NBA Project

Brady Biehn

```r
# Reading in the three datasets and loading necessary packages
library(tidyverse)
library(mosaic)
library(ggfortify)
library(car)
library(GGally)
teamDefenseStats <- read_csv("team_stats_defense_rs.csv")
teamAdvancedStats <- read_csv("team_stats_advanced_rs.csv")
teamTraditionalStats <- read_csv("team_stats_traditional_rs.csv")
```

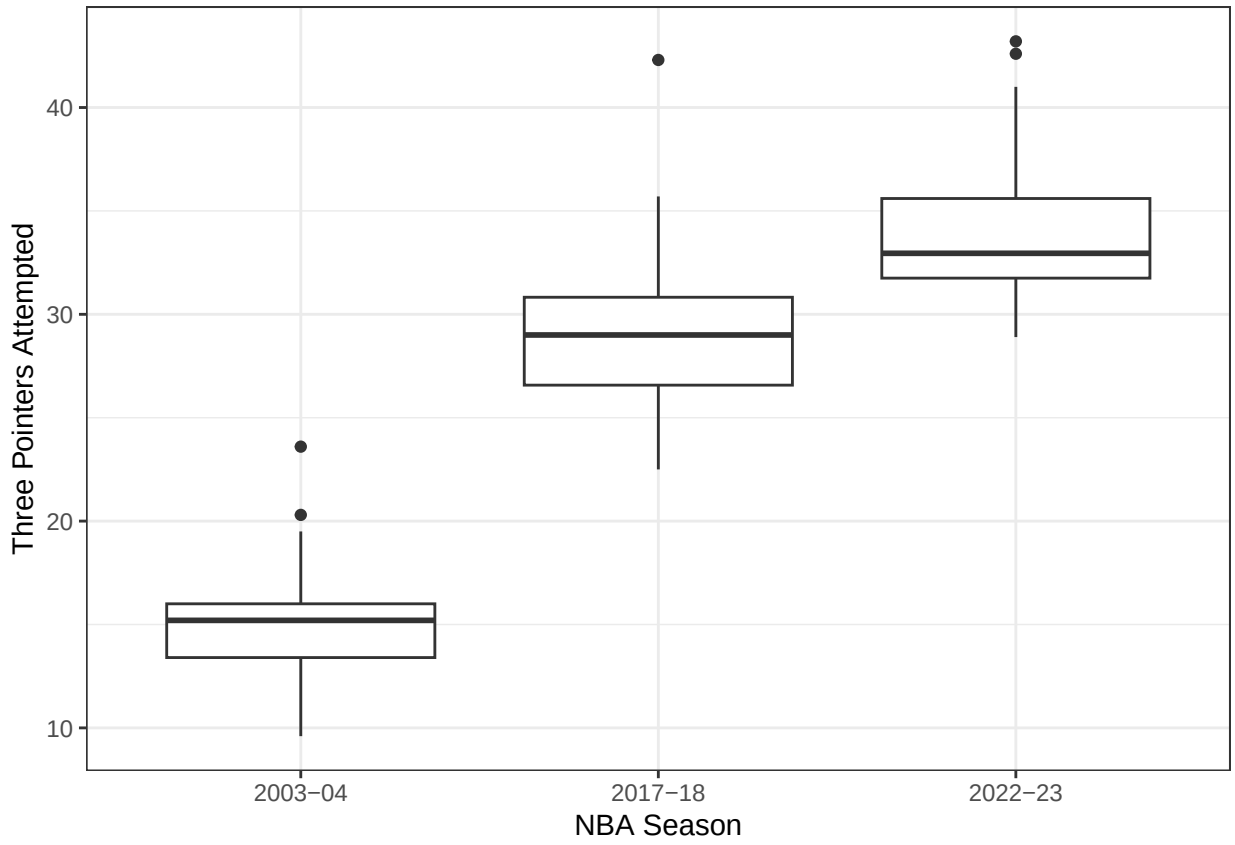
```r
# The data were joined together by both team ID and season, then the particular variables of interest t
fullJoinedData <- teamDefenseStats %>%
  # Selecting to certain entries to protect against duplicates within the final dataset
  select(c(1:15, 17:19, 21, 24, 26, 27, 55)) %>%
  left_join(
    teamAdvancedStats %>%
      select(-c(2:7, 12, 16:18, 21:45)),
    # joining by team_id and season
        by = c("TEAM_ID", "SEASON")) %>%
  select(1, SEASON, everything()) %>%
  # ordering by team name
  arrange(TEAM_NAME)
fullJoinedData <- fullJoinedData %>%
  # Selecting to certain entries to protect against duplicates within the final  dataset
  left_join(
    teamTraditionalStats %>%
      select(-c(2:7,28:54)),
    # joining team_id and season
        by = c("TEAM_ID", "SEASON"))
filteredJoinedData <- fullJoinedData %>%
  # creating a two-point field goal attempts column,
  # field goals normally encompass both three-pointers and two-pointers
  # therefore the three point attempts are removed to only include the two-point attemps
  mutate(FG2A = (FGA - FG3A)) %>%
  mutate(OPP_FG2A = (OPP_FGA - OPP_FG3A)) %>%
  # Converting win percentage to be an actual percentage instead of a proportion
  mutate(W_PCT = W_PCT * 100) %>%
  # Selecting the dataset to only include variables of interest listed above
  select(-c(1, 4:6, 8:9, 11:12, 14:15, 17:22, 24:26, 28:37, 39:40, 42:52)) %>%
  # Organizing columns to group identifier, offense, and defensive columns together
  select(c(1:2, 10:12, 8:9, 7, 6, 12, 4:5, 13)) %>%
  # filtering the dataset to only include data from the three seasons of interest
  filter(SEASON == "2003-04" | SEASON == "2017-18" | SEASON == "2022-23")
# Showing the structure of the dataset
```

```r
glimpse(filteredJoinedData)
```

```
## Rows: 89
## Columns: 12
## $ SEASON     <chr> "2003-04", "2017-18", "2022-23", "2003-04", "2017-18", "202~
## $ TEAM_NAME  <chr> "Atlanta Hawks", "Atlanta Hawks", "Atlanta Hawks", "Boston ~
## $ FTA        <dbl> 24.1, 20.2, 22.6, 25.5, 20.7, 21.6, 22.6, 22.1, 27.0, 23.6,~
## $ PTS        <dbl> 92.8, 103.4, 118.4, 95.3, 104.0, 117.9, 106.6, 113.4, 108.2~
## $ FG2A       <dbl> 64.4, 54.5, 61.9, 58.7, 54.7, 46.2, 51.1, 51.3, 59.5, 57.9,~
## $ DEF_RATING <dbl> 103.9, 110.1, 115.4, 102.2, 103.2, 110.6, 109.7, 113.5, 109~
## $ FG3A       <dbl> 15.2, 31.0, 30.5, 19.5, 30.4, 42.6, 35.7, 33.8, 27.2, 32.5,~
## $ OPP_PTS    <dbl> 97.5, 108.8, 118.1, 96.7, 100.4, 111.4, 110.3, 112.5, 108.0~
## $ OPP_FTA    <dbl> 25.3, 20.6, 23.2, 26.3, 21.3, 21.1, 23.4, 24.4, 18.2, 24.0,~
## $ OPP_FGA    <dbl> 82.2, 86.7, 90.2, 80.8, 85.0, 90.2, 89.4, 88.5, 87.8, 90.1,~
## $ OPP_FG3A   <dbl> 16.4, 30.7, 33.5, 18.7, 27.7, 33.7, 24.5, 32.2, 30.1, 34.3,~
## $ OPP_FG2A   <dbl> 65.8, 56.0, 56.7, 62.1, 57.3, 56.5, 64.9, 56.3, 57.7, 55.8,~
```

```r
# Deriving the summary statistics of FG3A by season
favstats(FG3A ~ SEASON, data = filteredJoinedData)
```

```
##    SEASON  min     Q1 median     Q3  max     mean       sd  n missing
## 1 2003-04  9.6 13.400  15.20 16.000 23.6 14.92069 3.104073 29       0
## 2 2017-18 22.5 26.575  29.00 30.825 42.3 28.99667 4.020506 30       0
## 3 2022-23 28.9 31.750  32.95 35.600 43.2 34.20667 3.693511 30       0
```

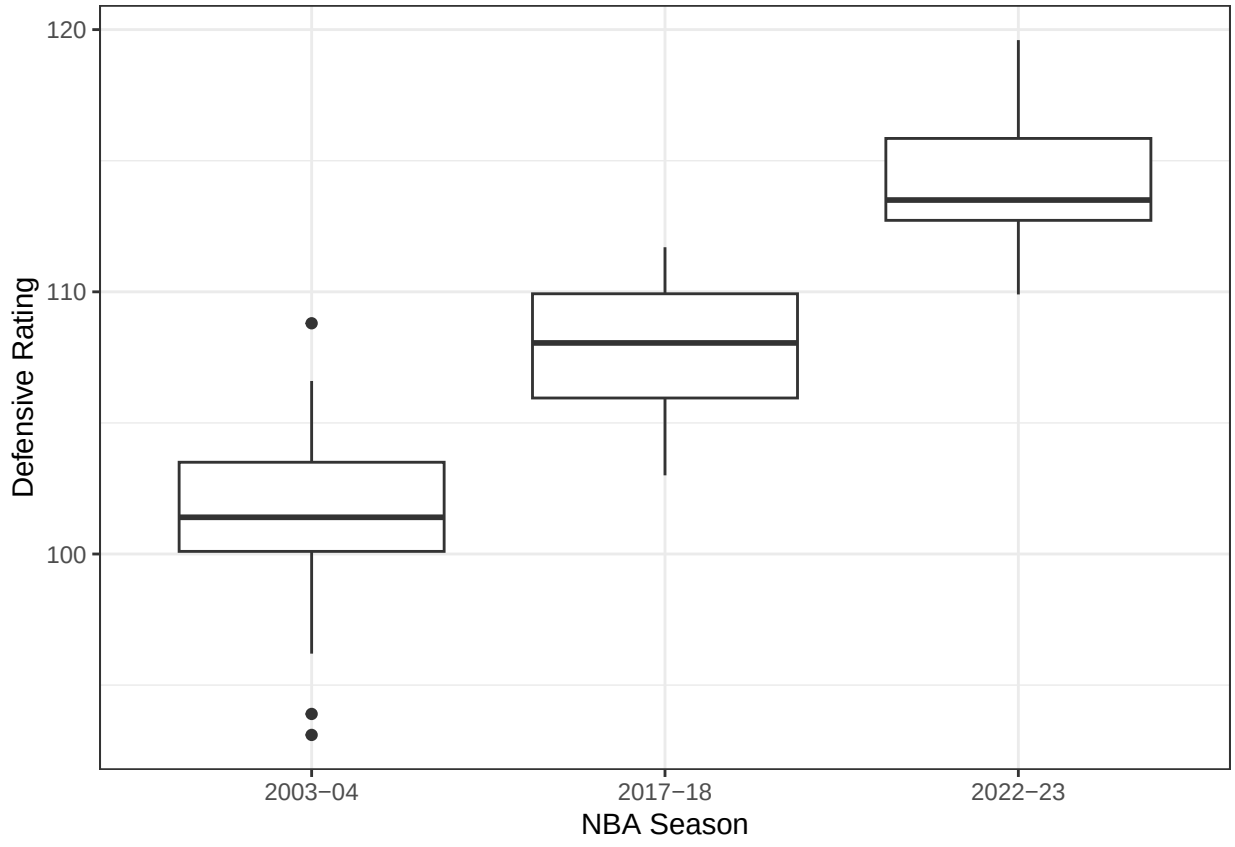


```r
# Showing all teams that shot over 40 three per game
filteredJoinedData %>%
  filter(FG3A > 40)
```

```
## # A tibble: 5 x 12
##   SEASON  TEAM_NAME     FTA   PTS  FG2A DEF_RATING  FG3A OPP_PTS OPP_FTA OPP_FGA
##   <chr>   <chr>       <dbl> <dbl> <dbl>      <dbl> <dbl>   <dbl>   <dbl>   <dbl>
## 1 2022-23 Boston Cel~  21.6  118.  46.2       111.  42.6    111.    21.1    90.2
## 2 2022-23 Dallas Mav~  25.1  114.  43.3       116.  41      114.    25      86.2
## 3 2022-23 Golden Sta~  20.2  119.  47         113.  43.2    117.    25.2    90.5
## 4 2017-18 Houston Ro~  25.1  112.  41.9       106.  42.3    104.    19.6    85.6
## 5 2022-23 Milwaukee ~  22.4  117.  50.1       111.  40.3    113.    21      93.2
## # i 2 more variables: OPP_FG3A <dbl>, OPP_FG2A <dbl>
```

```r
# Deriving the summary statistics of defensive rating by season
favstats(DEF_RATING ~ SEASON, data = filteredJoinedData)
```

```
##    SEASON   min      Q1 median      Q3   max     mean       sd  n missing
## 1 2003-04  93.1 100.100 101.40 103.500 108.8 101.3724 3.640633 29       0
## 2 2017-18 103.0 105.950 108.05 109.925 111.7 107.8400 2.456462 30       0
## 3 2022-23 109.9 112.725 113.50 115.850 119.6 114.0700 2.472490 30       0
```



```r
# finding the outliers for the 2003-04 season
filteredJoinedData %>%
  filter(SEASON == "2003-04") %>%
  filter(DEF_RATING < 95 | DEF_RATING > 108)
```

```
## # A tibble: 3 x 12
##   SEASON  TEAM_NAME     FTA   PTS  FG2A DEF_RATING  FG3A OPP_PTS OPP_FTA OPP_FGA
##   <chr>   <chr>       <dbl> <dbl> <dbl>      <dbl> <dbl>   <dbl>   <dbl>   <dbl>
## 1 2003-04 Detroit Pi~  25.3  90.1  65.2       93.9  11.8    84.3    21.1    77.8
## 2 2003-04 Orlando Ma~  24.4  94    67.3      109.   15.2   101.     24      83
## 3 2003-04 San Antoni~  25.2  91.5  64.6       93.1  13.9    84.3    22.5    77.9
## # i 2 more variables: OPP_FG3A <dbl>, OPP_FG2A <dbl>
```

```r
# Seeing if anyone matches with 2003-04 Orlando Magic in the present game
filteredJoinedData %>%
  filter(SEASON == "2022-23") %>%
  filter(DEF_RATING <= 108.8)
```

```
## # A tibble: 0 x 12
## # i 12 variables: SEASON <chr>, TEAM_NAME <chr>, FTA <dbl>, PTS <dbl>,
## #   FG2A <dbl>, DEF_RATING <dbl>, FG3A <dbl>, OPP_PTS <dbl>, OPP_FTA <dbl>,
## #   OPP_FGA <dbl>, OPP_FG3A <dbl>, OPP_FG2A <dbl>
```

```r
pointsScoredModel <- lm(PTS ~ FG3A + FG2A + FTA, data = filteredJoinedData)
summary(pointsScoredModel)
```

```
##
## Call:
## lm(formula = PTS ~ FG3A + FG2A + FTA, data = filteredJoinedData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1907 -2.0659 -0.3214  1.6266  8.7310
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -16.08191   10.30698  -1.560    0.122
## FG3A          1.63818    0.08667  18.901  < 2e-16 ***
## FG2A          1.03422    0.12462   8.299 1.42e-12 ***
## FTA           0.75786    0.14607   5.188 1.42e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.045 on 85 degrees of freedom
## Multiple R-squared:  0.9006, Adjusted R-squared:  0.8971
## F-statistic: 256.7 on 3 and 85 DF,  p-value: < 2.2e-16
```

```r
# Conducting partial F-tests test the significance of each
# of the explanatory variables in predicting points
pointsScoredReducedFree <-  lm(PTS ~ FG3A + FG2A, data = filteredJoinedData)
anova(pointsScoredReducedFree, pointsScoredModel)
```

```
## Analysis of Variance Table
##
## Model 1: PTS ~ FG3A + FG2A
## Model 2: PTS ~ FG3A + FG2A + FTA
##   Res.Df     RSS Df Sum of Sq      F    Pr(>F)
## 1     86 1037.60
## 2     85  788.04  1    249.56 26.918 1.422e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
pointsScoredReducedThree <- lm(PTS ~  FG2A + FTA, data = filteredJoinedData)
anova(pointsScoredReducedThree, pointsScoredModel)
```

```
## Analysis of Variance Table
##
## Model 1: PTS ~ FG2A + FTA
## Model 2: PTS ~ FG3A + FG2A + FTA
##   Res.Df  RSS Df Sum of Sq      F    Pr(>F)
## 1     86 4100
## 2     85  788  1      3312 357.24 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
pointsScoredReducedTwo <- lm(PTS ~ FG3A + FTA, data = filteredJoinedData)
anova(pointsScoredReducedTwo, pointsScoredModel)
```

```
## Analysis of Variance Table
##
## Model 1: PTS ~ FG3A + FTA
## Model 2: PTS ~ FG3A + FG2A + FTA
##   Res.Df     RSS Df Sum of Sq      F    Pr(>F)
## 1     86 1426.53
## 2     85  788.04  1    638.49 68.869 1.415e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Creating a 95% confidence interval for all of the variables in
# each of the variables in the model to gauge range of the values
confint(pointsScoredModel)
```

```
##                   2.5 %   97.5 %
## (Intercept) -36.5749376 4.411121
## FG3A          1.4658494 1.810506
## FG2A          0.7864338 1.282003
## FTA           0.4674293 1.048290
```

```
# SHOWING THE CONCERN OF MULTICOLINEARITY
# Creating the model with the three-pointers and two-pointers separate
defenseModel <- lm(DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON + SEASON * OPP_FG3A + SEASON * O
# Calculating the VIFs
vif(defenseModel, type = 'predictor')
```

```
## GVIFs computed for predictors
```

```
##               GVIF Df GVIF^(1/(2*Df))              Interacts With
## OPP_FG3A 1068203.9  5        4.007425                      SEASON
## OPP_FG2A  380970.6  5        3.614841                      SEASON
## OPP_FTA  1664281.0  5        4.189119                      SEASON
## SEASON         1.0 11        1.000000 OPP_FG3A, OPP_FG2A, OPP_FTA
##            Other Predictors
## OPP_FG3A  OPP_FG2A, OPP_FTA
## OPP_FG2A  OPP_FG3A, OPP_FTA
## OPP_FTA  OPP_FG3A, OPP_FG2A
## SEASON                  --
```

```
summary(defenseModel)
```

```
##
## Call:
## lm(formula = DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON +
##     SEASON * OPP_FG3A + SEASON * OPP_FG2A + SEASON * OPP_FTA,
##     data = filteredJoinedData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.5447 -1.6405  0.1399  1.4435  4.7298
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               15.8926    15.5138   1.024 0.308846
## OPP_FG3A                   1.4237     0.2655   5.363 8.31e-07 ***
## OPP_FG2A                   0.7702     0.1787   4.309 4.79e-05 ***
## OPP_FTA                    0.5883     0.2306   2.551 0.012712 *
## SEASON2017-18             59.7824    22.8488   2.616 0.010690 *
## SEASON2022-23             63.9106    23.1841   2.757 0.007288 **
## OPP_FG3A:SEASON2017-18    -0.7152     0.3726  -1.919 0.058653 .
## OPP_FG3A:SEASON2022-23    -1.2560     0.3671  -3.422 0.000999 ***
## OPP_FG2A:SEASON2017-18    -0.6717     0.2471  -2.718 0.008113 **
## OPP_FG2A:SEASON2022-23    -0.5539     0.2424  -2.285 0.025079 *
## OPP_FTA:SEASON2017-18     -0.3114     0.3166  -0.984 0.328405
## OPP_FTA:SEASON2022-23      0.1272     0.3646   0.349 0.728214
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.299 on 77 degrees of freedom
## Multiple R-squared:  0.8688, Adjusted R-squared:   0.85
## F-statistic: 46.35 on 11 and 77 DF,  p-value: < 2.2e-16
```

```
# Creating the model with the opponent's field goal attempted variable
defenseModel <- lm(DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON
                   + SEASON * OPP_FG2A + OPP_FTA * SEASON,
                   data = filteredJoinedData)
# Calculating the VIFs
vif(defenseModel)
```

```
## there are higher-order terms (interactions) in this model
## consider setting type = 'predictor'; see ?vif
```

```
##                          GVIF Df GVIF^(1/(2*Df))
## OPP_FG3A        2.625683e+01  1        5.124142
## OPP_FG2A        1.485534e+01  1        3.854263
## OPP_FTA         4.074696e+00  1        2.018588
## SEASON          5.243119e+05  2       26.908991
## OPP_FG2A:SEASON 1.725651e+05  2       20.381599
## OPP_FTA:SEASON  2.810578e+04  2       12.947887
```

```
# The VIF OPP_FGA < 10 means that multicolinearity is fine and not a concern :)
```

```
# Largest model all main effects and interaction terms
defenseModel <- lm(DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON
                   + SEASON * OPP_FG2A + OPP_FTA * SEASON + OPP_FG3A * SEASON,
                   data = filteredJoinedData)
# Reduced model of all main effect and interaction terms except for the season and opposing teams three
defenseRecudedModelThree <- lm(DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON
                   + SEASON * OPP_FG2A + OPP_FTA * SEASON,
                   data = filteredJoinedData)
# Reduced model all main effects and interaction terms except the season and opposing teams free-throw
defenseReducedTwoModel <- lm(DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON
                    + OPP_FTA * SEASON + OPP_FG3A * SEASON,
                   data = filteredJoinedData)

# Conducting a partial F-test to see the necessity of the season and opposing teams free-throw attempts
defenseReducedFreeThrowModel <- lm(DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON
                   + SEASON * OPP_FG2A + OPP_FG3A * SEASON,
                   data = filteredJoinedData)
# Conducting anova tests to measure
anova(defenseRecudedModelThree, defenseModel)
```

```
## Analysis of Variance Table
##
## Model 1: DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON + SEASON *
##     OPP_FG2A + OPP_FTA * SEASON
## Model 2: DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON + SEASON *
##     OPP_FG2A + OPP_FTA * SEASON + OPP_FG3A * SEASON
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     79 468.92
## 2     77 406.88  2    62.042 5.8705 0.004237 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(defenseReducedTwoModel, defenseModel)
```

```
## Analysis of Variance Table
##
## Model 1: DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON + OPP_FTA *
##     SEASON + OPP_FG3A * SEASON
## Model 2: DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON + SEASON *
```

```
##     OPP_FG2A + OPP_FTA * SEASON + OPP_FG3A * SEASON
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     79 451.07
## 2     77 406.88  2     44.19 4.1813 0.01888 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(defenseReducedFreeThrowModel, defenseModel)
```

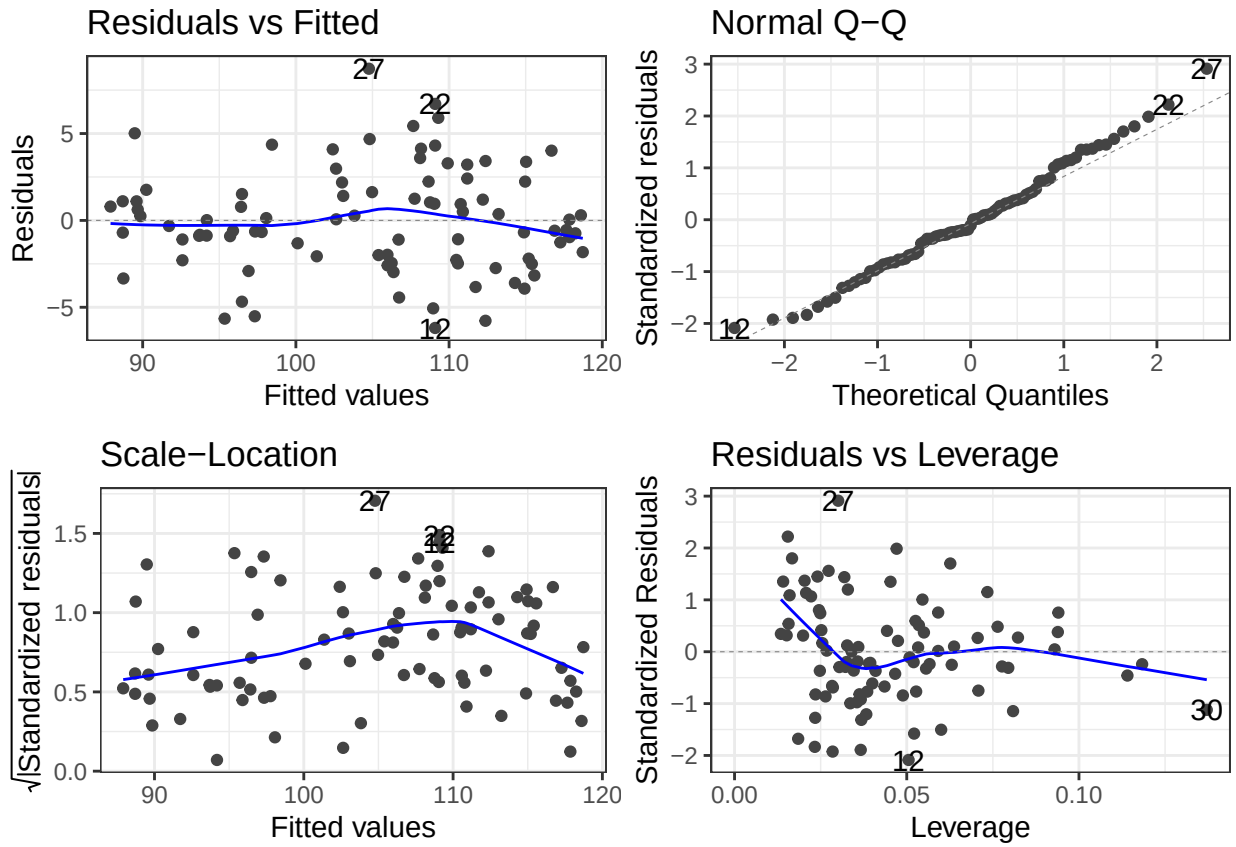
```
## Analysis of Variance Table
##
## Model 1: DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON + SEASON *
##     OPP_FG2A + OPP_FG3A * SEASON
## Model 2: DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON + SEASON *
##     OPP_FG2A + OPP_FTA * SEASON + OPP_FG3A * SEASON
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     79 416.30
## 2     77 406.88  2    9.4218 0.8915 0.4142
```

```
# Renaming final model for a less complex name
finalDefensiveModel <-
  lm(DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON
    + SEASON * OPP_FG3A + SEASON * OPP_FG2A, data = filteredJoinedData)
summary(finalDefensiveModel)
```

```
##
## Call:
## lm(formula = DEF_RATING ~ OPP_FG3A + OPP_FG2A + OPP_FTA + SEASON +
##     SEASON * OPP_FG3A + SEASON * OPP_FG2A, data = filteredJoinedData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.4464 -1.5535  0.0286  1.4954  4.7425
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              19.6098    13.7256   1.429 0.157032
## OPP_FG3A                  1.4353     0.2641   5.434 5.94e-07 ***
## OPP_FG2A                  0.7459     0.1722   4.332 4.30e-05 ***
## OPP_FTA                   0.4928     0.1377   3.578 0.000595 ***
## SEASON2017-18            48.6373    19.6235   2.479 0.015324 *
## SEASON2022-23            66.9865    19.5679   3.423 0.000983 ***
## OPP_FG3A:SEASON2017-18   -0.6577     0.3696  -1.780 0.078971 .
## OPP_FG3A:SEASON2022-23   -1.3003     0.3650  -3.562 0.000627 ***
## OPP_FG2A:SEASON2017-18   -0.6344     0.2409  -2.634 0.010161 *
## OPP_FG2A:SEASON2022-23   -0.5377     0.2366  -2.273 0.025761 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.296 on 79 degrees of freedom
## Multiple R-squared:  0.8657, Adjusted R-squared:  0.8504
## F-statistic:  56.6 on 9 and 79 DF,  p-value: < 2.2e-16
```

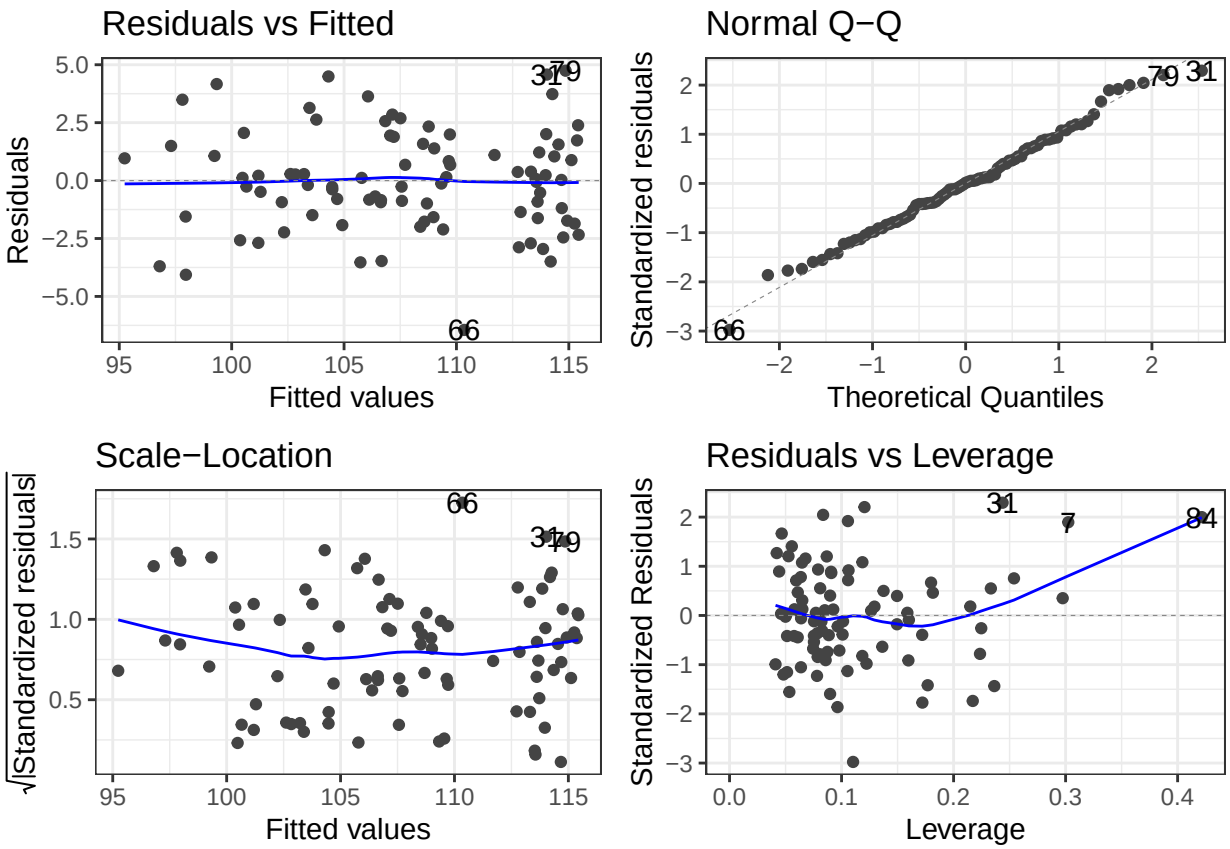
```r
confint(finalDefensiveModel)
```

```
##                                 2.5 %       97.5 %
## (Intercept)                -7.7103278  46.92989273
## OPP_FG3A                    0.9095183   1.96105290
## OPP_FG2A                    0.4031663   1.08858722
## OPP_FTA                     0.2186696   0.76692164
## SEASON2017-18               9.5776036  87.69692793
## SEASON2022-23              28.0374501 105.93548690
## OPP_FG3A:SEASON2017-18     -1.3933740   0.07788954
## OPP_FG3A:SEASON2022-23     -2.0269010  -0.57371404
## OPP_FG2A:SEASON2017-18     -1.1137866  -0.15491503
## OPP_FG2A:SEASON2022-23     -1.0085480  -0.06677729
```




```r
# Calculating VIF to test for multicolinearity
vif(pointsScoredModel)
```

```
##     FG3A     FG2A      FTA
## 5.656860 5.532300 1.060428
```




```
# Calculating GVIF to test for multicolinearity
vif(finalDefensiveModel, type = "predictor")
```

```
## GVIFs computed for predictors
```

```
##                   GVIF Df GVIF^(1/(2*Df))     Interacts With
## OPP_FG3A 4.188003e+05  5        3.649225              SEASON
## OPP_FG2A 7.968451e+04  5        3.091273              SEASON
## OPP_FTA  1.468989e+00  1        1.212018                --
## SEASON   1.468989e+00  8        1.024327 OPP_FG3A, OPP_FG2A
##                    Other Predictors
## OPP_FG3A           OPP_FG2A, OPP_FTA
## OPP_FG2A           OPP_FG3A, OPP_FTA
## OPP_FTA  OPP_FG3A, OPP_FG2A, SEASON
## SEASON                     OPP_FTA
```