# 410 EDA

## 410 Project EDA

```r
wbb <- read.csv("C:/Users/gigip/personal/personal projects/march madness/WRegularSeasonDetai

# make effective field goal percentage variable
wbb <- wbb %>%
  mutate(WeFG = (WFGM + 0.5 * WFGM3) / WFGA,
         LeFG = (LFGM + 0.5 * LFGM3) / LFGA)
```

look at variables

```r
summary(wbb)
```

```
     Season         DayNum          WTeamID          WScore          LTeamID
 Min.   :2010   Min.   :  0.00   Min.   :3101   Min.   : 30.00   Min.   :3101
 1st Qu.:2013   1st Qu.: 36.00   1st Qu.:3196   1st Qu.: 64.00   1st Qu.:3195
 Median :2017   Median : 73.00   Median :3283   Median : 71.00   Median :3287
 Mean   :2017   Mean   : 69.55   Mean   :3285   Mean   : 71.71   Mean   :3287
 3rd Qu.:2022   3rd Qu.:101.00   3rd Qu.:3376   3rd Qu.: 79.00   3rd Qu.:3377
 Max.   :2025   Max.   :132.00   Max.   :3480   Max.   :140.00   Max.   :3480
     LScore           WLoc               NumOT              WFGM
 Min.   : 11.00   Length:81308       Min.   :0.00000   Min.   : 9.00
 1st Qu.: 50.00   Class :character   1st Qu.:0.00000   1st Qu.:22.00
 Median : 57.00   Mode  :character   Median :0.00000   Median :25.00
 Mean   : 57.26                      Mean   :0.05167   Mean   :25.85
 3rd Qu.: 64.00                      3rd Qu.:0.00000   3rd Qu.:29.00
 Max.   :130.00                      Max.   :5.00000   Max.   :58.00
      WFGA            WFGM3             WFGA3             WFTM
 Min.   : 30.00   Min.   : 0.000   Min.   : 0.00   Min.   : 0.00
```

```
 1st Qu.: 53.00   1st Qu.: 4.000   1st Qu.:13.00   1st Qu.:10.00  
 Median : 59.00   Median : 6.000   Median :17.00   Median :13.00  
 Mean   : 58.96   Mean   : 6.282   Mean   :17.97   Mean   :13.73  
 3rd Qu.: 64.00   3rd Qu.: 8.000   3rd Qu.:22.00   3rd Qu.:17.00  
 Max.   :113.00   Max.   :30.000   Max.   :63.00   Max.   :49.00  
      WFTA            WOR             WDR             WAst      
 Min.   : 0.00   Min.   : 0.00   Min.   : 3.00   Min.   : 1.00  
 1st Qu.:14.00   1st Qu.: 9.00   1st Qu.:23.00   1st Qu.:12.00  
 Median :19.00   Median :12.00   Median :26.00   Median :15.00  
 Mean   :19.29   Mean   :12.08   Mean   :26.65   Mean   :14.96  
 3rd Qu.:24.00   3rd Qu.:15.00   3rd Qu.:30.00   3rd Qu.:18.00  
 Max.   :66.00   Max.   :45.00   Max.   :58.00   Max.   :45.00  
      WTO             WStl             WBlk              WPF       
 Min.   : 1.00   Min.   : 0.000   Min.   : 0.000   Min.   : 1.00  
 1st Qu.:12.00   1st Qu.: 6.000   1st Qu.: 2.000   1st Qu.:13.00  
 Median :15.00   Median : 8.000   Median : 3.000   Median :16.00  
 Mean   :15.04   Mean   : 8.636   Mean   : 3.677   Mean   :16.05  
 3rd Qu.:18.00   3rd Qu.:11.000   3rd Qu.: 5.000   3rd Qu.:19.00  
 Max.   :40.00   Max.   :36.000   Max.   :23.000   Max.   :37.00  
      LFGM            LFGA             LFGM3            LFGA3      
 Min.   : 3.00   Min.   : 25.00   Min.   : 0.000   Min.   : 0.00  
 1st Qu.:18.00   1st Qu.: 53.00   1st Qu.: 3.000   1st Qu.:13.00  
 Median :21.00   Median : 58.00   Median : 5.000   Median :17.00  
 Mean   :20.89   Mean   : 58.04   Mean   : 4.968   Mean   :17.93  
 3rd Qu.:24.00   3rd Qu.: 63.00   3rd Qu.: 7.000   3rd Qu.:22.00  
 Max.   :45.00   Max.   :111.00   Max.   :25.000   Max.   :80.00  
      LFTM            LFTA            LOR             LDR             LAst      
 Min.   : 0.00   Min.   : 0.0   Min.   : 0.00   Min.   : 1.00   Min.   : 0.00  
 1st Qu.: 7.00   1st Qu.:11.0   1st Qu.: 8.00   1st Qu.:19.00   1st Qu.: 8.00  
 Median :10.00   Median :15.0   Median :11.00   Median :22.00   Median :11.00  
 Mean   :10.51   Mean   :15.5   Mean   :11.34   Mean   :22.41   Mean   :10.94  
 3rd Qu.:14.00   3rd Qu.:20.0   3rd Qu.:14.00   3rd Qu.:26.00   3rd Qu.:13.00  
 Max.   :37.00   Max.   :52.0   Max.   :38.00   Max.   :53.00   Max.   :34.00  
      LTO             LStl             LBlk             LPF       
 Min.   : 1.00   Min.   : 0.000   Min.   : 0.00   Min.   : 3.00  
 1st Qu.:13.00   1st Qu.: 5.000   1st Qu.: 1.00   1st Qu.:15.00  
 Median :17.00   Median : 7.000   Median : 2.00   Median :18.00  
 Mean   :17.13   Mean   : 7.109   Mean   : 2.82   Mean   :18.18  
 3rd Qu.:20.00   3rd Qu.: 9.000   3rd Qu.: 4.00   3rd Qu.:21.00  
 Max.   :49.00   Max.   :26.000   Max.   :21.00   Max.   :47.00  
      WeFG             LeFG       
 Min.   :0.1899   Min.   :0.0600  
 1st Qu.:0.4386   1st Qu.:0.3529  
```

```
 Median :0.4909   Median :0.4032  
 Mean   :0.4932   Mean   :0.4041  
 3rd Qu.:0.5446   3rd Qu.:0.4537  
 Max.   :0.9592   Max.   :0.7619  
```

Variables I want to look at:

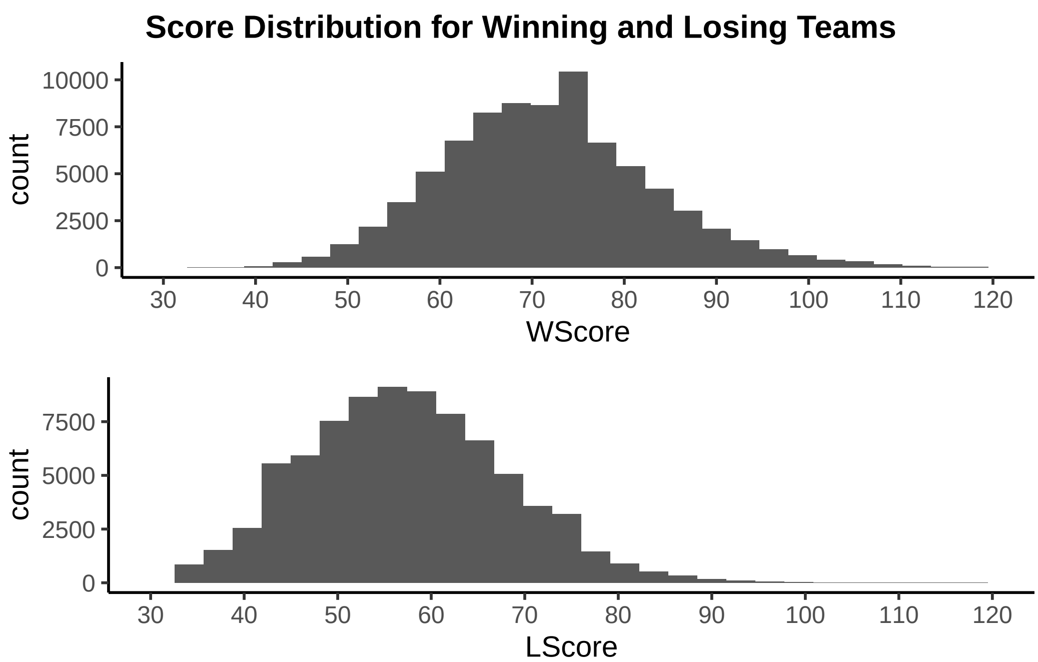
- eFG%
  - eFG% = (Field Goals Made + 0.5 * Three-Point Field Goals Made) / Field Goal Attempts.
- num TO
- FT %
- num assists
- num rebounds
- score and score difference between winning and losing teams
- location (as factor)

#### Histograms and Boxplots of Score Distribution for Winning and Losing Teams

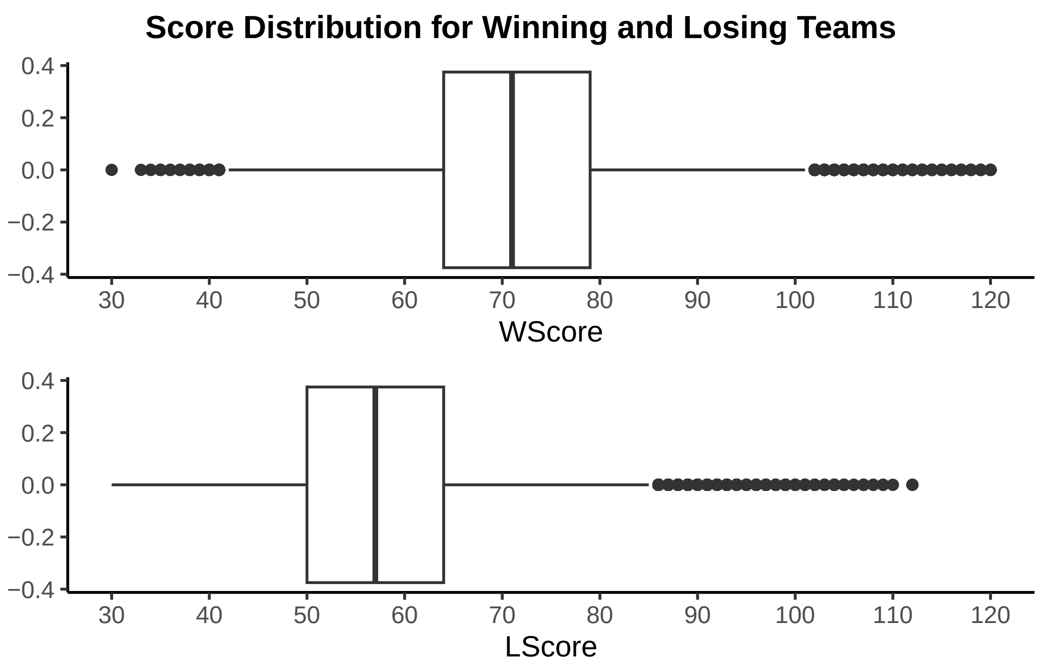
```r
wscore <- wbb %>%
  ggplot(aes(x = WScore)) +
  geom_histogram() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(30,120))

lscore <- wbb %>%
  ggplot(aes(x = LScore)) +
  geom_histogram() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(30,120))

plt <- ggarrange(wscore, lscore, ncol = 1, nrow = 2)
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
annotate_figure(plt, top = text_grob("Score Distribution for Winning and Losing Teams", face
```




```
wscore <- wbb %>%
  ggplot(aes(x = WScore)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 9, limits = c(30,120))
lscore <- wbb %>%
  ggplot(aes(x = LScore)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 9, limits = c(30,120))

plt <- ggarrange(wscore, lscore, ncol = 1, nrow = 2)
annotate_figure(plt, top = text_grob("Score Distribution for Winning and Losing Teams", face
```




### Score Difference Between Winning and Losing Teams

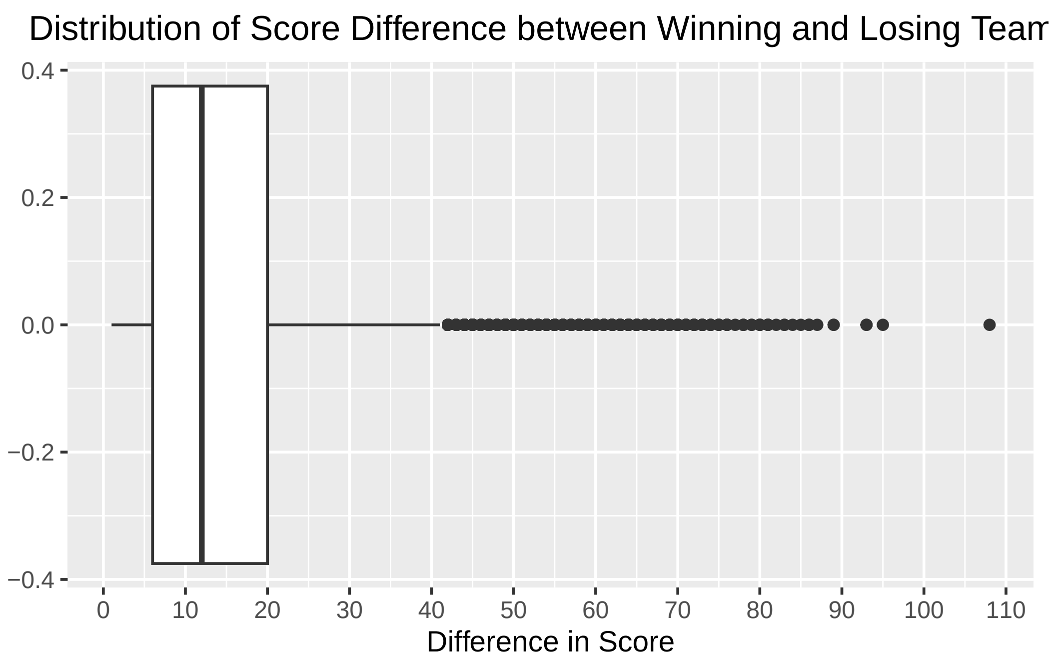
```
score_diff <- wbb %>%
  mutate(difference = WScore - LScore)
mean(score_diff$difference)
```

```
[1] 14.45296
```

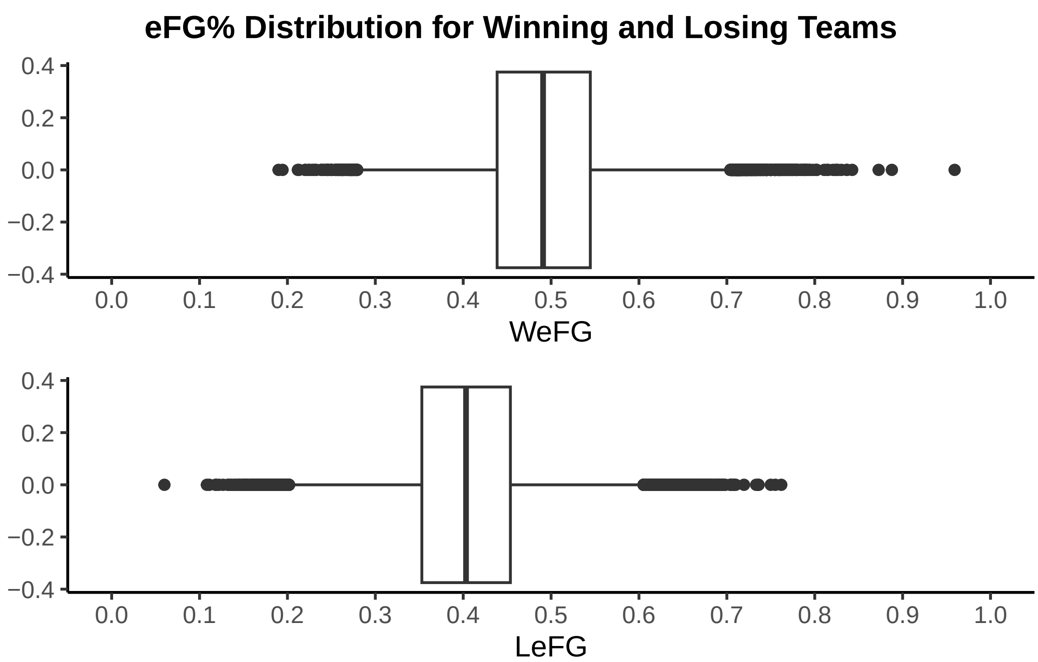
```
median(score_diff$difference)
```

```
[1] 12
```

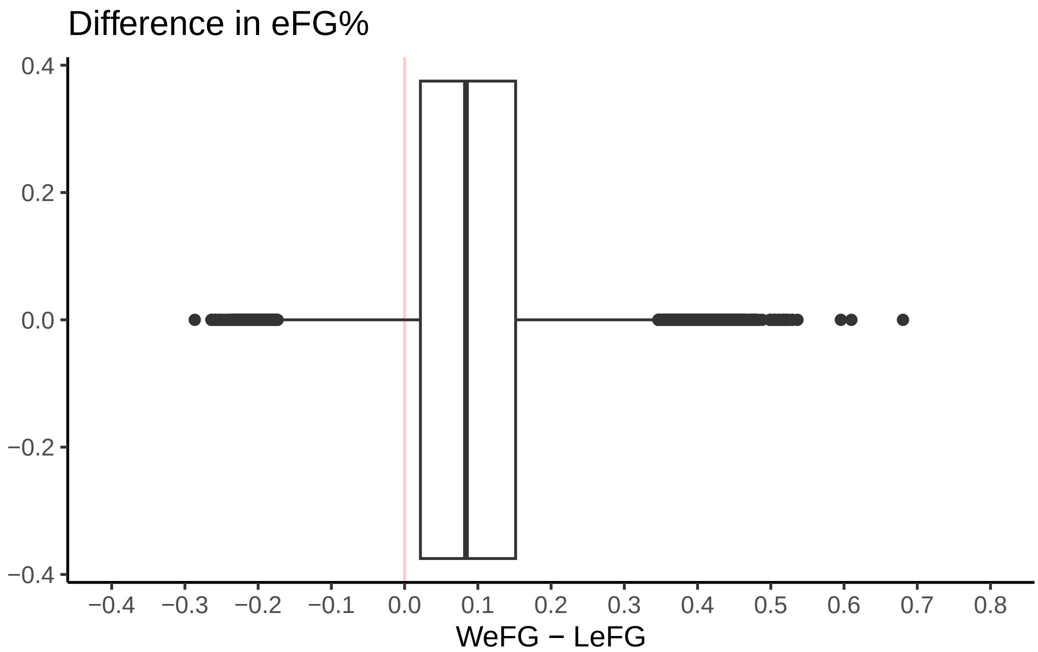
```
plt <- score_diff %>%
  ggplot(aes(x = difference)) +
  geom_boxplot() +
  ggtitle("Distribution of Score Difference between Winning and Losing Teams")

plt +
  theme(plot.title = element_text(hjust = 0.5)) +
  xlab('Difference in Score') + scale_x_continuous(n.breaks = 10)
```




```
wscore <- wbb %>%
  ggplot(aes(x = WeFG)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(0,1))
lscore <- wbb %>%
  ggplot(aes(x = LeFG)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(0,1))

plt <- ggarrange(wscore, lscore, ncol = 1, nrow = 2)
annotate_figure(plt, top = text_grob("eFG% Distribution for Winning and Losing Teams", face =
```




```
# effective field goal % difference
wbb %>%
  ggplot(aes(x = WeFG - LeFG)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(-0.4,0.8)) +
  geom_vline(xintercept = 0, alpha = 0.2, color = 'red') +
  ggtitle("Difference in eFG%")
```

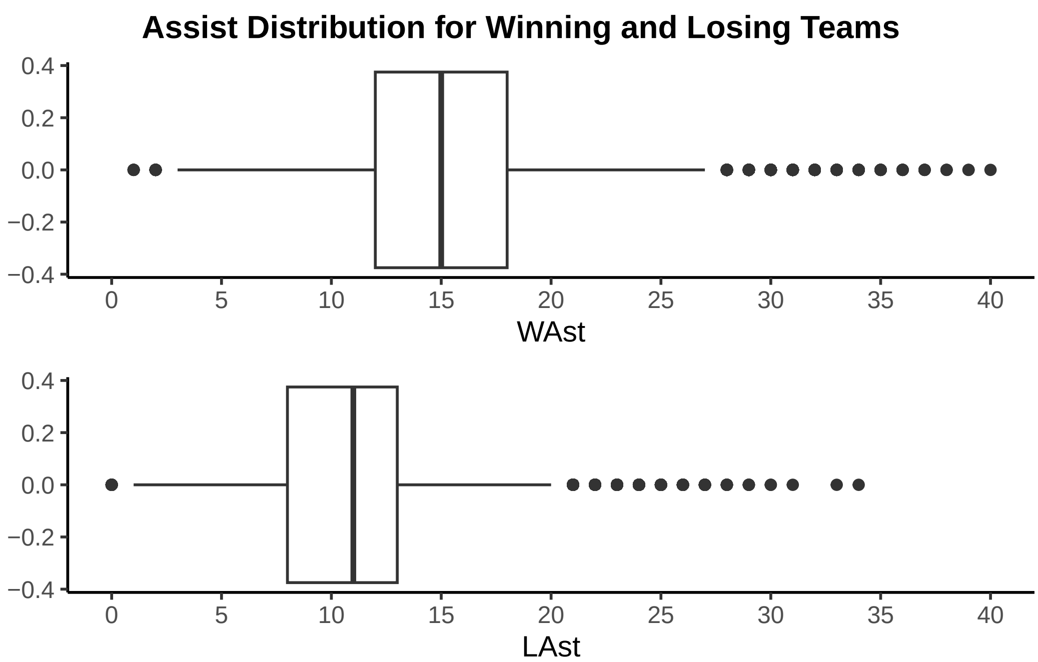



```
median(wbb$WeFG - wbb$LeFG)
```

```
[1] 0.08376271
```

**Assists, Rebounds, and Turnovers**

```
# assists
w <- wbb %>%
  ggplot(aes(x = WAst)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(0,40))
l <- wbb %>%
  ggplot(aes(x = LAst)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(0,40))

plt <- ggarrange(w, l, ncol = 1, nrow = 2)
annotate_figure(plt, top = text_grob("Assist Distribution for Winning and Losing Teams", face
```



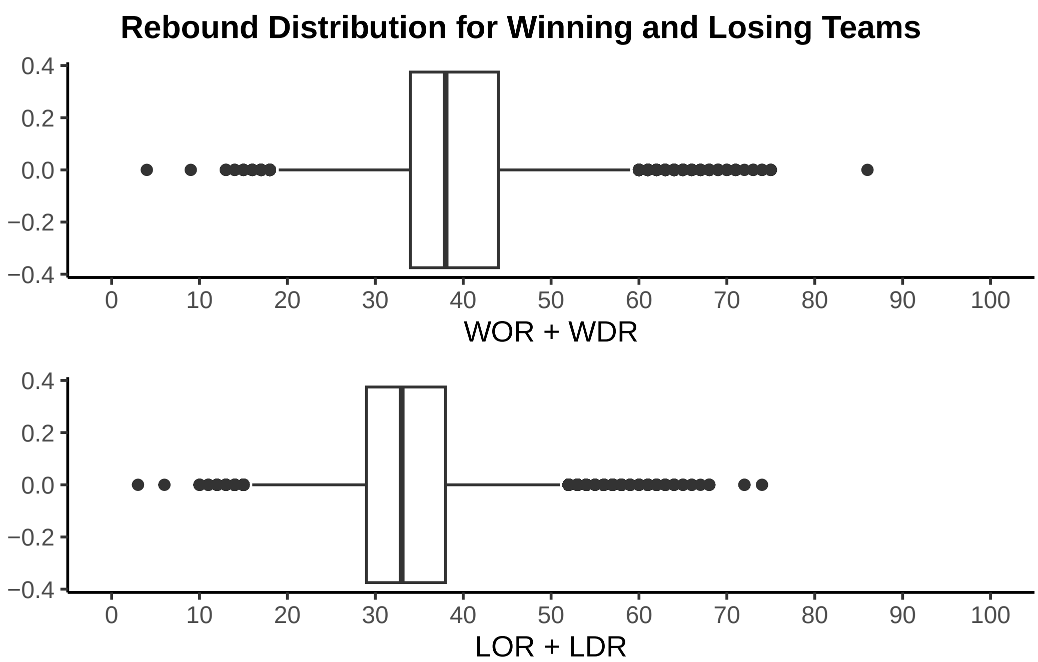

```
# rebounds
mean(wbb$WOR + wbb$WDR)
```

```
[1] 38.73694
```

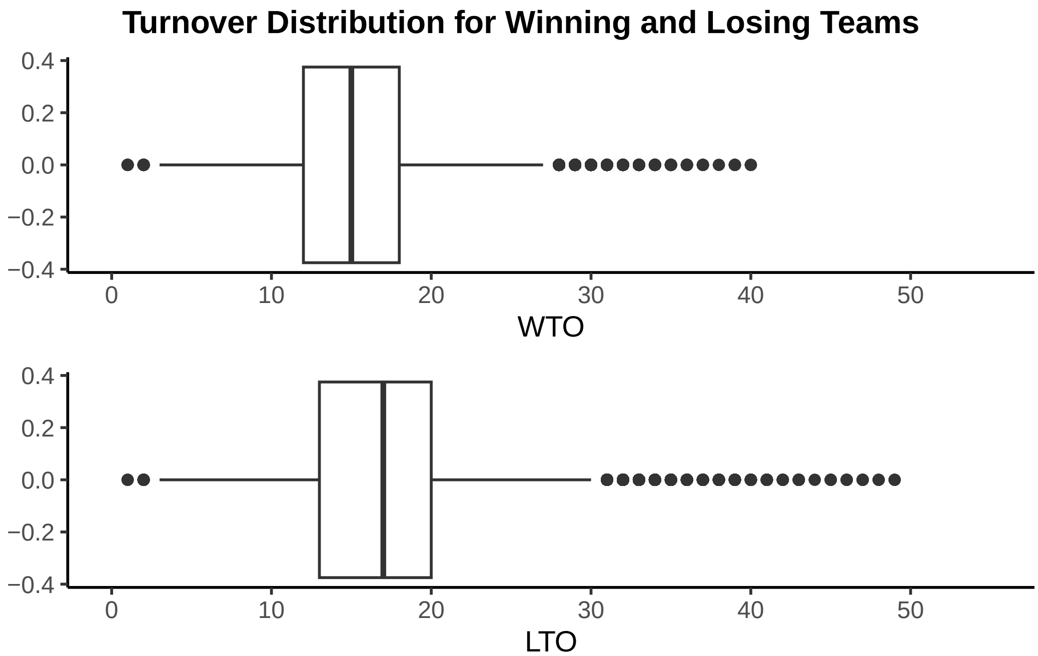
```
mean(wbb$LOR + wbb$LDR)
```

```
[1] 33.74551
```

```
w <- wbb %>%
  ggplot(aes(x = WOR + WDR)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(0,100))
l <- wbb %>%
  ggplot(aes(x = LOR + LDR)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(0,100))

plt <- ggarrange(w, l, ncol = 1, nrow = 2)
annotate_figure(plt, top = text_grob("Rebound Distribution for Winning and Losing Teams", fac
```




```
# turnovers
w <- wbb %>%
  ggplot(aes(x = WTO)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(0,55))
l <- wbb %>%
  ggplot(aes(x = LTO)) +
  geom_boxplot() +
  theme_classic() +
  scale_x_continuous(n.breaks = 10, limits = c(0,55))

plt <- ggarrange(w, l, ncol = 1, nrow = 2)
annotate_figure(plt, top = text_grob("Turnover Distribution for Winning and Losing Teams", fa
```




## 410 Project - Models

```
wbb <- read.csv("C:/Users/gigip/personal/personal projects/march madness/WRegularSeasonDetai

# make effective field goal percentage variable
wbb <- wbb %>%
  mutate(WeFG = 100*(WFGM + 0.5 * WFGM3) / WFGA,
         LeFG = 100*(LFGM + 0.5 * LFGM3) / LFGA)
```

## Regression Analysis

### Data Prep

#### Add Binary Outcome Variable

- add variable that signifies whether the home team won the game
  - 1 if home team won
  - 0 if home team lost

– removed neutral games

```r
wbb$HomeWin <- ifelse(
  wbb$WLoc == "H", 1,        # Home team won
  ifelse(wbb$WLoc == "A", 0, NA)  # Home team lost (Away won)
)
wbb <- wbb %>%
  filter(WLoc != "N") # Remove neutral games
```

### Put Variables in Terms of Home/Away Teams

```r
# Initialize new columns for home/away stats
home_stats <- c("Score", "FGM", "FGA", "FGM3", "FGA3", "FTM", "FTA", "OR", "DR", "Ast", "TO"
away_stats <- paste0("Away_", home_stats)  # e.g., "Away_FGM"
home_stats <- paste0("Home_", home_stats)  # e.g., "Home_FGM"

# Loop through each stat and assign home/away values
for (i in seq_along(home_stats)) {
  stat <- gsub("Home_", "", home_stats[i])  # e.g., "FGM"

  # If home team won, W stats = home team, L stats = away team
  wbb[home_stats[i]] <- ifelse(
    wbb$WLoc == "H",
    wbb[[paste0("W", stat)]],  # e.g., WFGM
    wbb[[paste0("L", stat)]]   # e.g., LFGM (home team lost)
  )

  wbb[away_stats[i]] <- ifelse(
    wbb$WLoc == "H",
    wbb[[paste0("L", stat)]],  # e.g., LFGM
    wbb[[paste0("W", stat)]]   # e.g., WFGM
  )
}
```

### Create Additional Features

```r
# Field goal difference
wbb$FGM_diff <- wbb$Home_FGM - wbb$Away_FGM
```

```r
# Turnover difference (negative means home team had more TOs)
wbb$TO_diff <- wbb$Home_TO - wbb$Away_TO

# Home/Away FG% ratio
wbb$FG_pct_ratio <- (wbb$Home_FGM / wbb$Home_FGA) /
                    (wbb$Away_FGM / wbb$Away_FGA)

# Home/Away eFG% ratio
wbb$eFG_pct_ratio <- wbb$Home_eFG / wbb$Away_eFG
```

### Cleaning

```r
wbb_subset <- wbb %>% filter(Season == 2024 | Season == 2025) %>% dplyr::select(HomeWin:eFG_p

# add a few more variables
wbb_subset$Home_R <- wbb_subset$Home_OR + wbb_subset$Home_DR
wbb_subset$Away_R <- wbb_subset$Away_OR + wbb_subset$Away_DR
wbb_subset$HomeWin <- as.factor(wbb_subset$HomeWin)

head(wbb_subset)
```

```
  HomeWin Home_Score Away_Score Home_FGM Away_FGM Home_FGA Away_FGA Home_FGM3
1       1         65         63       26       23       64       57         4
2       1         93         39       29       13       66       43         4
3       0         55         58       21       15       53       44         4
4       1         81         68       30       25       64       64        11
5       1         71         65       24       27       66       62         2
6       0         57         68       19       29       55       59         7
  Away_FGM3 Home_FGA3 Away_FGA3 Home_FTM Away_FTM Home_FTA Away_FTA Home_OR
1         7        12        23        9       10       15       19      10
2         1        16        10       31       12       44       17      20
3         8        14        22        9       20       10       29       8
4         4        25        13       10       14       17       23       7
5         3        12        14       21        8       31       13      14
6         5        15        20       12        5       18       11       5
  Away_OR Home_DR Away_DR Home_Ast Away_Ast Home_TO Away_TO Home_Stl Away_Stl
1       7      25      27       14       10      14      15        9        6
2       5      26      18       14        7       9      26       18        3
3       7      16      18        8       10      15      18       11        5
4       5      33      28       24        9      20      16        7       15
```

```
5        7       26       23         7        10       16       13          8         7
6        7       19       22        12        13       23       26         12        10
  Home_Blk Away_Blk Home_eFG Away_eFG FGM_diff TO_diff FG_pct_ratio
1        5        3 43.75000 46.49123        3      -1    1.0067935
2        4        7 46.96970 31.39535       16     -17    1.4533800
3        4        4 43.39623 43.18182        6      -3    1.1622642
4        2        3 55.46875 42.18750        5       4    1.2000000
5        4        2 37.87879 45.96774       -3       3    0.8350168
6        0        7 40.90909 53.38983      -10      -3    0.7028213
  eFG_pct_ratio NumOT Home_R Away_R
1     0.9410377     0     35     34
2     1.4960718     0     46     23
3     1.0049652     0     24     25
4     1.3148148     0     40     33
5     0.8240298     0     40     30
6     0.7662338     0     24     29
```

#### Create a Test Dataset Using 2023 Season

```r
wbb_subset_test <- wbb %>% filter(Season == 2023) %>% dplyr::select(HomeWin:eFG_pct_ratio, Nu

# add a few more variables
wbb_subset_test$Home_R <- wbb_subset_test$Home_OR + wbb_subset_test$Home_DR
wbb_subset_test$Away_R <- wbb_subset_test$Away_OR + wbb_subset_test$Away_DR
wbb_subset_test$HomeWin <- as.factor(wbb_subset_test$HomeWin)

head(wbb_subset_test)
```

```
  HomeWin Home_Score Away_Score Home_FGM Away_FGM Home_FGA Away_FGA Home_FGM3
1       0         63         67       20       22       57       61         5
2       1         98         51       31       16       58       64        13
3       1         69         68       26       26       72       73         4
4       0         50         70       17       23       50       58         4
5       1         88         50       33       18       68       50         9
6       1         81         53       33       16       63       47         3
  Away_FGM3 Home_FGA3 Away_FGA3 Home_FTM Away_FTM Home_FTA Away_FTA Home_OR
1         6        24        20       18       17       26       18      14
2         2        25        12       23       17       33       27      11
3         9        18        27       13        7       18       15      15
4         7        19        17       12       17       27       36       7
5         3        25        12       13       11       20       19      12
```

```
6          7         12         17         12         14         24         18        12
  Away_OR Home_DR Away_DR Home_Ast Away_Ast Home_TO Away_TO Home_Stl Away_Stl
1      10      21      25        8       16      17      11        7       11
2      14      35      21       18        5      12      13       10        3
3      13      28      27       11       17      12      15        9        9
4      11      25      20        7        3      14      13        2        6
5       5      26      23       22       11      10      19        6        4
6       3      26      15       20       10      22      26       18        9
  Home_Blk Away_Blk Home_eFG Away_eFG FGM_diff TO_diff FG_pct_ratio
1        5        4 39.47368 40.98361       -2       6    0.9728868
2        7        2 64.65517 26.56250       15      -1    2.1379310
3        5        7 38.88889 41.78082        0      -3    1.0138889
4        2        0 38.00000 45.68966       -6       1    0.8573913
5        5        5 55.14706 39.00000       15      -9    1.3480392
6        7        3 54.76190 41.48936       17      -4    1.5386905
  eFG_pct_ratio NumOT Home_R Away_R
1     0.9631579     0     35     35
2     2.4340771     0     46     35
3     0.9307832     0     43     40
4     0.8316981     0     32     31
5     1.4140271     0     38     28
6     1.3199023     0     38     18
```

### Logistic Regression

Max model with all variables

- do not include unique identifiers or non-statistical variables (season, day number, team ID)
- since the location was used to create the response variable, do not include this either
  - more interested in how stats impact win vs lose because teams will have to play at home and away no matter what

### Test Model

```r
mod <- glm(HomeWin ~ Home_eFG + Away_eFG + TO_diff, family = binomial(), data = wbb_subset)

summary(mod)
```

```
Call:
glm(formula = HomeWin ~ Home_eFG + Away_eFG + TO_diff, family = binomial(),
    data = wbb_subset)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.079243   0.280036  -0.283    0.777
Home_eFG     0.306576   0.007489  40.936   <2e-16 ***
Away_eFG    -0.298991   0.007386 -40.480   <2e-16 ***
TO_diff     -0.298823   0.008677 -34.440   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 12973.1  on 9672  degrees of freedom
Residual deviance:  4937.2  on 9669  degrees of freedom
AIC: 4945.2

Number of Fisher Scoring iterations: 7
```

```r
vif(mod)
```

```
Home_eFG Away_eFG  TO_diff
1.999204 2.046611 1.753894
```

### Backward Selection

```r
mod_max <- glm(HomeWin ~ . - Home_Score - Away_Score, data = wbb_subset, family = binomial()
mod_back <- step(mod_max, direction = "backward")
```

```r
summary(mod_back)
```

```
Call:
glm(formula = HomeWin ~ Home_FGM + Away_FGM + Home_FGM3 + Away_FGM3 +
    Home_FTM + Away_FTM, family = binomial(), data = wbb_subset)

Coefficients:
```

```
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    0.5813  2448.0144   0.000    1.000
Home_FGM      36.5272   767.1172   0.048    0.962
Away_FGM     -36.5330   767.7281  -0.048    0.962
Home_FGM3     18.2680   408.9150   0.045    0.964
Away_FGM3    -18.3245   409.5950  -0.045    0.964
Home_FTM      18.2715   388.0147   0.047    0.962
Away_FTM     -18.2743   387.7151  -0.047    0.962

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1.2973e+04  on 9672  degrees of freedom
Residual deviance: 5.0436e-06  on 9666  degrees of freedom
AIC: 14

Number of Fisher Scoring iterations: 25
```

```r
vif(mod_back)
```

```
 Home_FGM  Away_FGM Home_FGM3 Away_FGM3  Home_FTM  Away_FTM
75.473004 72.862053  7.756750  8.221323 24.619259 26.215169
```

```r
adj_mod <- glm(HomeWin ~ Home_FGM + Home_FGM3 + Away_FGM3 + Home_FTM + Away_FTM, wbb_subset,
summary(adj_mod)
```

```
Call:
glm(formula = HomeWin ~ Home_FGM + Home_FGM3 + Away_FGM3 + Home_FTM +
    Away_FTM, family = binomial(), data = wbb_subset)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -7.432214   0.232474  -31.97   <2e-16 ***
Home_FGM     0.416964   0.010329   40.37   <2e-16 ***
Home_FGM3    0.197173   0.012658   15.58   <2e-16 ***
Away_FGM3   -0.491642   0.014394  -34.16   <2e-16 ***
Home_FTM     0.258691   0.007671   33.72   <2e-16 ***
Away_FTM    -0.299051   0.008181  -36.55   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 12973.1  on 9672  degrees of freedom
Residual deviance:  6063.1  on 9667  degrees of freedom
AIC: 6075.1

Number of Fisher Scoring iterations: 6
```

```r
vif(adj_mod)
```

```
 Home_FGM Home_FGM3 Away_FGM3  Home_FTM  Away_FTM
 1.656165  1.157867  1.524349  1.471788  1.625096
```

### Accuracy of Backwards Selection

```r
# confusion matrix using 24-25 data
actual <- as.factor(wbb_subset$HomeWin)
predict_probs <- predict(adj_mod, type = "response")
predicted <- as.factor(ifelse(predict_probs > 0.5, 1, 0))

conf_mat <- confusionMatrix(actual, predicted)
conf_mat
```

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 3080  733
         1  596 5264

               Accuracy : 0.8626
                 95% CI : (0.8556, 0.8694)
    No Information Rate : 0.62
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7105

 Mcnemar's Test P-Value : 0.000191

            Sensitivity : 0.8379
```

```
            Specificity : 0.8778
         Pos Pred Value : 0.8078
         Neg Pred Value : 0.8983
             Prevalence : 0.3800
         Detection Rate : 0.3184
   Detection Prevalence : 0.3942
      Balanced Accuracy : 0.8578

       'Positive' Class : 0
```

### Accuracy for Backwards Selection on Test Dataset

```r
actual <- as.factor(wbb_subset_test$HomeWin)
predict_probs <- predict(adj_mod, newdata = wbb_subset_test, type = "response")
predicted <- as.factor(ifelse(predict_probs > 0.5, 1, 0))

conf_mat <- confusionMatrix(actual, predicted)
conf_mat
```

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1512  422
         1  349 2682

               Accuracy : 0.8447
                 95% CI : (0.8343, 0.8547)
    No Information Rate : 0.6252
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6712

 Mcnemar's Test P-Value : 0.009514

            Sensitivity : 0.8125
            Specificity : 0.8640
         Pos Pred Value : 0.7818
         Neg Pred Value : 0.8849
             Prevalence : 0.3748
```

```
         Detection Rate : 0.3045
   Detection Prevalence : 0.3895
      Balanced Accuracy : 0.8383

       'Positive' Class : 0
```

### Variable Selection Using Subset for Max Model

```r
mod_max <- glm(HomeWin ~ Home_eFG + Away_eFG + Home_R + Away_R + Home_Ast + Away_Ast + TO_di
               family = binomial(),
               data = wbb_subset)
summary(mod_max)
```

```
Call:
glm(formula = HomeWin ~ Home_eFG + Away_eFG + Home_R + Away_R +
    Home_Ast + Away_Ast + TO_diff + FGM_diff + eFG_pct_ratio,
    family = binomial(), data = wbb_subset)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -8.96295    1.94838  -4.600 4.22e-06 ***
Home_eFG       0.26694    0.04038   6.611 3.82e-11 ***
Away_eFG      -0.26397    0.03983  -6.627 3.42e-11 ***
Home_R         0.32620    0.01262  25.837  < 2e-16 ***
Away_R        -0.33976    0.01313 -25.884  < 2e-16 ***
Home_Ast       0.01575    0.01571   1.003    0.316
Away_Ast      -0.01997    0.01528  -1.307    0.191
TO_diff       -0.70328    0.02181 -32.251  < 2e-16 ***
FGM_diff      -0.18233    0.01962  -9.295  < 2e-16 ***
eFG_pct_ratio  9.34091    1.80045   5.188 2.12e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 12973.1  on 9672  degrees of freedom
Residual deviance:  2693.1  on 9663  degrees of freedom
AIC: 2713.1
```

```
Number of Fisher Scoring iterations: 9
```

```r
# stepwise selection
mod <- step(mod_max, direction = "both")
```

```
Start:  AIC=2713.07
HomeWin ~ Home_eFG + Away_eFG + Home_R + Away_R + Home_Ast +
    Away_Ast + TO_diff + FGM_diff + eFG_pct_ratio

                Df Deviance    AIC
- Home_Ast       1   2694.1 2712.1
- Away_Ast       1   2694.8 2712.8
<none>               2693.1 2713.1
- eFG_pct_ratio  1   2722.7 2740.7
- Home_eFG       1   2737.5 2755.5
- Away_eFG       1   2738.6 2756.6
- FGM_diff       1   2784.0 2802.0
- Away_R         1   3778.3 3796.3
- Home_R         1   3785.4 3803.4
- TO_diff        1   5267.0 5285.0


Step:  AIC=2712.07
HomeWin ~ Home_eFG + Away_eFG + Home_R + Away_R + Away_Ast +
    TO_diff + FGM_diff + eFG_pct_ratio

                Df Deviance    AIC
- Away_Ast       1   2695.4 2711.4
<none>               2694.1 2712.1
+ Home_Ast       1   2693.1 2713.1
- eFG_pct_ratio  1   2723.6 2739.6
- Away_eFG       1   2739.3 2755.3
- Home_eFG       1   2740.7 2756.7
- FGM_diff       1   2784.0 2800.0
- Away_R         1   3787.0 3803.0
- Home_R         1   3860.1 3876.1
- TO_diff        1   5279.7 5295.7
```

```
Step:  AIC=2711.41
HomeWin ~ Home_eFG + Away_eFG + Home_R + Away_R + TO_diff + FGM_diff +
    eFG_pct_ratio

                Df Deviance    AIC
<none>               2695.4 2711.4
+ Away_Ast       1   2694.1 2712.1
+ Home_Ast       1   2694.8 2712.8
- eFG_pct_ratio  1   2725.8 2739.8
- Home_eFG       1   2741.1 2755.1
- Away_eFG       1   2742.1 2756.1
- FGM_diff       1   2784.6 2798.6
- Away_R         1   3858.5 3872.5
- Home_R         1   3872.8 3886.8
- TO_diff        1   5293.6 5307.6
```

```r
summary(mod)
```

```
Call:
glm(formula = HomeWin ~ Home_eFG + Away_eFG + Home_R + Away_R +
    TO_diff + FGM_diff + eFG_pct_ratio, family = binomial(),
    data = wbb_subset)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -9.02258    1.93689  -4.658 3.19e-06 ***
Home_eFG       0.26728    0.03998   6.685 2.31e-11 ***
Away_eFG      -0.26588    0.03973  -6.693 2.19e-11 ***
Home_R         0.32696    0.01239  26.387  < 2e-16 ***
Away_R        -0.34066    0.01286 -26.500  < 2e-16 ***
TO_diff       -0.70412    0.02179 -32.319  < 2e-16 ***
FGM_diff      -0.17876    0.01941  -9.210  < 2e-16 ***
eFG_pct_ratio  9.43351    1.79868   5.245 1.57e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 12973.1  on 9672  degrees of freedom
Residual deviance:  2695.4  on 9665  degrees of freedom
AIC: 2711.4
```

```
Number of Fisher Scoring iterations: 9
```

```r
vif(mod)
```

```
     Home_eFG      Away_eFG        Home_R        Away_R       TO_diff 
    31.705973     31.181532      2.391160      2.645752      5.880335 
     FGM_diff eFG_pct_ratio 
     1.481757     32.775609 
```

```r
# adjust for vif
mod_adj <- glm(HomeWin ~ Home_eFG + Home_R + Away_R + TO_diff + FGM_diff,
               wbb_subset,
               family = binomial)
summary(mod_adj)
```

```
Call:
glm(formula = HomeWin ~ Home_eFG + Home_R + Away_R + TO_diff + 
    FGM_diff, family = binomial, data = wbb_subset)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept) -12.240853   0.534123  -22.92   <2e-16 ***
Home_eFG      0.179347   0.006885   26.05   <2e-16 ***
Home_R        0.219732   0.008043   27.32   <2e-16 ***
Away_R       -0.087063   0.006677  -13.04   <2e-16 ***
TO_diff      -0.239368   0.009207  -26.00   <2e-16 ***
FGM_diff      0.259613   0.011766   22.06   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 12973.1  on 9672  degrees of freedom
Residual deviance:  4987.4  on 9667  degrees of freedom
AIC: 4999.4

Number of Fisher Scoring iterations: 7
```

```r
vif(mod_adj)
```

```
Home_eFG   Home_R   Away_R  TO_diff FGM_diff 
1.778007 1.905363 1.317946 2.013816 1.066972 
```

### Accuracy for Subset Max Model

```r
# confusion matrix using 24-25 data
actual <- as.factor(wbb_subset$HomeWin)
predict_probs <- predict(mod_adj, type = "response")
predicted <- as.factor(ifelse(predict_probs > 0.5, 1, 0))

conf_mat <- confusionMatrix(actual, predicted)
conf_mat
```

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 3207  606
         1  529 5331
                                          
               Accuracy : 0.8827          
                 95% CI : (0.8761, 0.889) 
    No Information Rate : 0.6138          
    P-Value [Acc > NIR] : < 2e-16         
                                          
                  Kappa : 0.7535          
                                          
 Mcnemar's Test P-Value : 0.02408         
                                          
            Sensitivity : 0.8584          
            Specificity : 0.8979          
         Pos Pred Value : 0.8411          
         Neg Pred Value : 0.9097          
             Prevalence : 0.3862          
         Detection Rate : 0.3315          
   Detection Prevalence : 0.3942          
      Balanced Accuracy : 0.8782          
```

```
       'Positive' Class : 0
```

#### Accuracy for Stepwise Model on Test Dataset

```r
actual <- as.factor(wbb_subset_test$HomeWin)
predict_probs <- predict(mod_adj, newdata = wbb_subset_test, type = "response")
predicted <- as.factor(ifelse(predict_probs > 0.5, 1, 0))

conf_mat <- confusionMatrix(actual, predicted)
conf_mat
```

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1640  294
         1  290 2741

               Accuracy : 0.8824
                 95% CI : (0.8731, 0.8912)
    No Information Rate : 0.6113
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.7526

 Mcnemar's Test P-Value : 0.9012

            Sensitivity : 0.8497
            Specificity : 0.9031
         Pos Pred Value : 0.8480
         Neg Pred Value : 0.9043
             Prevalence : 0.3887
         Detection Rate : 0.3303
   Detection Prevalence : 0.3895
      Balanced Accuracy : 0.8764

       'Positive' Class : 0
```

### ROC Curve

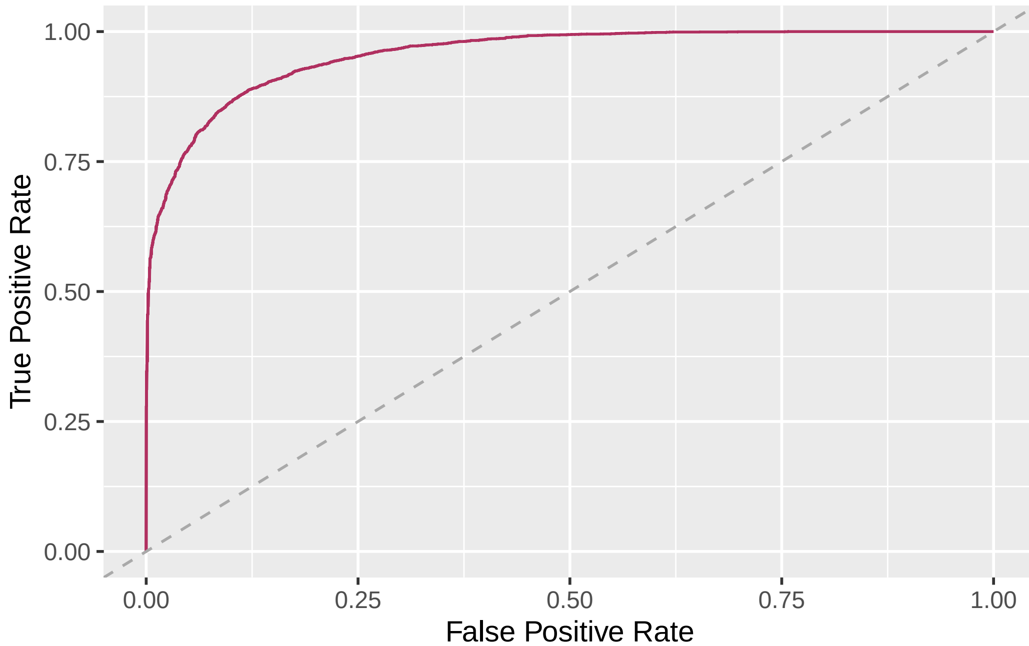
```
library(pROC)
```

```
Type 'citation("pROC")' for a citation.


Attaching package: 'pROC'

The following objects are masked from 'package:stats':

    cov, smooth, var
```

```
# get the predicted probabilities (pi-hats)
pi_hat <- predict(mod_adj, type = "response") # makes predictions off every observation in th

# roc() has two inputs: the actual response and the pi-hats
roc_obj <- roc(wbb_subset$HomeWin, pi_hat)
```

```
Setting levels: control = 0, case = 1

Setting direction: controls < cases
```

```
auc_value <- auc(roc_obj)

data.frame(fpr = 1 - roc_obj$specificities, tpr = roc_obj$sensitivities) %>%
  ggplot(aes(x = fpr, y = tpr)) +
  geom_line(color="maroon") +
  geom_abline(slope = 1, intercept = 0, linetype = "dashed", color = "darkgrey") +
  xlab("False Positive Rate") +
  ylab("True Positive Rate")
```

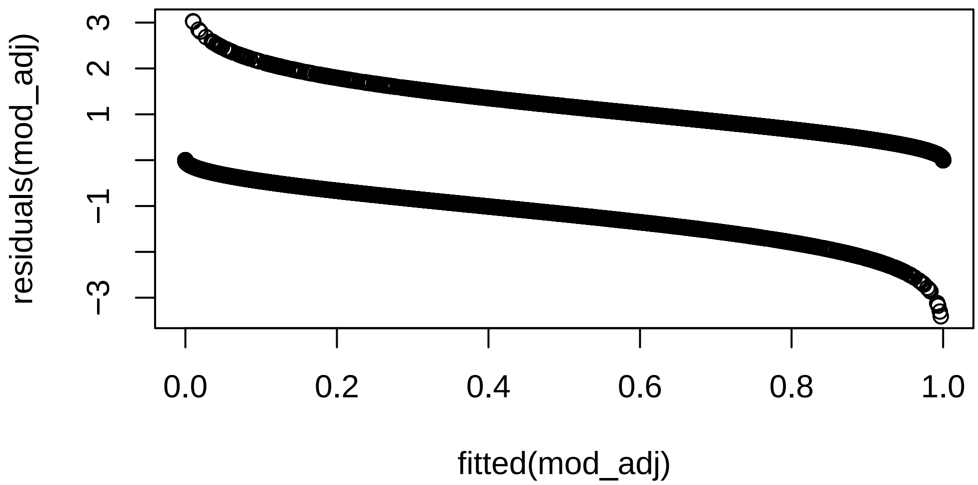



```r
auc_value
```

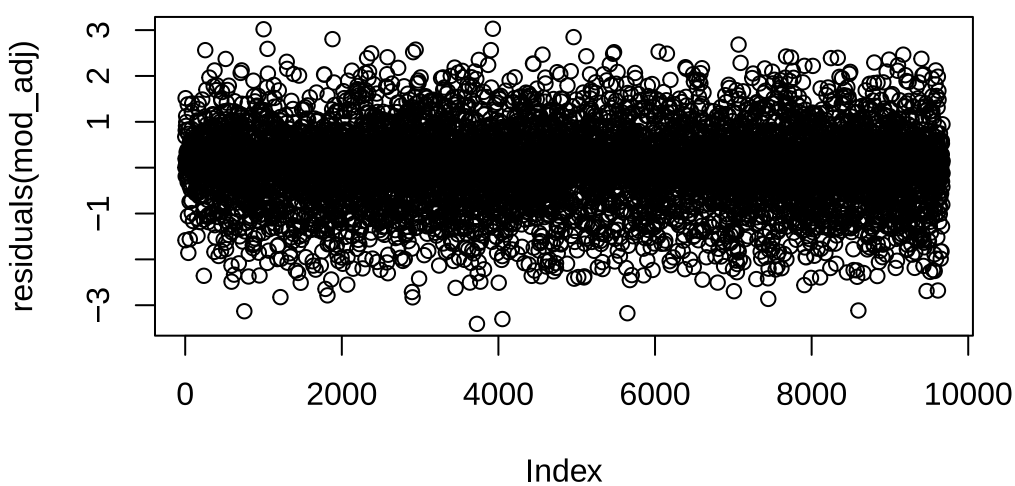
```
Area under the curve: 0.9567
```

### Check Deviance

```r
plot(fitted(mod_adj), residuals(mod_adj))
```




```r
plot(residuals(mod_adj))
```



### Compare With Random Forest

Compare best logistic regression model with a Random Forest to show trade-offs between simplicity and accuracy.

```r
rf_model <- randomForest(
  as.factor(HomeWin) ~ Home_eFG + Home_R + Away_R + TO_diff + FGM_diff,
  data = wbb_subset,
  importance = TRUE  # Shows variable importance
)
print(importance(rf_model))
```

```
                 0        1 MeanDecreaseAccuracy MeanDecreaseGini
Home_eFG  97.47402 64.28292             109.6919        1092.5077
Home_R   100.87496 86.22211             125.0437         638.4869
Away_R    43.18682 15.69876              43.8438         452.2356
TO_diff   84.45318 67.46901             102.2243         538.7227
FGM_diff  93.98068 74.76885             116.2026        1876.9997
```

### Random Forest Accuracy on Test Dataset

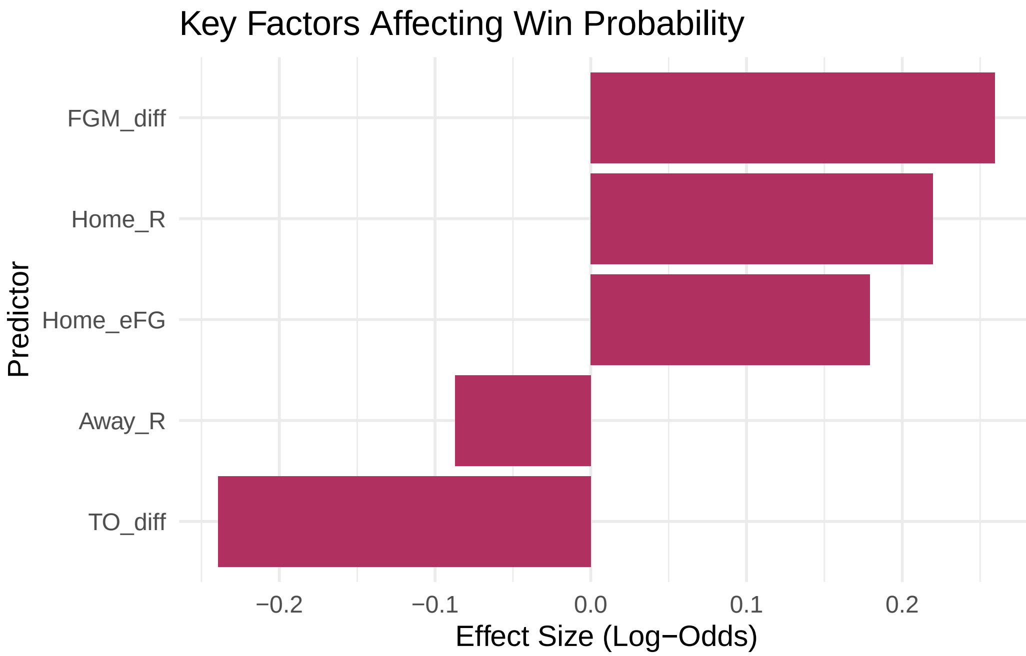
```r
predict_probs <- predict(rf_model, newdata = wbb_subset_test)

conf_matrix_rf <- confusionMatrix(predict_probs, wbb_subset_test$HomeWin)
conf_matrix_rf
```

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1606  314
         1  328 2717

               Accuracy : 0.8707
                 95% CI : (0.861, 0.8799)
    No Information Rate : 0.6105
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.7278

 Mcnemar's Test P-Value : 0.6079
```

```
            Sensitivity : 0.8304
            Specificity : 0.8964
         Pos Pred Value : 0.8365
         Neg Pred Value : 0.8923
             Prevalence : 0.3895
         Detection Rate : 0.3235
   Detection Prevalence : 0.3867
      Balanced Accuracy : 0.8634

       'Positive' Class : 0
```

## Coefficient Plot

```
library(ggplot2)
coef_df <- data.frame(
  Predictor = names(coef(mod_adj)),
  Effect = coef(mod_adj)) %>%
  filter(Predictor != '(Intercept)')

ggplot(coef_df, aes(x = Effect, y = reorder(Predictor, Effect))) +
  geom_col(fill = "maroon") +
  labs(title = "Key Factors Affecting Win Probability",
       y = "Predictor", x = "Effect Size (Log-Odds)") +
  theme_minimal()
```




Bars show which stats most influence win margins (longer = stronger effect).