Predicting Player Minutes based on Per 36 Rate Statistics

Group 4: Vignesh Mahalingam, Brandon Wang, Arjie Nanda

Introduction

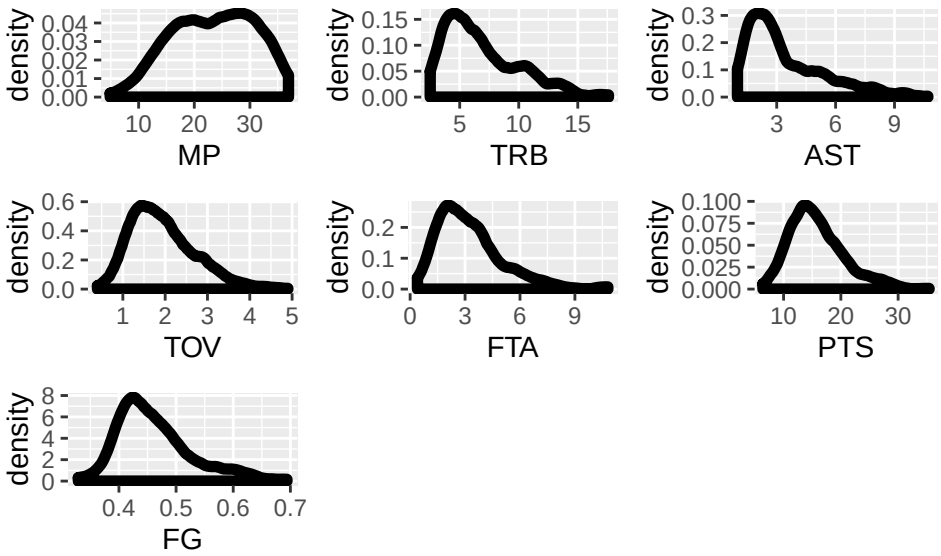
In this report, we plan creating models for predicting minutes played through the other types of basketball statistics. Minutes played is a valuable stat because it keeps track of how often a player is on the court. More minutes played gives players more time to contribute to the game. We will use a combination of both counting stats and rate stats. We hypothesize that the counting stats will be positively correlated with minutes played, while the rate stats should have no relationship. It is certainly possible that players who have higher rate stats, like three point percentage, are more valuable and will play more minutes as a result. On the other hand, players who play fewer minutes could have better rates due to the increased variability in a smaller sample. We are using the per 36 player data from the NBA's 2018-19 season, which can be found at https://www.basketball-reference.com/leagues/NBA_2019_per_minute.html . This data is taken from the NBA's own statistics page, and is a widely used industry source. Per 36 refers to adjusting the player's statistics to project what their stats would be if the player played 36 minutes per game. We will be using the per 36 data as it provides a good insight into a player's productivity without interference from the number of minutes played by the player. The individual population units are players with a minimum of 41 games played, which is half the season. We're trying to generalize this to future NBA seasons, as a method of estimating the average number of minutes played by a given player. We believe there are around 240 such players in the league. We also decided to restrict our predictor variables to points, rebounds, assists, turnovers, free throws attempted, and field goal percentage, as these are common stats used in player evaluation.

Exploratory Analysis

We first decided to do some exploratory data analysis on our data. For the initial univariate data exploration, we settled on using Kernel Density Estimations (KDEs) with an Epanechnikov kernel and Sheather-Jones bandwidth. We felt these KDEs would be most appropriate as they hold no assumptions about the underlying distribution of the data.

```
kde1 = ggplot(data = nbaper36select, aes(MP)) +
  geom_density(bw = "SJ", kernel = "epanechnikov", size = 2)


kde2 = ggplot(data = nbaper36select, aes(TRB)) +
  geom_density(bw = "SJ", kernel = "epanechnikov", size = 2)


kde3 = ggplot(data = nbaper36select, aes(AST)) +
  geom_density(bw = "SJ", kernel = "epanechnikov", size = 2)


kde4 = ggplot(data = nbaper36select, aes(TOV)) +
  geom_density(bw = "SJ", kernel = "epanechnikov", size = 2)
```

```
kde5 = ggplot(data = nbaper36select, aes(FTA)) +
  geom_density(bw = "SJ", kernel = "epanechnikov", size = 2)


kde6 = ggplot(data = nbaper36select, aes(PTS)) +
  geom_density(bw = "SJ", kernel = "epanechnikov", size = 2)


kde7 = ggplot(data = nbaper36select, aes(FG)) +
  geom_density(bw = "SJ", kernel = "epanechnikov", size = 2)

grid.arrange(kde1, kde2, kde3, kde4, kde5, kde6, kde7, ncol = 3)
```

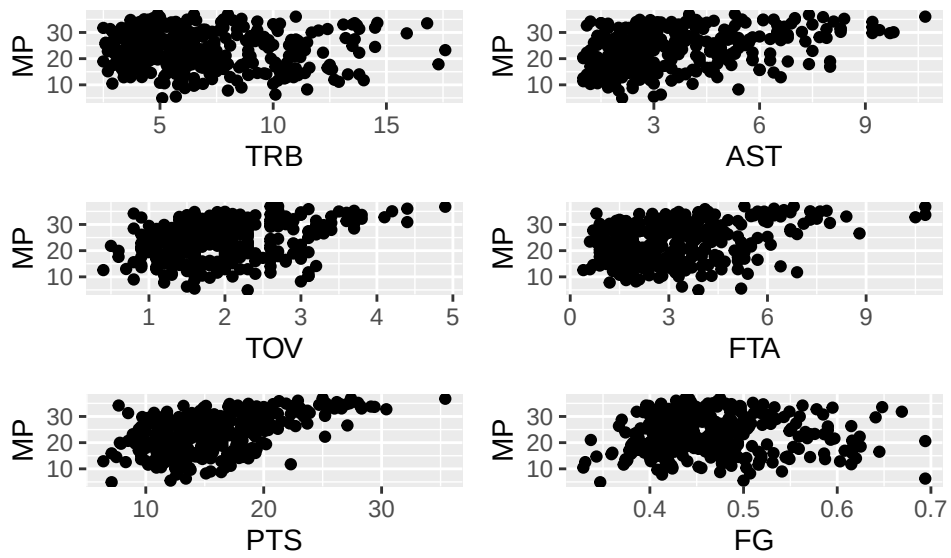


From the KDEs we can see a marked difference in the KDEs of the predictor variables and the response variables. The predictor variables have distributions with early peaks and long tails, illustrating that the bulk of NBA players have average stats, with only a few players having lower rate statistics and relatively more players having higher rate statistics as compared to the mean. The `MP` KDE shows a much different pattern, instead having a slightly bimodal distribution that corresponds with the minutes played for bench players and starters.

We also created a series of scatterplots comparing `MP` with each predictor variable.

```
splot1 <- ggplot(data = nbaper36select, aes(x = TRB, y = MP)) +
  geom_point()

splot2 <- ggplot(data = nbaper36select, aes(x = AST, y = MP)) +
  geom_point()
splot3 <- ggplot(data = nbaper36select, aes(x = TOV, y = MP)) +
  geom_point()
splot4 <- ggplot(data = nbaper36select, aes(x = FTA, y = MP)) +
  geom_point()
splot5 <- ggplot(data = nbaper36select, aes(x = PTS, y = MP)) +
  geom_point()
splot6 <- ggplot(data = nbaper36select, aes(x = FG, y = MP)) +
  geom_point()
```

```r
grid.arrange(splot1, splot2, splot3, splot4, splot5, splot6, ncol = 2)
```



From this we can see that some of the variables have a positive relationship with MP, while some have a relationship that is more variable.

OLM and Bootstrap

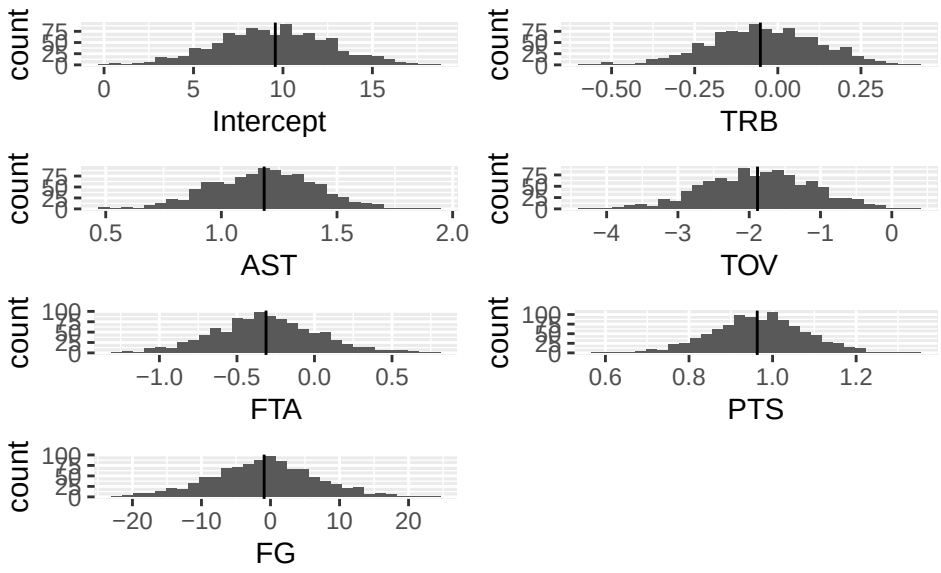
```r
#OLM
model_olm = lm(data = nbaper36, MP ~ TRB + BLK + TOV + FTA + PTS + FG)
summary(model_olm) # Only PTS is significant s
```

```
##
## Call:
## lm(formula = MP ~ TRB + BLK + TOV + FTA + PTS + FG, data = nbaper36)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.9754  -3.9438   0.3711   4.1442  17.4790
##
## Coefficients:
##             Estimate Std. Error t value         Pr(>|t|)
## (Intercept)  11.2326     1.5476   7.258 0.00000000000266 ***
## TRB          -0.2459     0.1610  -1.527          0.12762
## BLK          -0.4072     0.6590  -0.618          0.53704
## TOV           0.6613     0.5788   1.142          0.25408
## FTA          -0.3994     0.3859  -1.035          0.30133
## PTS           1.0317     0.3650   2.827          0.00498 **
## FG           -0.3579     0.8810  -0.406          0.68485
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.022 on 342 degrees of freedom
```

```
## Multiple R-squared:  0.3197, Adjusted R-squared:  0.3077
## F-statistic: 26.78 on 6 and 342 DF,  p-value: < 0.00000000000000022
```

We also ran a bootstrap on the kitchen sink OLM to see if the relationship between the predictor variables is or is not impacted by bootstrapping. We ran the bootstrap with 1000 repetitions, getting a histogram for the t statistic for each OLM coefficient.

```
set.seed(1109)


# function to obtain regression weights
bs <- function(formula, data, indices) {
  d <- data[indices,] # allows boot to select sample
  fit <- lm(formula, data=d)
  return(coef(fit))
}
# bootstrapping with 1000 replications
results <- boot(data = nbaper36select, statistic=bs,
   R=1000, formula = MP ~ TRB + AST + TOV + FTA + PTS + FG)

# view results
tvalues = as.data.frame(results[2])
tnames = colnames(nbaper36select)[c(-1, -2)]
colnames(tvalues) = c("Intercept", tnames)
origtvalues = as.numeric(unlist(results$t0))

tplot1 = ggplot(data = tvalues, aes(x = Intercept)) + geom_histogram() + geom_vline(xintercept = origtva
tplot2 = ggplot(data = tvalues, aes(x = TRB)) + geom_histogram() + geom_vline(xintercept = origtvalues[
tplot3 = ggplot(data = tvalues, aes(x = AST)) + geom_histogram() + geom_vline(xintercept = origtvalues[
tplot4 = ggplot(data = tvalues, aes(x = TOV)) + geom_histogram() + geom_vline(xintercept = origtvalues[
tplot5 = ggplot(data = tvalues, aes(x = FTA)) + geom_histogram() + geom_vline(xintercept = origtvalues[
tplot6 = ggplot(data = tvalues, aes(x = PTS)) + geom_histogram() + geom_vline(xintercept = origtvalues[
tplot7 = ggplot(data = tvalues, aes(x = FG)) + geom_histogram() + geom_vline(xintercept = origtvalues[7

grid.arrange(tplot1, tplot2, tplot3, tplot4, tplot5, tplot6, tplot7, ncol = 2)
```



From this, we can see that the t-statistic for `TRB`, `FG`, and `FTA` are not indicative of predictive value, white the t-statistic for `PTS`, `AST`, and `TOV` are.

Full OLS Multiple Regression Test

$H_0 : \beta_1 = \beta_2 = \beta_3 = \beta_4 = \beta_5 = \beta_6 = 0$

$H_A : \exists \beta_j \ni [\beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6] s.t. \beta_j \neq 0$

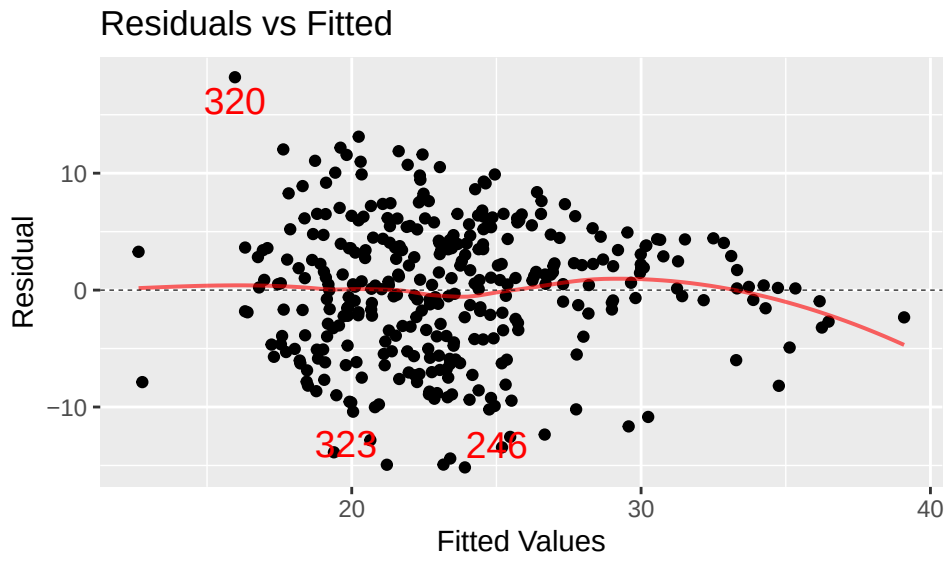
Where β_1 is the effect of points on minutes played, β_2 is the effect of turnovers on minutes played, β_3 is the effect of field goal percentage on minutes played, β_4 is the effect of total rebounds on minutes played, β_5 is the effect of assists on minutes played, and β_6 is the effect of free throw attemps on minutes played.

We will be using a significance level of $\alpha = 0.05$

```
#OLM
model_olm = lm(data = nbaper36select, MP ~ TRB + AST + TOV + FTA + PTS + FG)
summary(model_olm)
```

```
##
## Call:
## lm(formula = MP ~ TRB + AST + TOV + FTA + PTS + FG, data = nbaper36select)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.1651  -3.9609   0.4821   3.9609  18.2068
##
## Coefficients:
##             Estimate Std. Error t value             Pr(>|t|)
## (Intercept)   9.5726     2.7795   3.444             0.000644 ***
## TRB          -0.0525     0.1608  -0.326             0.744306
## AST           1.1850     0.2463   4.811          0.00000226 ***
## TOV          -1.8836     0.7723  -2.439             0.015241 *
## FTA          -0.3123     0.3195  -0.977             0.329059
## PTS           0.9633     0.1105   8.721 < 0.0000000000000002 ***
## FG           -0.9235     6.8308  -0.135             0.892530
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.831 on 342 degrees of freedom
## Multiple R-squared:  0.3621, Adjusted R-squared:  0.3509
## F-statistic: 32.36 on 6 and 342 DF,  p-value: < 0.00000000000000022
```

```
#Residual Plot
mplot(model_olm, which = c(1))
```



Conditions: As illustrated by the residual vs fitted plot, the residuals are approximately normally distributed as there is random scatter throughout the distribution. The plot all illuysrates the condition of equal variance being met as the variance is quite variable throughout the entirety of the data. We can also say that the data are independent of one another as one player's statistics are not influenced by that of another player.

The F- Statistic is 32.36 on 6 and 342 degrees of freedom, with an assocaited p-value less than 0.0001, so at the signficance level of $\alpha = 0.05$, the model is singifiance and we have significant evidence to reject the H_0 and claimed that $\exists \beta_j \ni [\beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6] s.t. \beta_j \neq 0$

Reduced OLS Multiple Regression Test

$H_0 : \beta_1 = \beta_2 = \beta_3 = 0$

$H_A : \exists \beta_j \ni [\beta_1, \beta_2, \beta_3] s.t. \beta_j \neq 0$

Where β_1 is the effect of points on minutes played, β_2 is the effect of turnovers on minutes played, and β_3 is the effect of assists on minutes played.

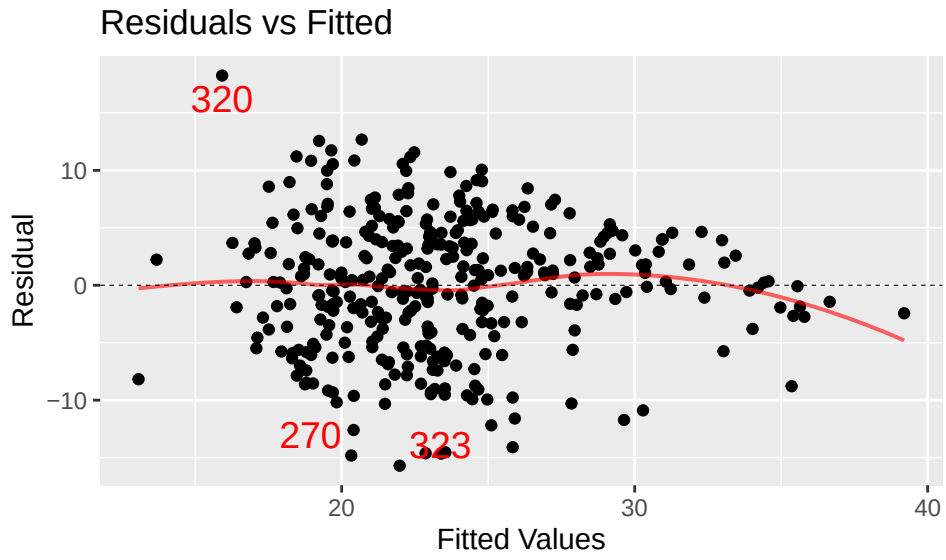
We will be using a significance level of $\alpha = 0.05$

```
ols_mod1 <- lm(MP ~   AST + TOV  + PTS , data = nbaper36select)
summary(ols_mod1)
```

```
##
## Call:
## lm(formula = MP ~ AST + TOV + PTS, data = nbaper36select)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.7055  -3.8498   0.4424   4.0048  18.2504
##
## Coefficients:
##             Estimate Std. Error t value             Pr(>|t|)
## (Intercept)  9.33953    1.12506   8.301  0.00000000000000236 ***
## AST          1.29371    0.21301   6.073  0.00000000330613938 ***
## TOV         -2.31980    0.66750  -3.475             0.000575 ***
```

```
## PTS          0.89405    0.08353  10.704 < 0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.822 on 345 degrees of freedom
## Multiple R-squared:  0.3586, Adjusted R-squared:  0.353
## F-statistic:  64.3 on 3 and 345 DF,  p-value: < 0.00000000000000022
```

```
mplot(ols_mod1, which = c(1))
```



Conditions: As illustrated by the residual vs fitted plot, the residuals are approximately normally distributed as there is random scatter throughout the distribution. The plot all illuysrates the condition of equal variance being met as the variance is quite variable throughout the entirety of the data. We can also say that the data are independent of one another as one player's statistics are not influenced by that of another player.

The F- Statistic is 64.3 on 3 and 345 degrees of freedom, with an assocaited p-value less than 0.0001, so at the signficance level of $\alpha = 0.05$, the model is singifiance and we have significant evidence to reject the H_0 and claimed that $\exists \beta_j \ni [\beta_1, \beta_2, \beta_3] s.t. \beta_j \neq 0$.

Full JHM Model

$H_0 : \beta_1 = \beta_2 = \beta_3 = \beta_4 = \beta_5 = \beta_6 = 0$

$H_A : \exists \beta_j \ni [\beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6] s.t. \beta_j \neq 0$

Where β_1 is the effect of points on minutes played, β_2 is the effect of turnovers on minutes played, β_3 is the effect of field goal percentage on minutes played, β_4 is the effect of total rebounds on minutes played, β_5 is the effect of assists on minutes played, and β_6 is the effect of free throw attemps on minutes played.

We will be using a significance level of $\alpha = 0.05$

```
model_f = rfit(data = nbaper36select, MP ~ TRB + AST + TOV + FTA + PTS + FG)
summary(model_f)
```

```
## Call:
## rfit.default(formula = MP ~ TRB + AST + TOV + FTA + PTS + FG,
```

```
##     data = nbaper36select)
##
## Coefficients:
##             Estimate Std. Error t.value               p.value
## (Intercept)  9.29426    2.88793  3.2183              0.001413 **
## TRB         -0.08985    0.16647 -0.5397              0.589737
## AST          1.17618    0.25498  4.6128 0.0000056244537609492 ***
## TOV         -1.76449    0.79942 -2.2072              0.027963 *
## FTA         -0.36538    0.33075 -1.1047              0.270064
## PTS          0.98719    0.11433  8.6344 0.0000000000000002277 ***
## FG           0.34104    7.07063  0.0482              0.961558
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.3429951
## Reduction in Dispersion Test: 29.75734 p-value: 0
```

Conditions: There are no conditions for the nonparametric JHM hypothesis test, aside from the fact that the data are independent. We can say this condition is met as one player's statistics are not influenced by others.

The Wald Statistic has a value of 188.9863 with an associated p-value of esentially 0, so we have significant evidence to reject the H_0 and say that $\exists \beta_j \ni [\beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6] s.t. \beta_j \neq 0$.

Reduced JHM Model

$H_0 : \beta_1 = \beta_2 = \beta_3 = 0$

$H_A : \exists \beta_j \ni [\beta_1, \beta_2, \beta_3] s.t. \beta_j \neq 0$

Where β_1 is the effect of points on minutes played, β_2 is the effect of turnovers on minutes played, and β_3 is the effect of assists on minutes played.

We will be using a significance level of $\alpha = 0.05$

```
model_f1 = rfit(data = nbaper36select, MP ~ AST + TOV + PTS)
summary(model_f1)
```

```
## Call:
## rfit.default(formula = MP ~ AST + TOV + PTS, data = nbaper36select)
##
## Coefficients:
##             Estimate Std. Error t.value               p.value
## (Intercept)  9.42386    1.18856  7.9288   0.00000000000003102 ***
## AST          1.29958    0.21993  5.9091   0.00000000825647954 ***
## TOV         -2.27201    0.68918 -3.2967               0.00108 **
## PTS          0.90833    0.08624 10.5325 < 0.00000000000000022 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.3388507
## Reduction in Dispersion Test: 58.93953 p-value: 0
```

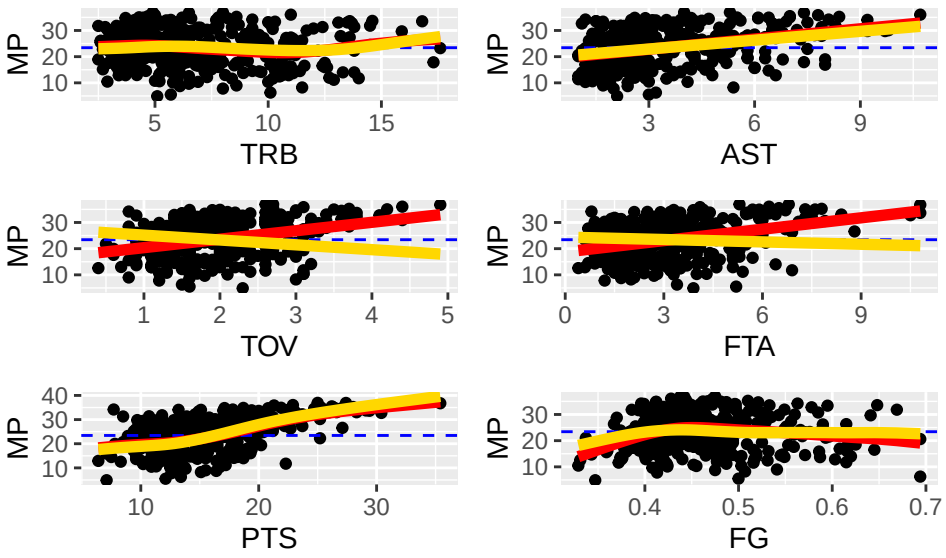
Conditions: There are no conditions for the nonparametric JHM hypothesis test, aside from the fact that the data are independent. We can say this condition is met as one player's statistics are not influenced by others.

The Wald Statistic has a value of 188.395 with an associated p-value of esentially 0, so we have significant evidence to reject the H_0 and say that $\exists \beta_j \ni [\beta_1, \beta_2, \beta_3] s.t. \beta_j \neq 0$.

GAM

Once we fit these models, we fit a generalized additive model (GAM) on the data. Initially, we created a smoothing spline and simple linear regression (SLR) for each of our six predictors. For each predictor, we compared the AIC of the smoothing spline vs the SLR and chose the method that produced a better fit. Once we had the best fit for all of our predictors, we built a GAM using those fits. The full GAM with all predictors is plotted below.

```
#graphing gam
grid.arrange(plot1, plot2, plot3, plot4, plot5, plot6, ncol = 2)
```

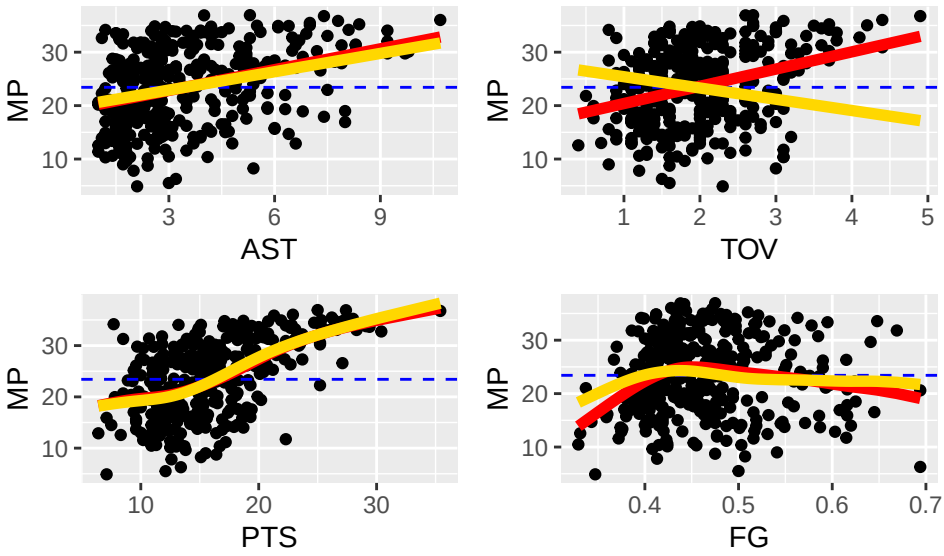



```
AIC(full_gam)
```

```
## [1] 2221.533
```

The first GAM returned an AIC of 2221.5334353. From the plot of the GAM, we noticed that the slopes for `TRB` and `FTA` were very flat, suggesting that these coefficients are close to zero. We attempted to fit a reduced GAM dropping these predictors from the model. The hope was that by dropping predictors and fitting a reduced model we would improve our model.

```
grid.arrange(plot7, plot8, plot9, plot10, ncol=2)
```



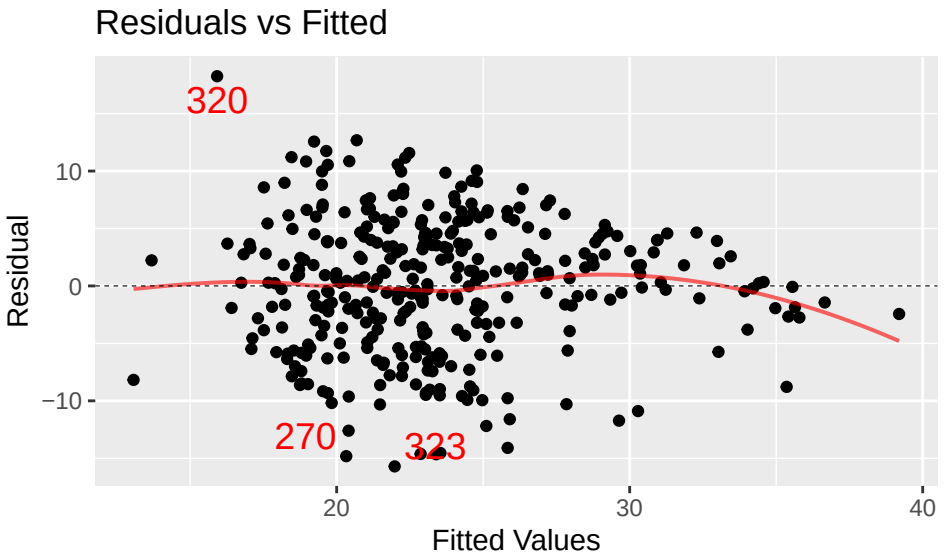

```
AIC(full_gam1)
```

```
## [1] 2219
```

The reduced GAM is plotted above. The coefficient for `FG` still looks a little flat, but attempts to drop this predictor resulted in a worse model. Using AIC to compare the two models, we find that the reduced GAM performs better than the full GAM. The AIC for the reduced GAM is 2219.0000079, which is slightly lower than the AIC for the full GAM we found above.

Kolmogorov Smirnov Test

```
# OLS KS Test Residual Plot
mplot(ols_mod1, which = c(1))
```



```r
#OLS KS Code
ks.test(x = resid(ols_mod1), y = pnorm, mean = 0, sd = 5.796699)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  resid(ols_mod1)
## D = 0.037301, p-value = 0.7165
## alternative hypothesis: two-sided
```

H_0 : F(t) = G(t) for all t.

H_A : F(t) $\neq$ G(t) for at least one t.

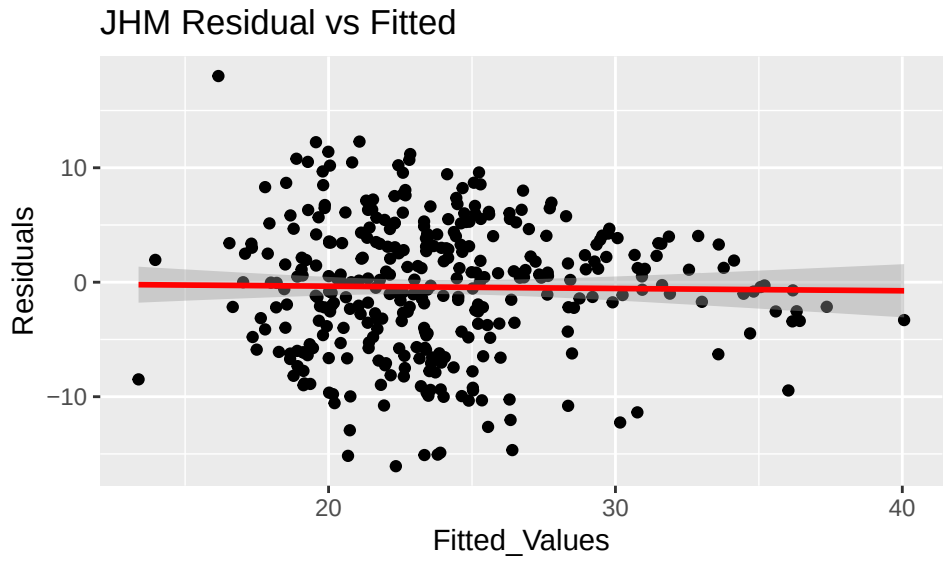
where F is the normal CDF of mean zero and standard deviation 5.7967 and G is the distribution of the residuals of the reduced ols.

We will be using a significance level of $\alpha = 0.05$

Conditions: There is independence in the distributions and the data comes from a continuous population. For both the normal and residual distribution, we can say they are continuous. We all can assume independence.

The test-statistic, D-value, generated by the Kolmogorov-Smirnov test is 0.049634 with an associated p-value of 0.3562. At the significance level of $\alpha = 0.05$, this indicates we do not have significant evidence to reject the H_0 and claim the residuals follow a roughly normal distribution.

```r
jhmdataframe = data.frame(model_f1$residuals, model_f1$fitted.values)
colnames(jhmdataframe) = c("Residuals", "Fitted_Values")
jhmresid = ggplot(data = jhmdataframe, aes(x = Fitted_Values, y = Residuals)) +
                    geom_point() + geom_smooth(method = lm, color = "red") + labs(title = "JHM Residual

plot(jhmresid)
```



H_0 : F(t) = G(t) for all t.

H_A : F(t) $\neq$ G(t) for at least one t.

where F is the normal CDF of mean zero and standard deviation 5.7975 and G is the distribution of the residuals of the reduced JHM model

We will be using a significance level of $\alpha = 0.05$

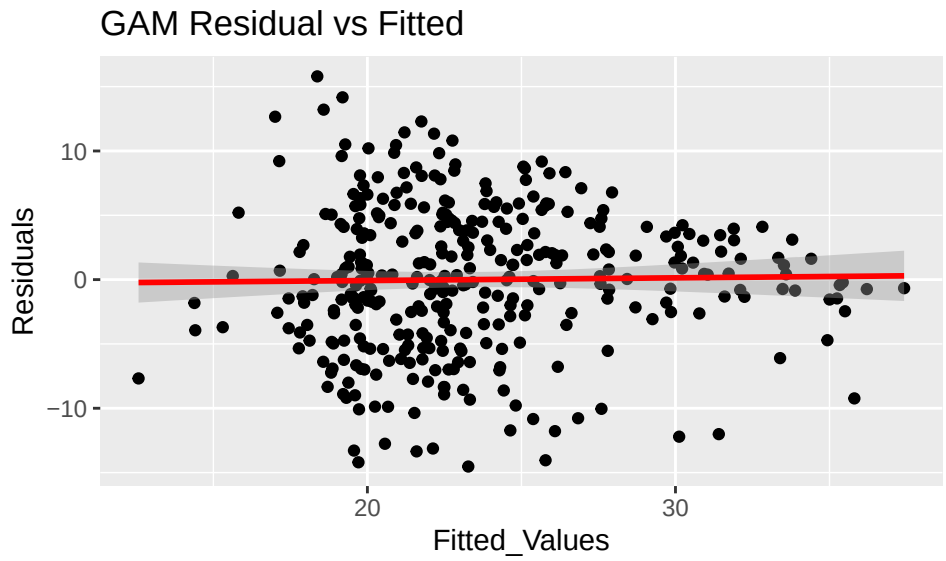
Conditions: There is independence in the distributions and the data comes from a continuous population. For both the normal and residual distribution, we can say they are continuous. We all can assume independence.

The test-statistic, D-value, generated by the Kolmogorov-Smirnov test is 0.046003 with an associated p-value of 0.4511 At the significance level of $\alpha = 0.05$, this indicates we do not have significant evidence to reject the H_0 and claim the residuals follow a roughly normal distribution.

```
#JHM KS Test
ks.test(x = resid(model_f), y = pnorm, mean = 0, sd = 5.797549)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  resid(model_f)
## D = 0.045965, p-value = 0.4522
## alternative hypothesis: two-sided
```

```
gamdataframe = data.frame(full_gam1$residuals, full_gam1$fitted.values)
colnames(gamdataframe) = c("Residuals", "Fitted_Values")
gamresid = ggplot(data = gamdataframe, aes(x = Fitted_Values, y = Residuals)) +
                  geom_point() + geom_smooth(method = lm, color = "red") + labs(title = "GAM Residual
plot(gamresid)
```



H_0 : F(t) = G(t) for all t.

H_A : F(t) $\neq$ G(t) for at least one t.

where F is the normal CDF of mean zero and standard deviation 5.6249 and G is the distribution of the residuals of the reduced GAM model

We will be using a significance level of $\alpha = 0.05$

Conditions: There is independence in the distributions and the data comes from a continuous population. For both the normal and residual distribution, we can say they are continuous. We all can assume independence.

The test-statistic, D-value, generated by the Kolmogorov-Smirnov test is 0.050011 with an associated p-value of 0.3472 At the significance level of $\alpha = 0.05$, this indicates we do not have significant evidence to reject the H_0 and claim the residuals follow a roughly normal distribution.

```
#Gam KS Test
ks.test(x = resid(full_gam), y = pnorm, mean = 0, sd = 5.624864)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  resid(full_gam)
## D = 0.050011, p-value = 0.3472
## alternative hypothesis: two-sided
```

Model Fit and CV

We also decided to see the fit statistics for the three models, as well as running a 5-fold crossvalidation on the fit statistics. Using that, we got the following fit statistics.

```
fit_final1 = rbind(fit_ols(ols_mod1),fit_jhm(model_f1),fit_gam(full_gam1))
rownames(fit_final1) =c("OLS", "JHM", "GAM")
kable(round(fit_final1, 4))%>% kable_styling(position = "center")
```

	rsq	adjrsq	propL1
OLS	0.3586	0.3530	0.2446
JHM	0.3550	0.3494	0.2473
GAM	0.3961	0.3782	0.2787

```
out10a =cv_rmc(dat = nbaper36select, ols_mod = ols_mod1, jhm_mod = model_f1, gam_mod = full_gam1)
kable(round(out10a,4))%>% kable_styling(position = "center")
```

	cv.rsq	cv.adjrsq	cv.propL1
OLS	0.3434	0.3377	0.2362
JHM	0.3402	0.3345	0.2375
GAM	0.3516	0.3324	0.2543

Overall, all the values indicate low predictive power. Interestingly, the GAM displays higher values in both R^2 and $L1_{prop}$ when compared to the OLS and JHM, yet displays a lower value for R^2_{Adj} wheen compared to the other two models. This seems to indicate the GAM is overfit. This issue along with the normality of the residuals as shown in the Kolmogorov-Smirnov test seem to indicate that the OLS model is best suited for prediction.

Conclusion

In conclusion, it seems that the models we created do an overall poor job of predicting the minutes played of a given NBA player. The different models also all had comparable fit statistics, indicating there wasn't much of a difference in performance between the three models. Future avenues of exploration could include creating position-specific models, using other statistics including defensive statistics or other advanced statistics, changing the filter conditions to include more or fewer players, or using multiple seasons worth of data.