# Exploratory Data Analysis

```
data <- read.csv("nba2024.csv", header = TRUE, check.names = FALSE)
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.4.2
```

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

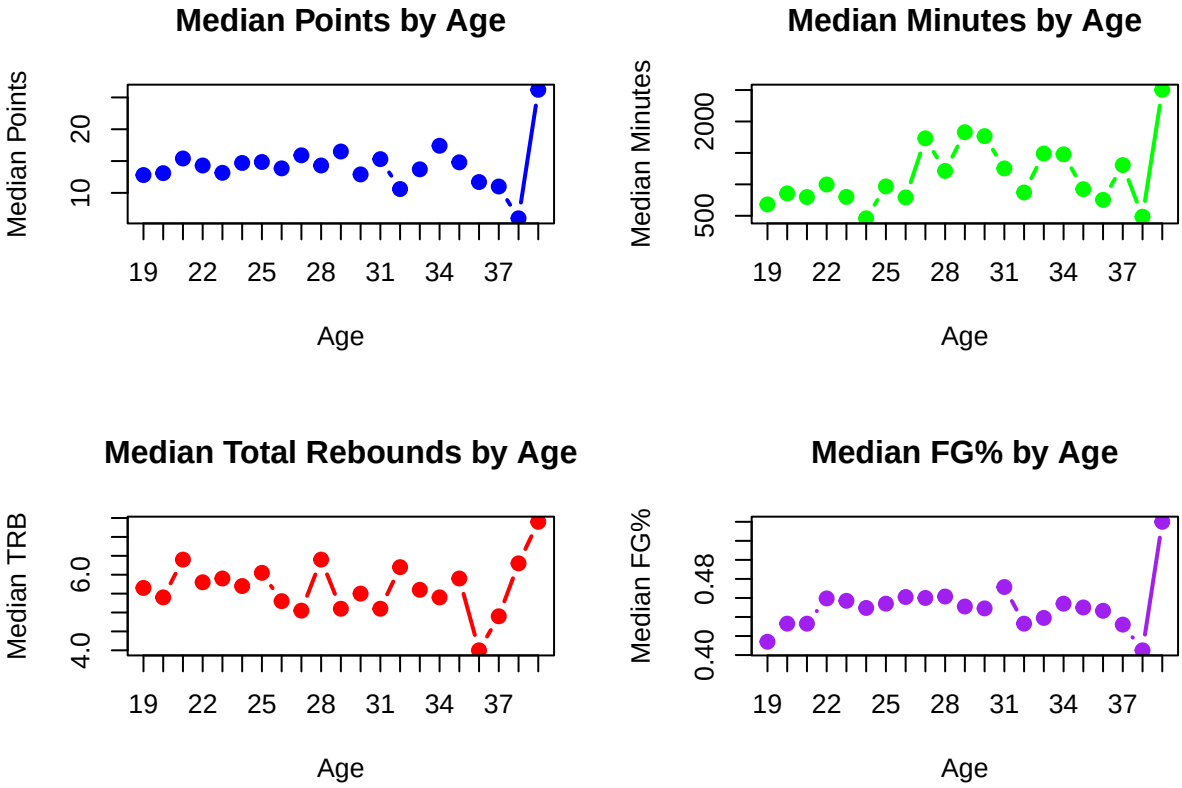
```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

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

```
data <- na.omit(data)

## convert data to numeric (might get errors otherwise)
data$PTS <- as.numeric(data$PTS)
data$MP <- as.numeric(data$MP)
data$TRB <- as.numeric(data$TRB)
data$`FG%` <- as.numeric(data$`FG%`)
data$Age <- as.numeric(data$Age)
```

```
## create age vs median statistic plots

# Calculate medians
age_medians <- data %>%
 group_by(Age) %>%
 summarise(
   med_pts = median(PTS, na.rm=TRUE),
   med_mp = median(MP, na.rm=TRUE),
   med_trb = median(TRB, na.rm=TRUE),
   med_fg = median(`FG%`, na.rm=TRUE)
 )

# Create line plots
par(mfrow=c(2,2))  # 2x2 grid of plots

# Points plot
plot(age_medians$Age, age_medians$med_pts, type="b",
    main="Median Points by Age",
    xlab="Age", ylab="Median Points",
```

```r
    col="blue", lwd=2, pch=19,
    xaxt="n")
axis(1, at=seq(min(age_medians$Age), max(age_medians$Age), by=1))

# Minutes played plot
plot(age_medians$Age, age_medians$med_mp, type="b",
    main="Median Minutes by Age",
    xlab="Age", ylab="Median Minutes",
    col="green", lwd=2, pch=19,
    xaxt="n")
axis(1, at=seq(min(age_medians$Age), max(age_medians$Age), by=1))

# Total rebounds plot
plot(age_medians$Age, age_medians$med_trb, type="b",
    main="Median Total Rebounds by Age",
    xlab="Age", ylab="Median TRB",
    col="red", lwd=2, pch=19,
    xaxt="n")
axis(1, at=seq(min(age_medians$Age), max(age_medians$Age), by=1))

# Field goal percentage plot
plot(age_medians$Age, age_medians$med_fg, type="b",
    main="Median FG% by Age",
    xlab="Age", ylab="Median FG%",
    col="purple", lwd=2, pch=19,
    xaxt="n")
axis(1, at=seq(min(age_medians$Age), max(age_medians$Age), by=1))
```




As we can see, there is a small general decline in most statistics as age increases. However, the decline doesn't seem to be as strong as one might think. We notice that median minutes is the most around the age of 30, which is likely due to player experience.

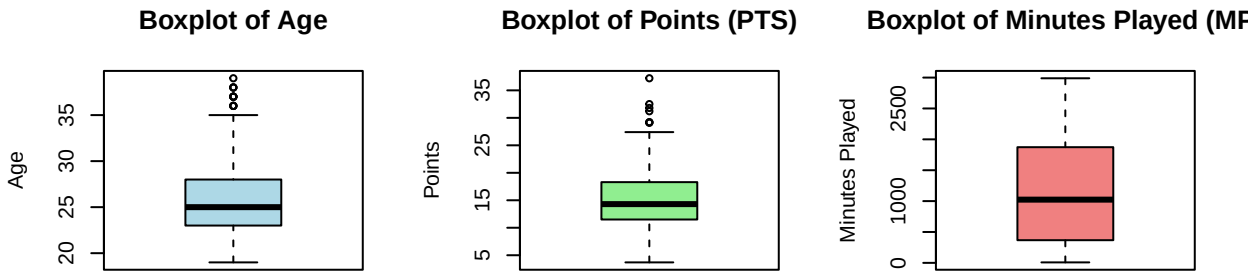
```
## Explain outliers, players 38 or above
older_players <- subset(data, Age >= 38)
print(older_players[, c("Player", "Age", "PTS", "MP", "TRB", "FG%")])
```

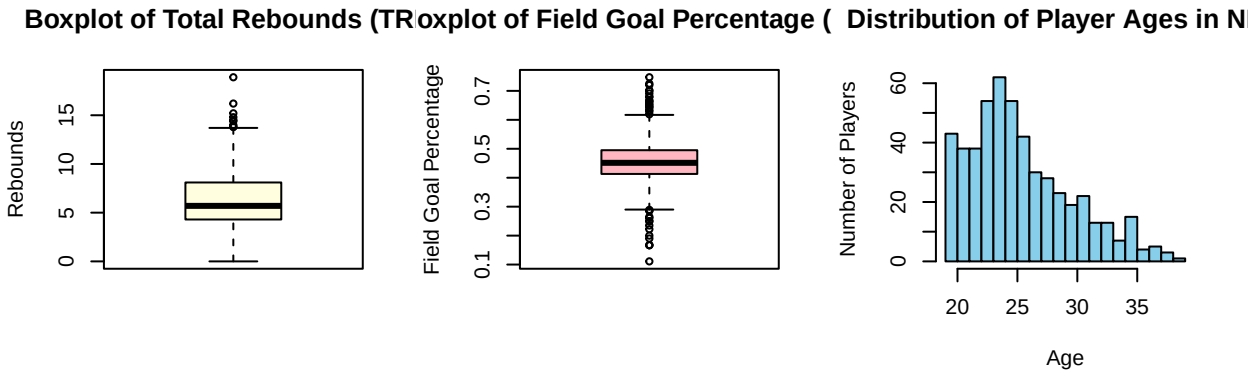
```
##           Player Age  PTS   MP TRB   FG%
## 169   Taj Gibson  38  6.0  204 6.5 0.405
## 250 LeBron James  39 26.2 2504 7.4 0.540
## 390   Chris Paul  38 12.5 1531 5.3 0.441
## 509  P.J. Tucker  38  3.9  486 6.3 0.360
```

It is also important to note outliers such as Lebron James, a star player with extremely high and unusual statistics for his age.

```
## Analyzing distribution of data

par(mfrow=c(2,3))  # 2x3 grid of plots
```

```r
# Boxplot for Age
boxplot(data$Age,
        main = "Boxplot of Age",
        ylab = "Age",
        col = "lightblue")

# Boxplot for Points (PTS)
boxplot(data$PTS,
        main = "Boxplot of Points (PTS)",
        ylab = "Points",
        col = "lightgreen")

# Boxplot for Minutes Played (MP)
boxplot(data$MP,
        main = "Boxplot of Minutes Played (MP)",
        ylab = "Minutes Played",
        col = "lightcoral")

# Boxplot for Total Rebounds (TRB)
boxplot(data$TRB,
        main = "Boxplot of Total Rebounds (TRB)",
        ylab = "Rebounds",
        col = "lightyellow")

# Boxplot for Field Goal Percentage (FG%)
boxplot(data$`FG%`,
        main = "Boxplot of Field Goal Percentage (FG%)",
        ylab = "Field Goal Percentage",
        col = "lightpink")

# Histogram of Age Distribution
hist(data$Age,
     main="Distribution of Player Ages in NBA",
     xlab="Age",
     ylab="Number of Players",
     col="skyblue",
     breaks=seq(min(data$Age), max(data$Age), by=1))  # breaks by 1 year
```




## Looking at the histogram, we can see that there are many more younger players than old as the data is skewed, so this is also important to consider when analyzing data.

```
## Five-number summaries:

# Five-number summary for Age
summary(data$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   19.00   23.00   25.00   25.79   28.00   39.00
```

```
# Five-number summary for Points (PTS)
summary(data$PTS)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    3.70   11.50   14.30   15.19   18.30   37.20
```

```
# Five-number summary for Minutes Played (MP)
summary(data$MP)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     8.0   367.8  1024.0  1132.7  1872.8  2989.0
```

```r
# Five-number summary for Total Rebounds (TRB)
summary(data$TRB)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   4.300   5.700   6.404   8.075  18.900
```

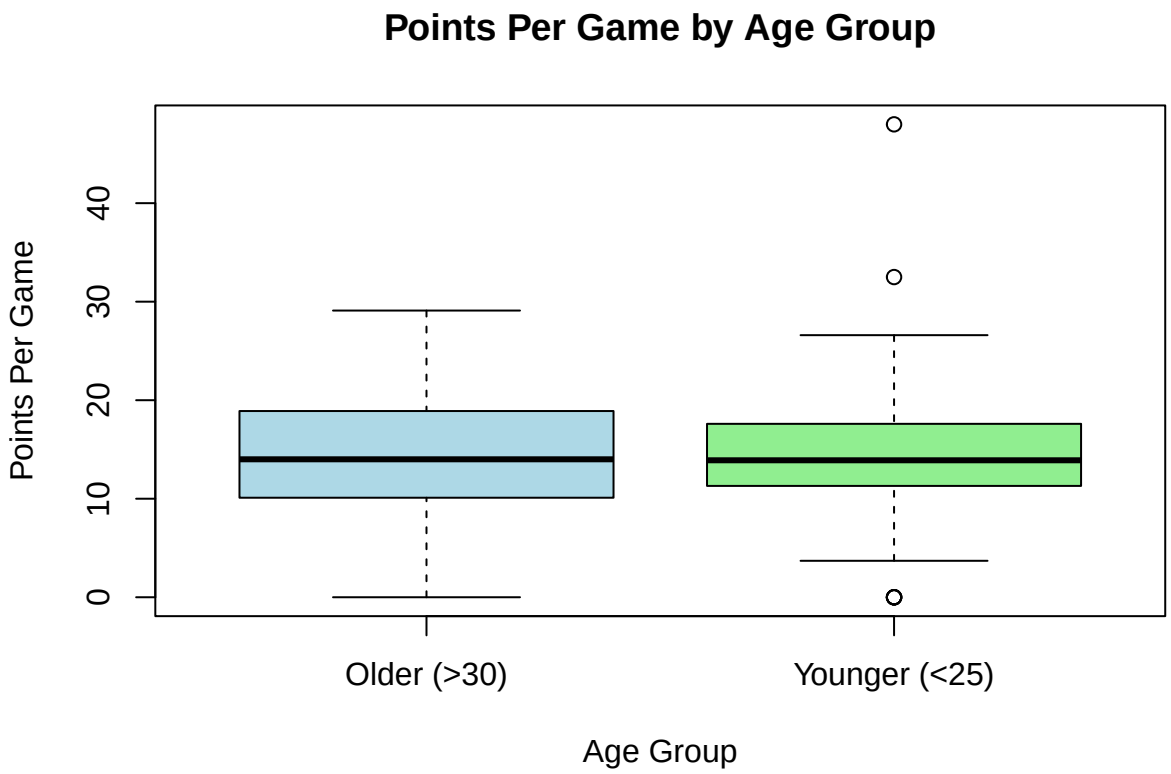
```r
# Five-number summary for Field Goal Percentage (FG%)
summary(data$`FG%`)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1110  0.4133  0.4515  0.4558  0.4945  0.7470
```

# Analysis #1

```r
#Analysis #1

# Load the CSV file containing data into a data frame
data <- read.csv("nba2024.csv", header = TRUE, check.names = FALSE)
```

```r
# Create a subset of players younger than 25 years
younger <- subset(data, Age < 25)

# Create a subset of players older than 30 years
older <- subset(data, Age > 30)

# Add a new column 'AgeGroup' to label younger players as "Younger (<25)"
younger$AgeGroup <- "Younger (<25)"

# Add a new column 'AgeGroup' to label older players as "Older (>30)"
older$AgeGroup <- "Older (>30)"

# Combine the younger and older player data frames into one for visualization purposes
combined_data <- rbind(younger, older)

# Visualization 1: Create a boxplot to compare Points Per Game (PTS) between the two age groups
boxplot(PTS ~ AgeGroup, data = combined_data,
        main = "Points Per Game by Age Group", # Title of the boxplot
        xlab = "Age Group",                    # Label for the x-axis
        ylab = "Points Per Game",              # Label for the y-axis
        col = c("lightblue", "lightgreen"))    # Colors for the boxplot categories
```




**Load data:**

```
# Perform a two-sample t-test to compare the mean Points Per Game (PTS) between younger and older playe
t_test_points <- t.test(younger$PTS, older$PTS)

# Display the results of the t-test, including the p-value, confidence intervals, and test statistic
t_test_points
```

```
##
##  Welch Two Sample t-test
##
## data:  younger$PTS and older$PTS
## t = 0.048125, df = 142.13, p-value = 0.9617
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.373953  1.442520
## sample estimates:
## mean of x mean of y
##  14.56302  14.52874
```

**Summary of Results**

**Means:** Younger players (<25): Average points per game = 14.56. Older players (>30): Average points per game = 14.53. The difference between the two means is very small (0.03 points), which suggests that both groups perform very similarly on average based off just looking at it.

**Statistical Results:**

- t-Statistic (t = 0.048125): The t-statistic is very close to 0, which indicates that the difference between the two means is negligible compared to the variability in the data.
- Degrees of Freedom (df = 142.13): The degrees of freedom indicate the effective sample size, accounting for both groups and the variance estimation.
- p-Value (p = 0.9617): The p-value is much greater than the common threshold of 0.05.

**Interpretation:** There is no statistically significant difference in points per game between younger and older players.

**95% Confidence Interval:** -1.373953 to 1.442520

In this case, we are 95% confident that the true difference in means lies between -1.37 and 1.44. The interval suggests that younger players might score up to 1.44 more points per game, or up to 1.37 fewer points per game, compared to older players.

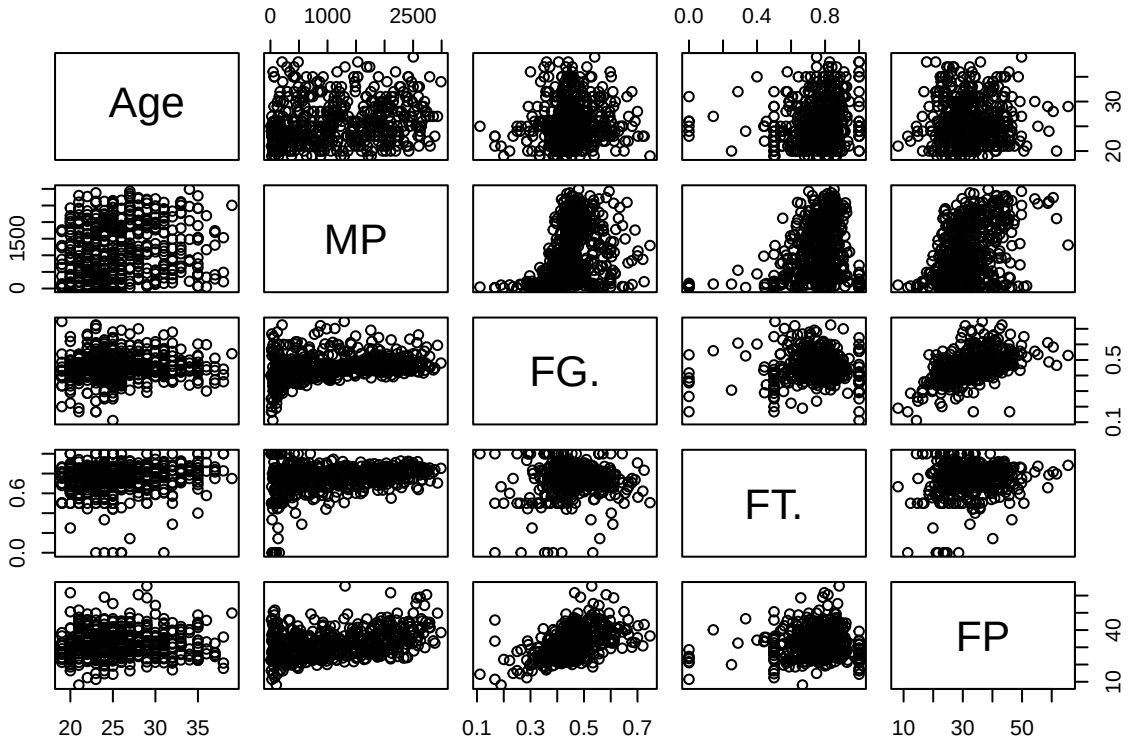
**Conclusion:** We conducted a two-sample t-test to compare the average points per game between younger players (<25 years old) and older players (>30 years old) to determine if younger players outperform older players in scoring. The results showed no statistically significant difference (p = 0.9617) between the two groups, as the 95% confidence interval for the difference in means (-1.37 to 1.44) included 0, suggesting that any observed difference in scoring is likely due to random variation rather than a true disparity. In the real world, this indicates that younger and older players contribute similarly in terms of points scored, and factors like experience, skill, or role on the team may play a more significant role in performance than age alone. It is also important to note that we didn't include the ages 25-30, which may have an average point per game that IS significantly different than both younger and older players.

# Analysis #2

```r
#load and clean data
data <- read.csv("nba2024.csv", header = TRUE)
data[data == ""] <- NA

data <- na.omit(data)
data <- data[, c("PTS", "ORB", "DRB", "AST", "STL", "BLK", "TOV", "Age", "MP", "FG.", "FT.")]
#calculte Fantasy Points
data$FP <- data$PTS +
           1.2 * (data$ORB + data$DRB) +
           1.5 * data$AST +
           3 * (data$STL + data$BLK) -
           data$TOV
```

The scatter plot matrix shows the relationships among the key variables (Age, Minutes Played, Field Goal Percentage, Free Throw Percentage, and Fantasy Points). The plot visualizes potential correlations and trends for our analysis. Looking at Age vs FP, it appears to have a weak realtionship around 0.

```r
#scatter plots of the variables and FP
plot(data[, 8:12])
```




Creates a linear regression model predicts Fantasy Points using Age, Minutes Played, FG%, and FT% as independent variables. The summary provides information on the coefficients which indicates the effect of each variable on FP. The significance(Pr(>|t|)) indicates which variables are significant and contribute the most to the model. Age and FT. are insignificant with P-values over 0.05 while MP And FG. are significant with a small P-Value below 0.05. The Adjusted R-Squared of the model is 0.3184 meaning 31.84% of the variance in Fantasy Points can be explained using the model.

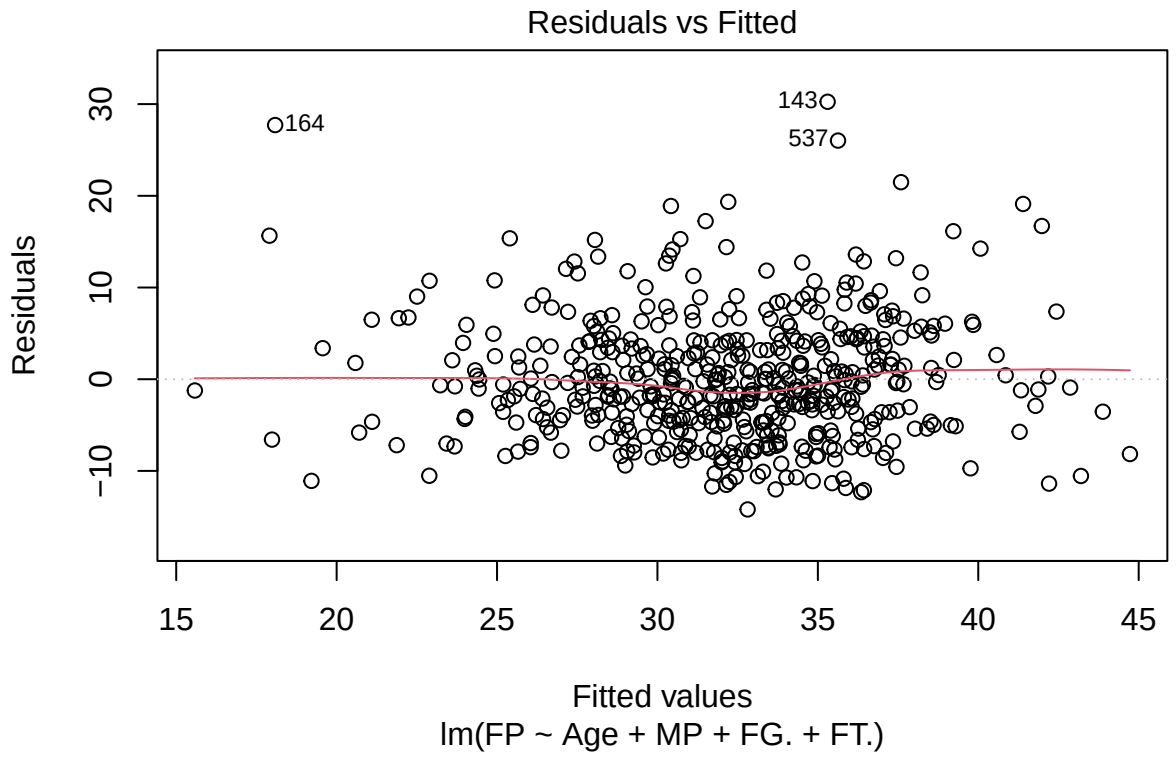
```
#linear regression model and its summary
m_fp <- lm(FP ~ Age + MP + FG. + FT., data = data)
summary(m_fp)
```

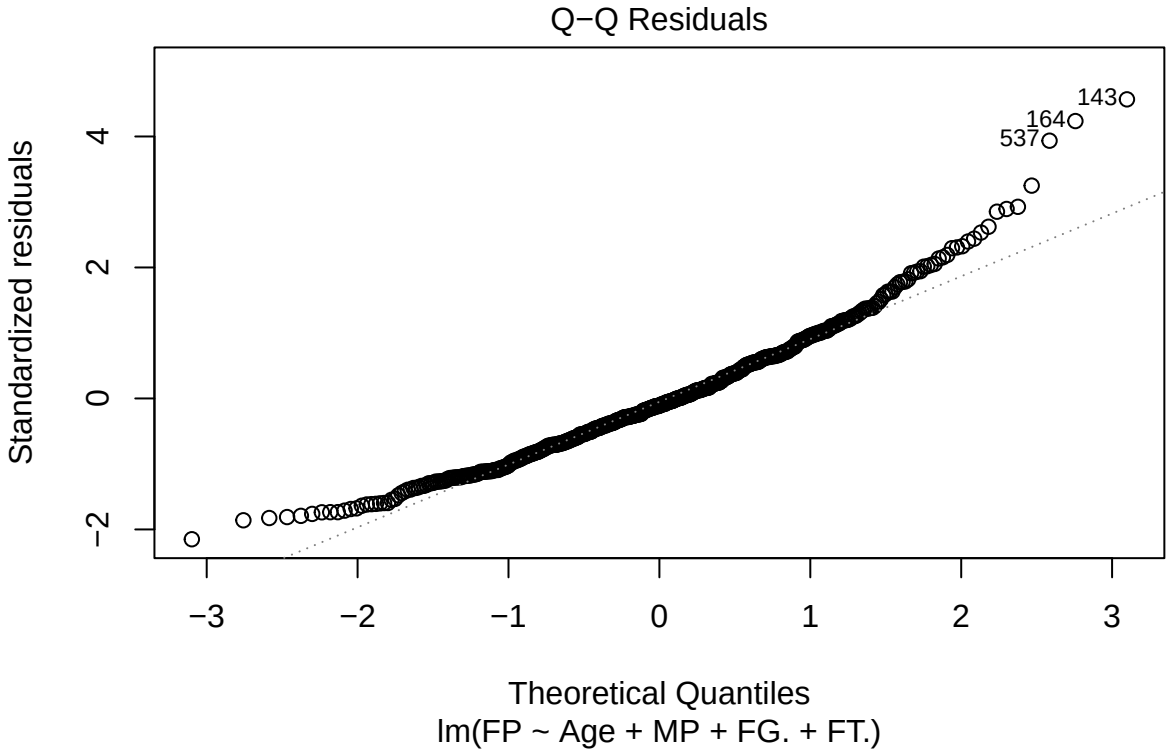
```
##
## Call:
## lm(formula = FP ~ Age + MP + FG. + FT., data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.2004  -4.6223  -0.7211   3.9329  30.2559
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.0769564  2.6525142   4.930 1.11e-06 ***
## Age         -0.0780778  0.0692745  -1.127    0.260
## MP           0.0027071  0.0003846   7.039 6.28e-12 ***
## FG.         39.7141925  3.6108314  10.999  < 2e-16 ***
## FT.         -0.0690619  2.0012242  -0.035    0.972
```

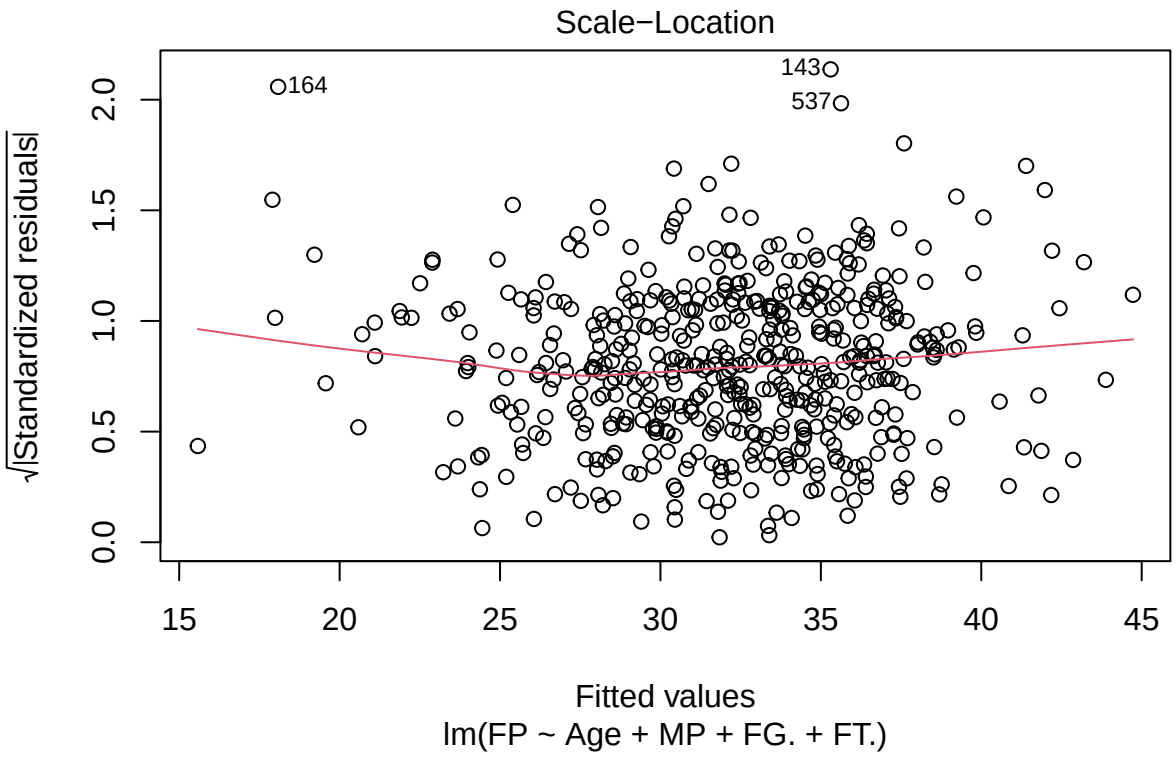
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.644 on 509 degrees of freedom
## Multiple R-squared:  0.3184, Adjusted R-squared:  0.3131
## F-statistic: 59.46 on 4 and 509 DF,  p-value: < 2.2e-16
```

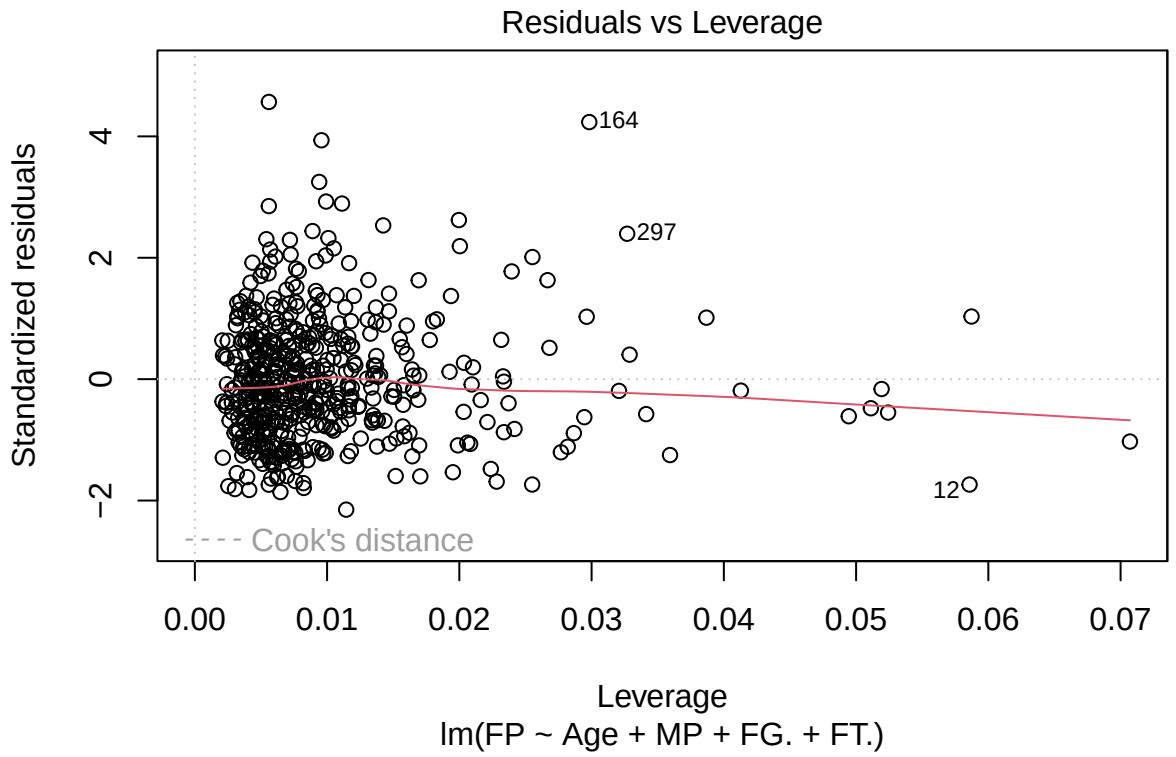
Residuals vs Fitted: Checks for non linearity in residuals. Q-Q Residuals: Examines if the residuals follow a normal distribution Scale-Location: Examines the uniformity of residual variance Residuals vs Leverage: Points out influential data and outliers

```
#diagnostic plots
plot(m_fp)
```




Q−Q Residuals
Standardized residuals
Theoretical Quantiles
lm(FP ~ Age + MP + FG. + FT.)
143
164
537
−2
0
2
4
−3
−2
−1
0
1
2
3

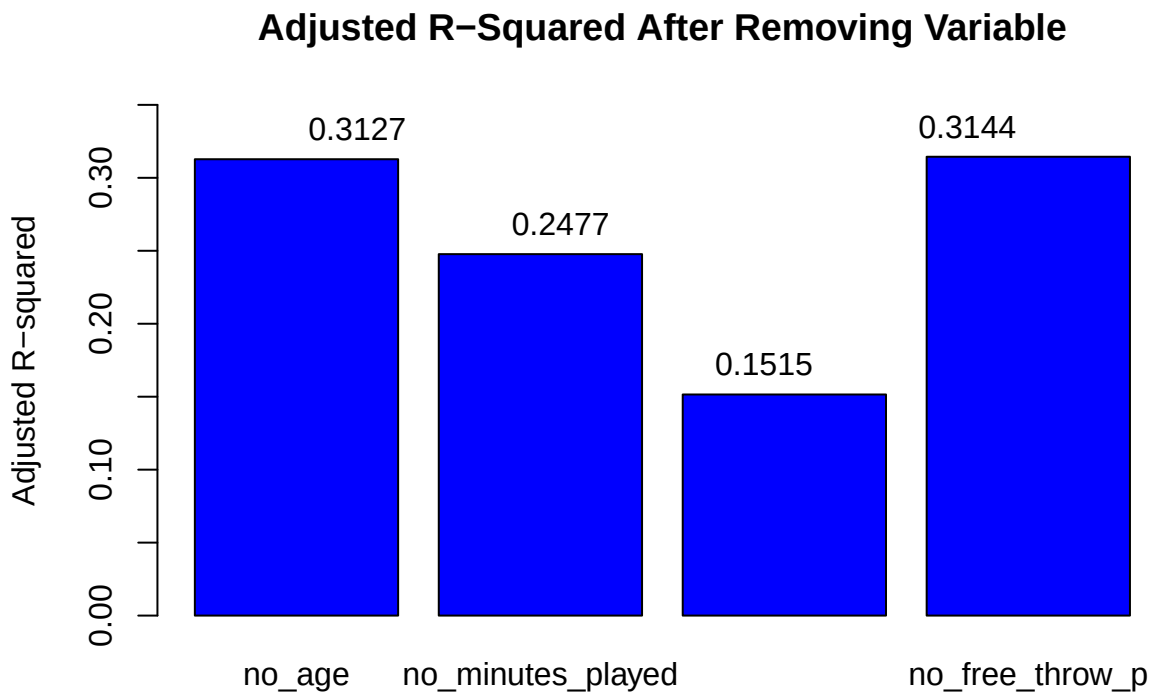

Scale−Location
164
143
537
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
15
20
25
30
35
40
45
Fitted values
lm(FP ~ Age + MP + FG. + FT.)




Each predictor is removed from the model to examine the change in the model's adjusted R-Squared. The bar plot is used to visually show the difference. Removing age or Free Throw Percentage hardly changed the adjusted R-Squared, showing its insignificance in predicting Fantasy Points. On the contrary, Minutes Played and Field Goal Percentage significantly impact the model in predicting FP as seen with the significant drop in adjusted r-squared when removed.

```
#find the adjusted r-squared if we were to remove one of the variables
no_age <- lm(FP ~ MP + FG. + FT., data = data)
r2age <- summary(no_age)$adj.r.squared

no_mp <- lm(FP ~ Age + FG. + FT., data = data)
r2mp <- summary(no_mp)$adj.r.squared

np_fg <- lm(FP ~ Age + MP + FT., data = data)
r2fg <- summary(np_fg)$adj.r.squared

np_ft <- lm(FP ~ Age + MP + FG., data = data)
r2ft <- summary(np_ft)$adj.r.squared
r2_vals <- c(no_age = r2age, no_minutes_played = r2mp, no_field_goal_p = r2fg, no_free_throw_p = r2ft)

#barplot to show differences in r-squared if each variable were to be removed
barplot(r2_vals, main = "Adjusted R-Squared After Removing Variable", col = "blue", ylab = "Adjusted R-
text(x = seq_along(r2_vals),
     y = r2_vals,
     labels = round(r2_vals,4),
     pos = 3,
```

```
     cex = 1,
     col = "black")
```




The model is used to predict FP for statistics based on their Age, MP, FG% and FT%.

```
#predicting FP given values

new_data <- data.frame(Age = c(29, 25, 30, 35, 40),
                       MP = c(37.6, 35, 25, 40, 30),
                       FG. = c(0.56, 0.45, 0.50, 0.40, 0.35),
                       FT. = c(0.81, 0.90, 0.80, 0.95, 0.85))
predictions <- predict(m_fp, newdata = new_data)
print(predictions)
```

```
##        1        2        3        4        5
## 33.09850 29.02899 30.60415 26.27259 23.87632
```

# Analysis#3

```
library(ggplot2)
library(caret)
library(pROC)

suppressWarnings({
    library(ggplot2)
    library(caret)
    library(pROC)
})
```

```
nba_data <- read.csv("nba2024.csv")

# 'High-Scoring' label
average_points <- mean(nba_data$PTS, na.rm = TRUE)
nba_data$HighScoring <- ifelse(nba_data$PTS > average_points, 1, 0)

features <- nba_data[, c("Age", "MP", "FG.")]
target <- as.factor(nba_data$HighScoring)

# missing values handle
nba_data_clean <- na.omit(data.frame(features, HighScoring = target))
```

**Explanation of Relationships/Variables**

In analysis 3, we will predict whether a player is high-scoring (above-average points) based on:

- Age: Older players might bring more experience but could also decline in physical performance
- Minutes Played (MP): Directly related to a player's opportunity to score points
- Field Goal Percentage (FG%): Reflects a player's scoring efficiency

**Explanatory Variables (Independent):**

- Age
- MP
- FG%

**Response Variable (Dependent):**

- High Scoring (binary - 1 = high scoring, 2 = not high scoring)

```
# Logistic regression model
logistic_model <- glm(HighScoring ~ Age + MP + FG., data = nba_data_clean, family = binomial)

summary(logistic_model)
```

```
##
## Call:
## glm(formula = HighScoring ~ Age + MP + FG., family = binomial,
##     data = nba_data_clean)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.4555739  0.7343038  -1.982 0.047451 *
## Age         -0.0541852  0.0228431  -2.372 0.017689 *
## MP           0.0009889  0.0001204   8.215  < 2e-16 ***
## FG.          3.4590367  1.0410514   3.323 0.000892 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 781.48  on 567  degrees of freedom
## Residual deviance: 673.58  on 564  degrees of freedom
## AIC: 681.58
##
## Number of Fisher Scoring iterations: 4
```

**Logistic Regression Interpretation**

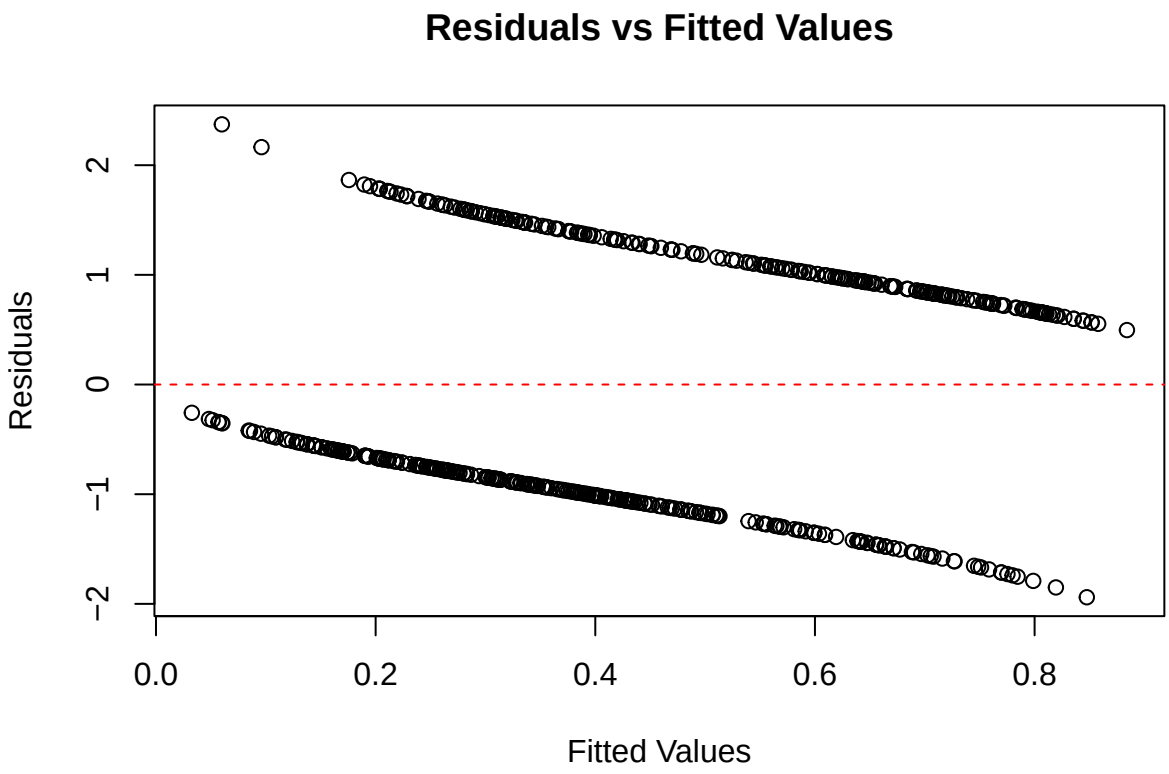
**Coefficients**:

- Intercept: the baseline log odds of being high scoring when "age", "mp", and "FG%" are all zero is -1.46.
- Age: coefficient is -0.054. Each additional year decreases the log-odds of being high scoring by 0.054. P-value is 0.0177, olrder players are slightly less likely to be high scoring
- Minutes Played (MP): coefficient is 0.00099, which means that each additional MP increases the log odds of being high scoring by 0.00099. Since the p-value is significantly low, almost near to 0, playing time has a strong positive effect
- Field Goal Percentage (FG%): Coefficient is 3.459, therefore it means that 1% increase in FG% increases the log odds of being high scoring by 3.46. Since p value is 0.00089, which is very low compared to standard significant level at 0.05, therefore we can say that shooting efficiency is the most impactful predictor.

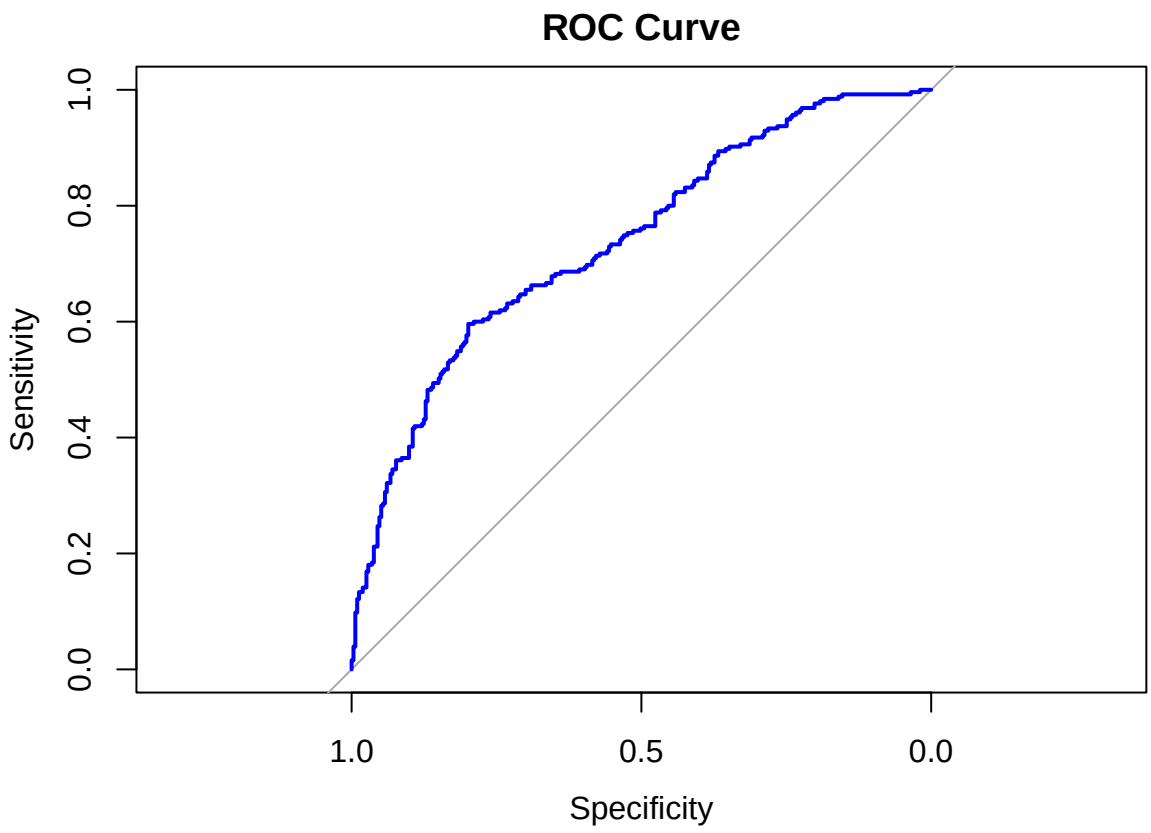
**Model Fit:**

- Null Deviance: 781.48
- Residual Deviance: 673.58 – meaning the model explains some variability in high scoring
- AIC: 681.58: model improves over the null model

```r
plot(logistic_model$fitted.values, residuals(logistic_model),
     xlab = "Fitted Values", ylab = "Residuals", main = "Residuals vs Fitted Values")
abline(h = 0, col = "red", lty = 2)
```




```
roc_curve <- roc(nba_data_clean$HighScoring, logistic_model$fitted.values)
plot(roc_curve, col = "blue", main = "ROC Curve")
```




```
auc(roc_curve)
```

```
## Area under the curve: 0.7379
```

**Residuals vs Fitted**

At the start (fitted values near 0), the lower points (residuals) are close to 0, meaning the model predicts these cases well. However, the upper points are near +2, showing the model overestimated the likelihood of being high-scoring for some players. As we move toward the middle (fitted values near 0.5), the residuals spread out more, with the lower points dropping closer to -2 and the upper points moving down closer to 0, meaning the model is having a hard time predicting probabilities around this range. Toward the end (fitted values near 1), the residuals tighten up again, with the lower points staying near 0 and the upper points closer to +2, showing better predictions for players who are highly likely to be high-scoring. This pattern reflects that the model is more confident for players with very low or very high probabilities but less so for those in the middle range.

**ROC Interpretation**

AUC of 0.7379 means that the model has a 73.79% chance of correctly distinguishing between a high-scoring and a non-high-scoring player. The curve being above the diagonal line (random guess, AUC = 0.5) shows that the model is better than random at classification, but there is room for improvement in its predictive power.

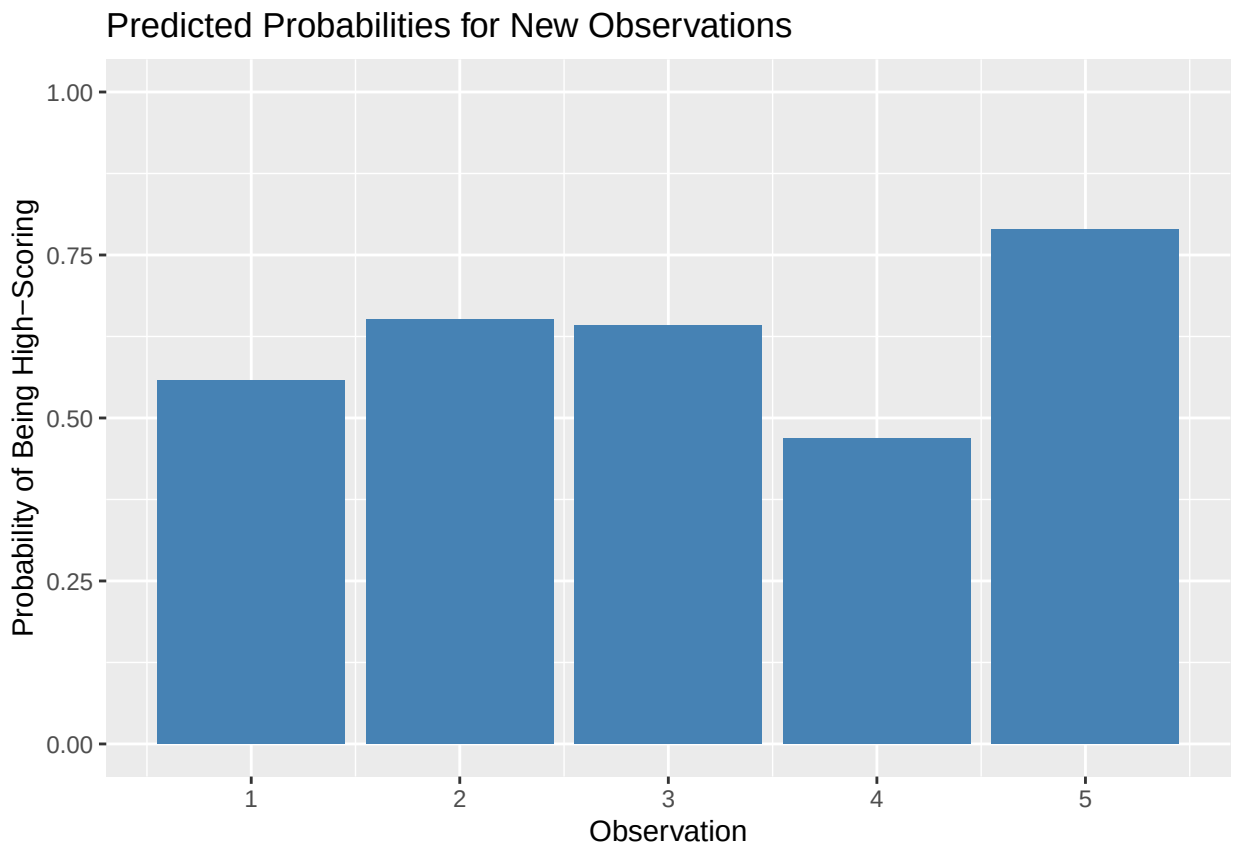
```
new_data <- data.frame(
  Age = c(25, 30, 22, 28, 24),
  MP = c(1500, 2000, 1800, 1200, 2200),
```

```
  FG. = c(0.45, 0.50, 0.42, 0.48, 0.55)
)

predictions <- predict(logistic_model, new_data, type = "response")

data.frame(new_data, Probability = predictions)
```

```
##   Age   MP  FG. Probability
## 1  25 1500 0.45   0.5571880
## 2  30 2000 0.50   0.6516382
## 3  22 1800 0.42   0.6422678
## 4  28 1200 0.48   0.4686148
## 5  24 2200 0.55   0.7895304
```

**Interpretation** - players with higher MP and FG% have a higher probability of being high scoring - player 5, with the highest MP and FG% has the highest predicted probability of 79%.

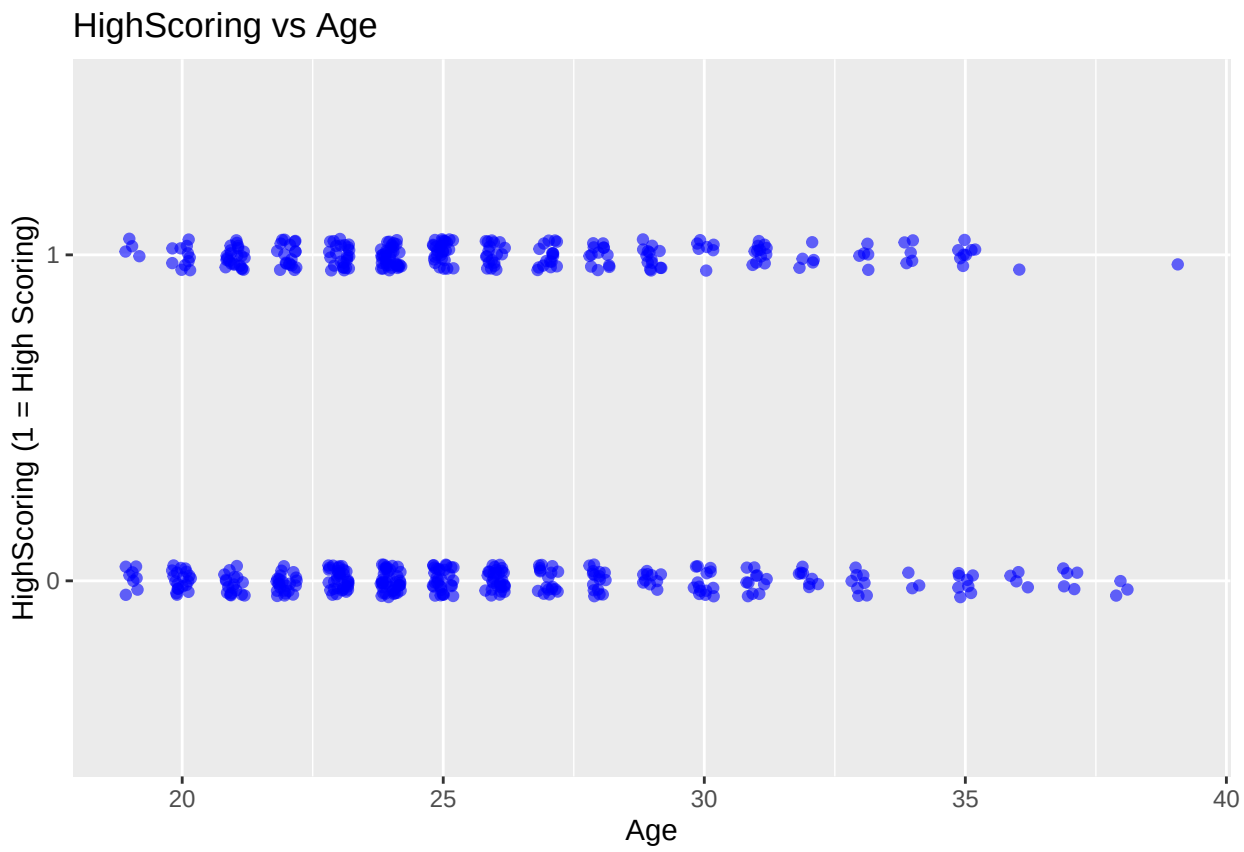
```
new_data$Probability <- predictions

ggplot(new_data, aes(x = 1:5, y = Probability)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(title = "Predicted Probabilities for New Observations", x = "Observation", y = "Probability of Be
  ylim(0, 1)
```



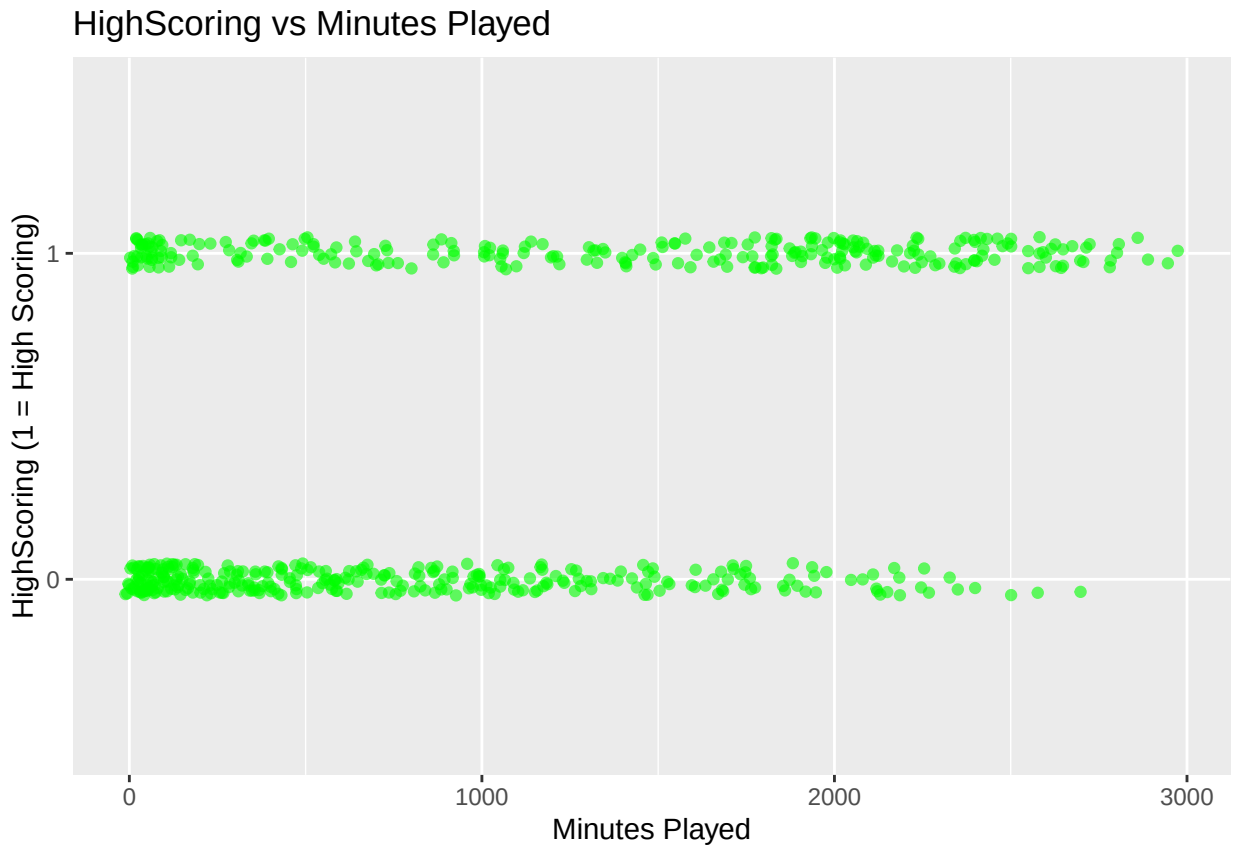

**Interpretation**

Observation 5 has highest predicted probabilty, showing that the model strongly predicts this player as high scoring. Observation 4 has lowest predicted probability, meaning that model considers this player less likely to be high scoring with age of 28. Obersvation 2 and 3 has similar probabilty of high scoring, even though observation 2's age is 30 and 3's age is 22. Observation 1 has slightly lower probability, meaning the player is not going to score high.

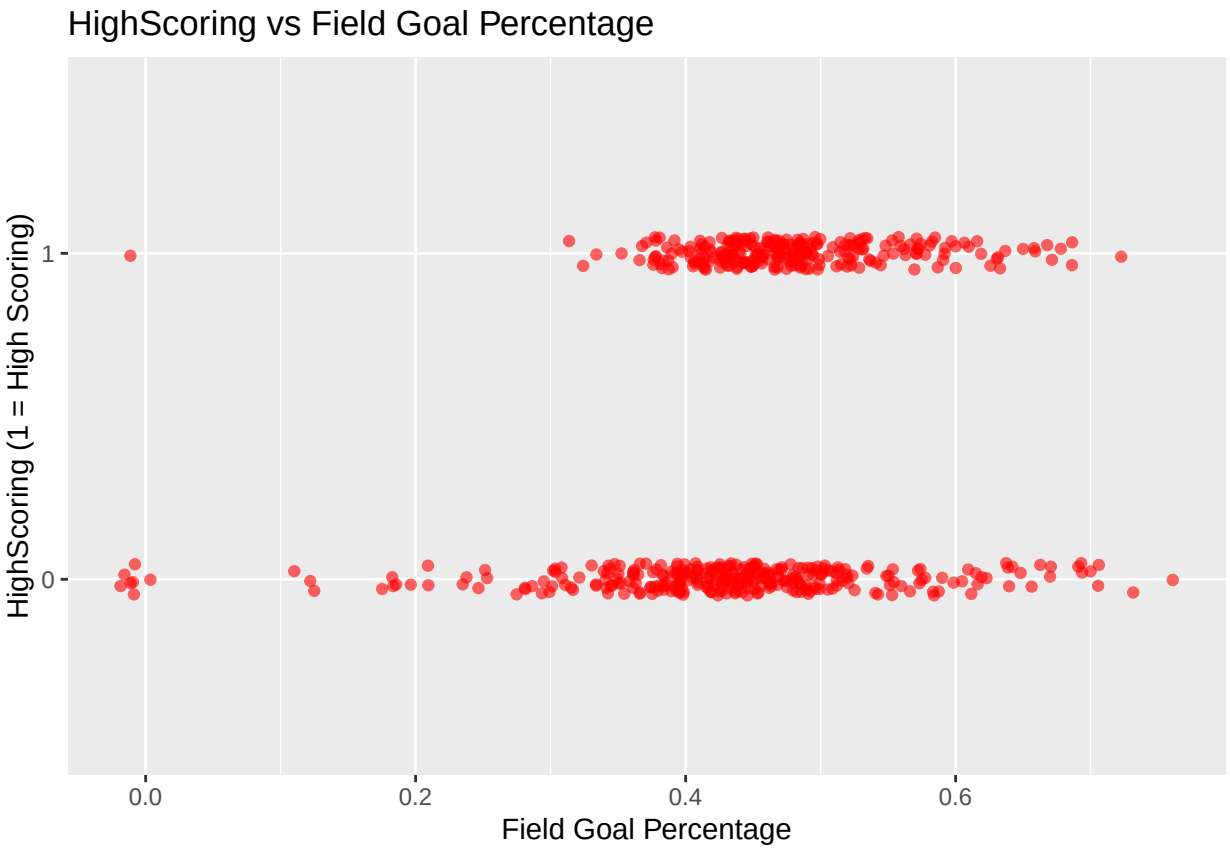
```
#  Age vs HighScoring
ggplot(nba_data_clean, aes(x = Age, y = HighScoring)) +
  geom_jitter(width = 0.2, height = 0.05, color = "blue", alpha = 0.6) +
  labs(title = "HighScoring vs Age", x = "Age", y = "HighScoring (1 = High Scoring)")
```




```
# Minutes Played vs HighScoring
ggplot(nba_data_clean, aes(x = MP, y = HighScoring)) +
  geom_jitter(width = 20, height = 0.05, color = "green", alpha = 0.6) +
  labs(title = "HighScoring vs Minutes Played", x = "Minutes Played", y = "HighScoring (1 = High Scorin
```




```
# Field Goal Percentage vs HighScoring
ggplot(nba_data_clean, aes(x = FG., y = HighScoring)) +
  geom_jitter(width = 0.02, height = 0.05, color = "red", alpha = 0.6) +
  labs(title = "HighScoring vs Field Goal Percentage", x = "Field Goal Percentage", y = "HighScoring (1
```




**High Scoring vs Age**

The points are evenly distributed across ages, with no clear trend, suggesting that age has minimal influence on whether a player is high-scoring. High-scoring players (1) are slightly more concentrated at younger ages, but there is variability across all age ranges.

**High Scoring vs MP**

Players with more minutes played tend to have a higher likelihood of being high-scoring (concentration near $y = 1$ increases as minutes played rise). The separation between high-scoring and non-high-scoring players becomes more apparent as minutes played exceed 1500, meaning playing time is a strong predictor of scoring status.

**High Scoring vs FG%**

Players with higher field goal percentages (above 0.5) are more likely to be high-scoring, as seen by the denser concentration near $y = 1$. Conversely, players with lower field goal percentages (below 0.4) are predominantly non-high-scoring. This means that field goal percentage is a strong predictor of whether a player is high-scoring.

```
predicted_classes <- ifelse(predict(logistic_model, type = "response") > 0.5, 1, 0)
confusion_matrix_logistic <- confusionMatrix(as.factor(predicted_classes), nba_data_clean$HighScoring)
print(confusion_matrix_logistic)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 242 102
```

```
##          1  71 153
##
##                Accuracy : 0.6954
##                  95% CI : (0.6557, 0.7331)
##     No Information Rate : 0.5511
##     P-Value [Acc > NIR] : 1.328e-12
##
##                   Kappa : 0.3774
##
##  Mcnemar's Test P-Value : 0.02256
##
##             Sensitivity : 0.7732
##             Specificity : 0.6000
##          Pos Pred Value : 0.7035
##          Neg Pred Value : 0.6830
##              Prevalence : 0.5511
##          Detection Rate : 0.4261
##    Detection Prevalence : 0.6056
##       Balanced Accuracy : 0.6866
##
##        'Positive' Class : 0
##
```

**Confusion Matrix Interpretation**

1. **Accuracy**:

   - 69.54% : model correctly classifies 69.54 of cases
   - higher than no information rate, so the model performs better than random guessing

2. **Sensitivity (TP)**:

   - 77.32% : model correctly identifies 77.32% of non high scoring players (class 0)
   - 77.32% is pretty high sensitivity, so it means that performance is good in identifying player who are not high scoring

3. **Specificity (TN):**

   - 60% : correctly identifies 60% of high scoring players (class 1)
   - low is good, since it menas that the model struggles more with identifying high scoring players

4. **Positive Predictive Value (FP):**

   - 70.35% : model predicts a player is not high scoring, correct 70.35% of time

5. **Negative Predictive Value:**

   - 68.30% : when model predicts player is high scoring

6. **Kappa Statistic:**

- 0.3774 : moderate agreement.

**McNemar's Test:**

- p value of 0.02256 shows significant imbalance, meaning the model is more likely to misclassify high-scoring players (Class 1) than non-high-scoring players (Class 0).

The random forest model highlights MP (Minutes Played) as the most influential factor, followed by FG% (Field Goal Percentage), with Age having minimal impact.

```
logLik_model <- logLik(logistic_model)
null_model <- glm(HighScoring ~ 1, data = nba_data_clean, family = binomial)
logLik_null <- logLik(null_model)
r2_mcfadden <- 1 - (as.numeric(logLik_model) / as.numeric(logLik_null))
print(r2_mcfadden)
```

```
## [1] 0.1380745
```

**Interpretation**

Performed this test to evaluate how well the logistic regression model fits the data compared to a null model with no predictors. We got the value of 0.138, and this means that the model explains approximately 13.8% of the variability in whether a player is high scoring, which leaves a room for improvement.

**Conclusion**

Logistic regression to predict whether a player is high-scoring (above-average Points) based on their age, playing time (Minutes Played), and shooting efficiency, we can conclude that Minutes Played (MP) and Field Goal Percentage (FG%) are the two significant predictors of high-scoring status. FG% is the most impactful predictor among these three variables. Even though age has a minor negative effect, it is not a strong predictor in comparison to the other two variables. This does not mean that age does not affect a player's performance, as the statistics show that older players are slightly less likely to be high-scoring. However, the effect of age is not as strong as that of MP and FG%. Thus, we can conclude that while age does matter in an NBA player's performance, different factors are more significantly important in a player's performance.