outliers, but none are higher than 3, plus or minus, values, so we'll keep data points in our data because they aren't exceptionally large. Now, we'll look for potential leverage points.



We're mostly concerned with the top right of this plot and there are no problematic leverage points to be found. Finally, we'll look at a cooks d plot.



The only points of any real concern are 36 and 237, but since they are both below 0.06, they aren't much of an issue and we'll continue with these data points in our set.

Interpretations and Conclusions

Based off of this study, we conclude that, in order to best predict how well a QB will perform in fantasy football based off their passing tracking data, the best predictors are LCAD, AYTS, and CP. A QB's ability to push the ball downfield and to the first down markers while prioritizing making efficient throws and decisions is conducive to their success in scoring fantasy points.

Unfortunately, this study was cut short so there is still plenty of room for improvement. With tracking data being so potentially powerful, we may consider using a regularization procedure like ridge regression to include each variable or we can use lasso to get subset variables. Also, football is incredibly complex and nuanced sport that is

incredibly difficult to explain through data. There are many other factors that could be in play that could greatly influence our study. For QB's, a strong supporting cast and offensive genius coaches can greatly aid a lesser QB to unseen heights, or a poor collection of talent and terrible coaching or management can bring down even the most talented players to levels that weren't fathomable in college. Hopefully, as techniques become more advanced, the NFL can become a more heavily data-driven league and maybe there will be more publicly released data that can greatly enhance our understanding of player's performance. For now, tracking data still has endless possibilities and there's so much more to explore.

Appendix

```
# import data

#install.packages("readxl")

library(readxl)

setwd("~/Clemson University/Summer 2024/Final")

ngs <- read_excel("~/Clemson University/Summer 2024/Final/NGS.xlsx")



#create new variable for response variable of interest

ngs$FP <-ngs$TD*4 + ngs$YDS*.04 - ngs$INT*2


# remove irrelavent data columns and rename variables

library(dplyr)

ngs <- ngs %>% select(-`PLAYER NAME`,- TEAM, -ATT, -YDS, -TD, -INT, - RATE)


names(ngs)[names(ngs) == "AGG%"] <- "AGG"

names(ngs)[names(ngs) == "COMP%"] <- "COMP"

names(ngs)[names(ngs) == "xCOMP%"] <- "xCOMP"


# Reshape the data to long format for easier plotting

library(tidyr)


#correlation matrix
```

```
cor(ngs)
#install.packages("corrplot")
library(corrplot)

cor.ngs <-cor(ngs)

#install.packages("RColorBrewer")
library(RColorBrewer)

#create colored correlation matrix
clemson_palette <- colorRampPalette(c("orange", "purple", "white"))(200)

png("correlation.matrix.png", width = 800, height = 600)
corrplot(cor.ngs, method = "circle", col = clemson_palette, addCoef.col = NA)
dev.off()


# Plot fantasy points vs. time to throw
library(ggplot2)
ggplot(ngs, aes(x = TT, y = FP)) +
  geom_point() +
  geom_smooth(method = "lm", color = "orange") +
  ggtitle("Time to Throw vs. Fantasy Points") +
```

```
  xlab("Time to Throw") +
  ylab("Fantasy Points")

# Plot fantasy points vs. Average Completed Air Yards
ggplot(ngs, aes(x = CAY, y = FP)) +
  geom_point() +
  geom_smooth(method = "lm", color = "orange") +
  ggtitle("Average Completed Air Yards vs. Fantasy Points") +
  xlab("Average Completed Air Yards") +
  ylab("Fantasy Points")

# Plot fantasy points vs. Average Intended Air Yards
ggplot(ngs, aes(x = IAY, y = FP)) +
  geom_point() +
  geom_smooth(method = "lm", color = "orange") +
  ggtitle("Average Average Intended Air Yards vs. Fantasy Points") +
  xlab("Average Intended Air Yards") +
  ylab("Fantasy Points")

# Plot fantasy points vs. Aggressiveness
ggplot(ngs, aes(x = AGG, y = FP)) +
  geom_point() +
  geom_smooth(method = "lm", color = "orange") +
```

```
  ggtitle("Aggressiveness  vs. Fantasy Points") +
  xlab("Aggressiveness ") +
  ylab("Fantasy Points")

# Plot fantasy points vs. Average Air Yards Differential (AYD)
ggplot(ngs, aes(x = AYD, y = FP)) +
  geom_point() +
  geom_smooth(method = "lm", color = "orange") +
  ggtitle("Average Air Yards Differential (AYD)  vs. Fantasy Points") +
  xlab("Average Air Yards Differential (AYD) ") +
  ylab("Fantasy Points")


# Plot fantasy points vs. Longest Completed Air Distance
ggplot(ngs, aes(x = LCAD, y = FP)) +
  geom_point() +
  geom_smooth(method = "lm", color = "orange") +
  ggtitle("Longest Completed Air Distance  vs. Fantasy Points") +
  xlab("Longest Completed Air Distance ") +
  ylab("Fantasy Points")

# Plot fantasy points vs. Air Yards to the Sticks
ggplot(ngs, aes(x = AYTS, y = FP)) +
```

```
  geom_point() +
  geom_smooth(method = "lm", color = "orange") +
  ggtitle("Air Yards to the Sticks   vs. Fantasy Points") +
  xlab("Air Yards to the Sticks ") +
  ylab("Fantasy Points")

# Plot fantasy points vs. Completion Probability
ggplot(ngs, aes(x = COMP, y = FP)) +
  geom_point() +
  geom_smooth(method = "lm", color = "orange") +
  ggtitle("Completion Probability   vs. Fantasy Points") +
  xlab("Completion Probability") +
  ylab("Fantasy Points")

# Plot fantasy points vs. Expected Completion Percentage
ggplot(ngs, aes(x = xCOMP, y = FP)) +
  geom_point() +
  geom_smooth(method = "lm", color = "orange") +
  ggtitle("Expected Completion Percentage vs. Fantasy Points") +
  xlab("Expected Completion Percentage") +
  ylab("Fantasy Points")

# Plot fantasy points vs. Completion Percentage Above Expectation
```

```
ggplot(ngs, aes(x = CPAE, y = FP)) +
  geom_point() +
  geom_smooth(method = "lm", color = "orange") +
  ggtitle("Completion Percentage Above Expectation vs. Fantasy Points") +
  xlab("Completion Percentage Above Expectation") +
  ylab("Fantasy Points")



#full model
full_model <- lm(FP ~ ., data = ngs)
summary(full_model)

#best model
library(leaps)
best.mods <- regsubsets(FP ~ ., data = ngs)

#plot BIC
plot(summary(best.mods)$bic, pch= 20, xlab= "No. variables",
    ylab= "BIC")
lines(summary(best.mods)$bic, type= "c", lwd= 2)

#stepwise regression
library(MASS)
```

```
null<- lm(FP ~ 1, data = ngs)
full<- lm(FP ~ ., data = ngs)
best <- stepAIC(full, scope= list(lower= null, upper= full),
      direction= "both", trace= FALSE)

#loop 10 k folds for cv error
set.seed(314)
k <- 10
folds <- sample(1:k, nrow(ngs), replace= TRUE) # Sample from
# 1,2,...,10 n times with replacement
head(folds)
cv.errors <- matrix(nrow= k, ncol= 10)

# Loop over each of k holdout sets
for (h.out in 1:k){

 # Get the best fitting models for each size
 best.fit <- regsubsets(FP ~ ., data= ngs[folds != h.out, ],
                  nvmax= 10)

 mod.mat <- model.matrix(FP~., data= ngs[folds == h.out, ])

 # Loop through model size
```

```
  for (i in 1:10){

    coefi <- coef(best.fit, id= i)
    pred <- mod.mat[, names(coefi)]%*%coefi

    cv.errors[h.out, i] <- mean((ngs$FP[folds == h.out] - pred)^2)

  }  # End loop over model size


}  # End loop over sets

(mean.cv.errors <- apply(cv.errors, 2, mean))

#plot cv errors
plot(mean.cv.errors, type= "b", pch= 20, xlab= "Model Size",
     ylab= "CV Error")

#choose best three predictors variables
coef(best.fit, 3)


library(ggplot2)
```

```
library(gridExtra)

#final model
final.mod <-lm(FP ~ LCAD + AYTS + COMP , data = ngs)
summary(final.mod)

library(car)

#vif
vif(final.mod)

# residual plots
residualPlots(final.mod)

## boxplot

boxplot(final.mod$residuals, horizontal=TRUE)

library(olsrr)

## Q-Q Plot
ols_plot_resid_qq(final.mod)
```

```
## outliers

ols_plot_resid_stud_fit(final.mod)$outliers


## resid-leverage plot

ols_plot_resid_lev(final.mod)


## Cooks D plot

ols_plot_cooksd_chart(final.mod)
```