Analyzing the Association between Dallas Cowboys Offensive and Defensive Season Rankings and Margin of Victory

Rithvik Saravanan

November 20, 2020

Data

```
# load data
data <- read.csv('./dallas_cowboys_season_data.csv')
mydata <- data[c("MoV", "PtsScoredRank", "PtsAllowedRank",
                 "YdsGainedRank", "YdsAllowedRank")]
```

Research Questions

- If the Dallas Cowboys are ranked first in offensive and defensive season rankings in yards and points in a future season, what is a range of their predicted margin of victory?
- What is the range of the mean predicted margin of victory for a future season where the Dallas Cowboys are ranked last in offensive and defensive season rankings in yards and points?
- Is there a relationship between points scored and yards gained for the Dallas Cowboys?
- Is there a relationship between points allowed and yards allowed for the Dallas Cowboys?

To answer these research questions, I plan to fit a multiple linear regression model where the predictors of offensive and defensive season rankings in yards and points will be used to predict the margin of victory. I plan to build this model by comparing the results and fit of both forward and backward stepwise selection as well as best subsets regression. I also plan to investigate whether there are relationships between any of the predictors, and if so, I will consider adding interaction effects to the model.

Model

To first analyze the relationships between the predictors and between the predictors and response, we can create a base multiple linear regression model with all four first-order predictors (points scored rank, points allowed rank, yards gained rank, yards allowed rank).

```
# multiple regression model with all predictors
reg1 <- lm(MoV ~ PtsScoredRank + PtsAllowedRank + YdsGainedRank + YdsAllowedRank, mydata)
summary(reg1)
```

```
##
## Call:
## lm(formula = MoV ~ PtsScoredRank + PtsAllowedRank + YdsGainedRank +
##     YdsAllowedRank, data = mydata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.1024  -0.3633   0.5633   1.8565   6.6565
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    12.386533   1.079058  11.479 3.17e-16 ***
## PtsScoredRank  -0.476896   0.119479  -3.991 0.000196 ***
## PtsAllowedRank -0.242511   0.100176  -2.421 0.018812 *
## YdsGainedRank  -0.004942   0.128988  -0.038 0.969575
## YdsAllowedRank -0.144042   0.104504  -1.378 0.173678
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.942 on 55 degrees of freedom
## Multiple R-squared:  0.6848, Adjusted R-squared:  0.6619
## F-statistic: 29.88 on 4 and 55 DF,  p-value: 3.212e-13
```

The regression equation of this relationship is modeled by

$$y = 12.3865 - 0.4769x_1 - 0.2425x_2 - 0.0049x_3 - 0.1440x_4$$

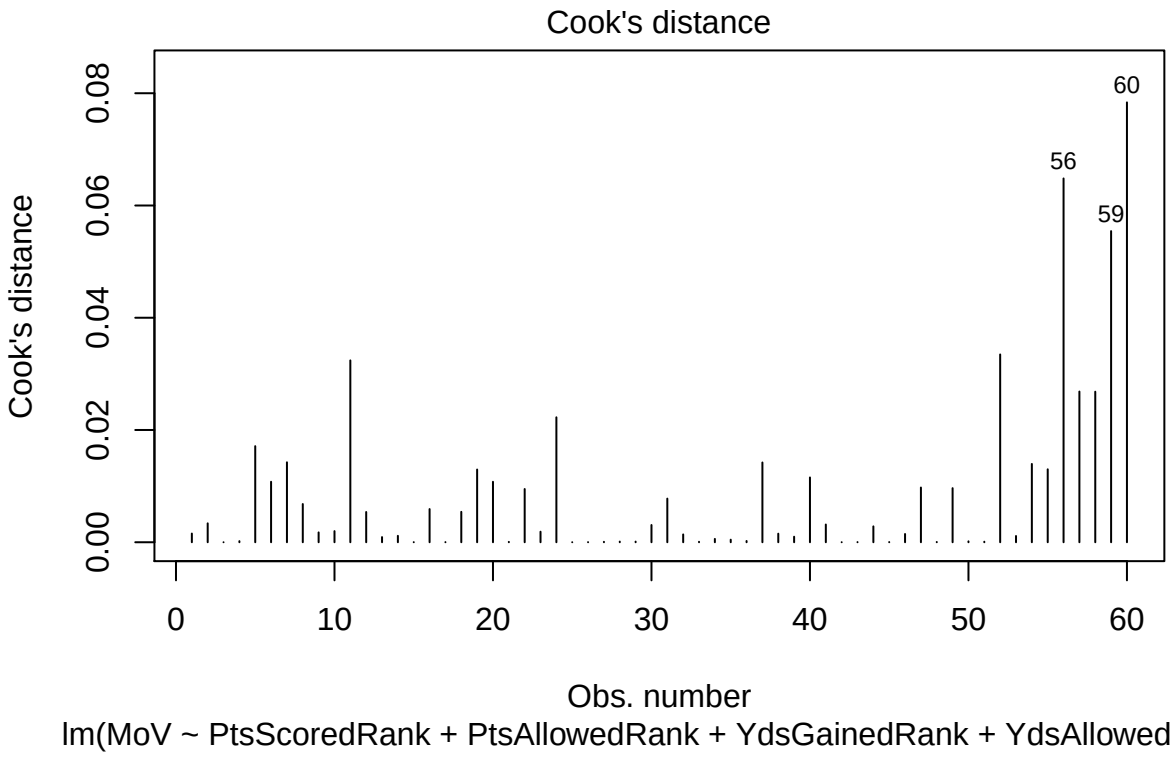
where y represents the margin of victory, x_1 represents the season rank in points scored, x_2 represents the season rank in points allowed, x_3 represents the season rank in yards gained, and x_4 represents the season rank in yards allowed..

From this base regression model, we notice that the R^2 is 0.6848 and the R^2_{adj} is 0.6619. The adjusted coefficient of determination of 0.6619 indicates that 66.19% of the variation in the response variable of `MoV` is accounted for by the predictor variables. In other words, when these predictor variables are considered, the variation in `MoV` is reduced by 66.19%.

Additionally, we observe that the p-values for `YdsGainedRank` and `YdsAllowedRank` are above the 0.05 significance level, while the p-values for `PtsScoredRank` and `PtsAllowedRank` are below the 0.05 significance level. This indicates that the slope coefficients for `YdsGainedRank` and `YdsAllowedRank` are not significantly different from zero while the slope coefficients for `PtsScoredRank` and `PtsAllowedRank` are indeed significantly different from zero.

To help build a better model, we can first remove any outliers using Cook's distance.

```
# identifying outliers with Cook's distance
plot(reg1, which = 4, cook.levels = cutoff)
```




```
mydata <- mydata[-56, -59, -60]
reg1 <- lm(MoV ~ PtsScoredRank + PtsAllowedRank + YdsGainedRank + YdsAllowedRank, mydata)
```

After removing outliers, we can verify the VIF values for each predictor to detect multicollinearity.

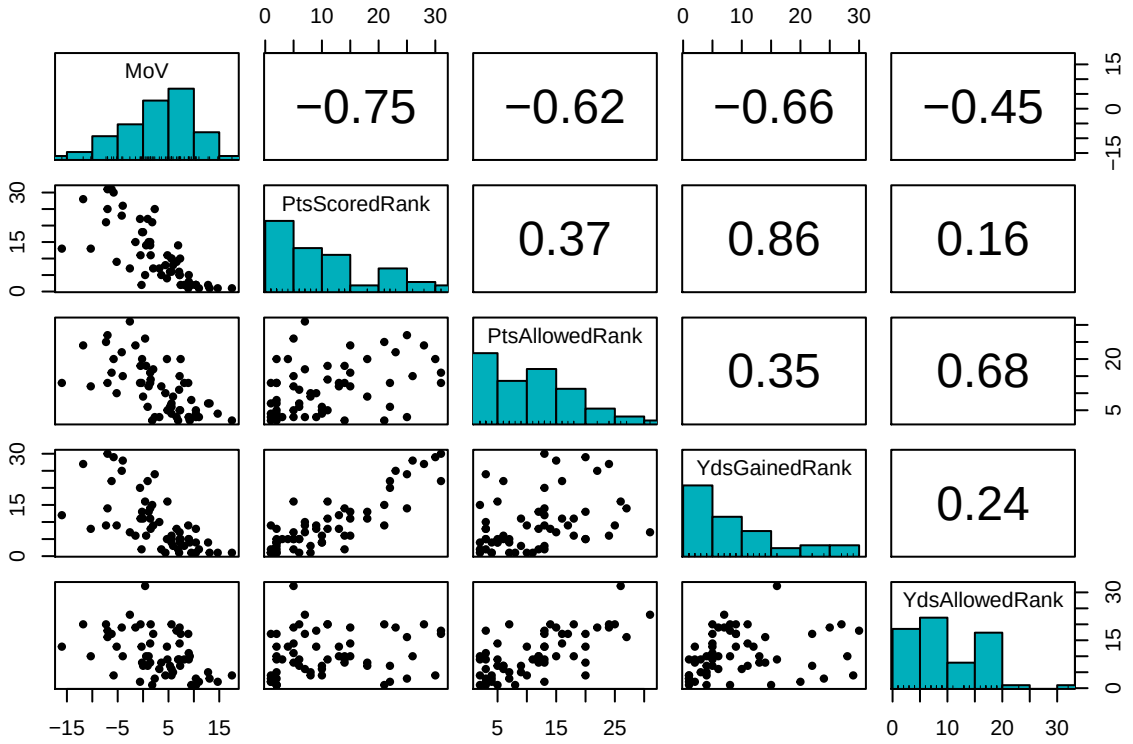
```
vif(reg1)
```

```
##  PtsScoredRank PtsAllowedRank  YdsGainedRank YdsAllowedRank
##       4.181204       2.140909       4.032055       1.958268
```

We note that the VIF for `PtsScoredRank` and `YdsGainedRank` is close to 5. This indicates that there may be some multicollinearity and we should proceed cautiously.

Next, we can plot a correlation matrix to observe the pairwise relationships.

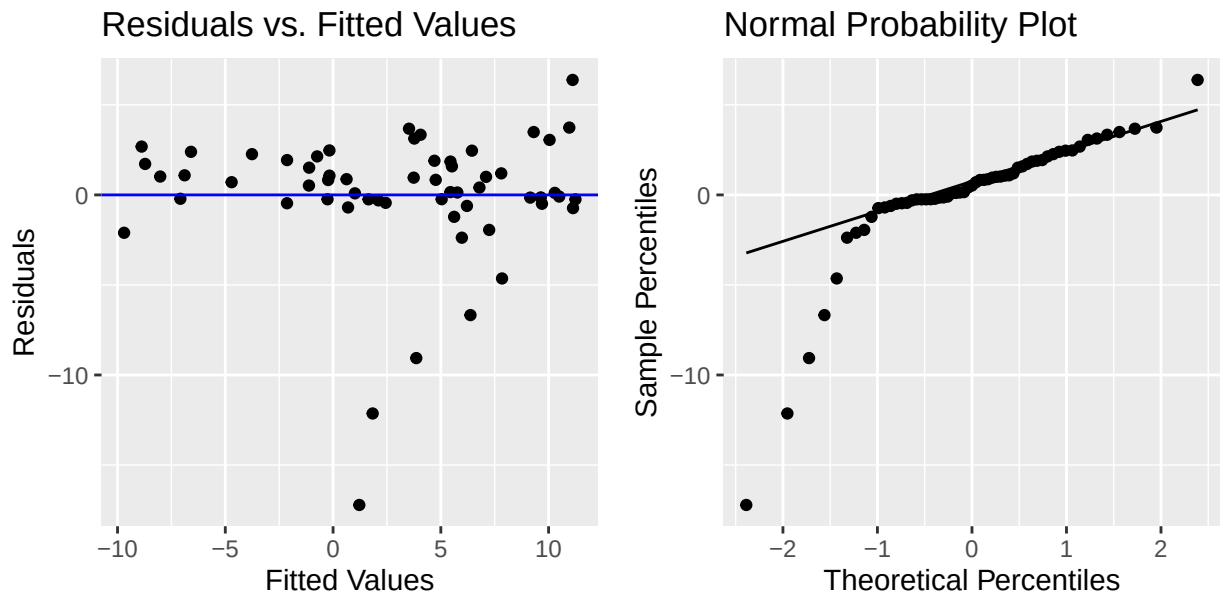
```
# scatter plot matrix
pairs.panels(mydata,
             method = "pearson", # correlation method
             hist.col = "#00AFBB", # color of histogram
             smooth = FALSE,
             density = FALSE,
             ellipses = FALSE)
```



From this correlation matrix, we notice that all of the predictors show a negative, linear, moderate to strong correlation with the response. We also observe that `PtsScoredRank` and `YdsGainedRank` as well as `PtsAllowedRank` and `YdsAllowedRank` have positive, linear, strong correlations. This indicates that interaction effects between these pairs of predictors will be useful in improving the model. We also understand that logarithmic transformations are not feasible for this model because the response variable `MoV` can be negative.

We can then check the diagnostics of this model to verify the assumptions.

```
# residuals vs. fitted values
mydata$resids1 <- residuals(reg1)
mydata$predicted1 <- predict(reg1)
ggplot(mydata, aes(x = predicted1, y = resids1)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "blue") +
  labs(title ="Residuals vs. Fitted Values",
       x = "Fitted Values",
       y = "Residuals")

# normal probability plot
ggplot(mydata, aes(sample = resids1)) +
  stat_qq() +
  stat_qq_line() +
  labs(title = "Normal Probability Plot",
       x = "Theoretical Percentiles",
       y = "Sample Percentiles")
```



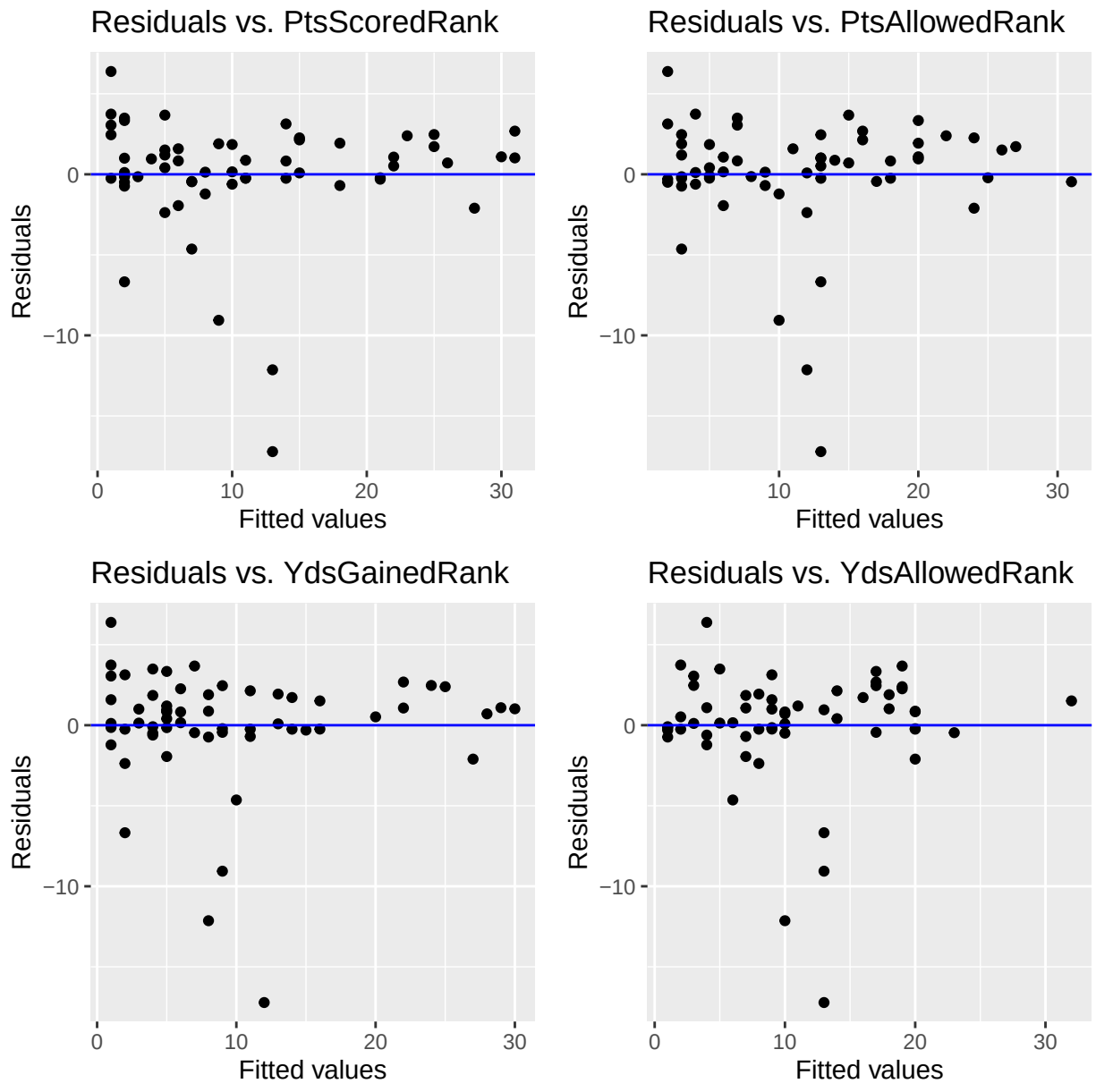
From this residual plot, we observe that there is approximately equal variance because the residuals are approximately equal in average magnitude for the fitted values across the plot. Excluding the few outliers near the bottom of the plot, we note that there is no obvious pattern since the fitted values are scattered mostly randomly across the plot. Therefore, this plot indicates linearity.

From this normal probability plot, we can observe that the data points form an approximately straight line and line up mostly along the line shown in the plot. This indicates that the normal distribution is a good model because the plot shows no significant deviation from a normal distribution of error terms. Since there is deviation from the line around the extremes (specifically at the lower extreme), this plot indicates heavy tails. Otherwise, the assumption that the residuals are normally distributed is approximately met.

We can also look at the residual plots for each predictor to understand whether a non-linear or higher-order model is viable.

```
# residuals vs. predictors
ggplot(mydata, aes(x = PtsScoredRank, y = resids1)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "blue") +
  labs(title = "Residuals vs. PtsScoredRank",
  x = "Fitted values",
  y = "Residuals")

ggplot(mydata, aes(x = PtsAllowedRank, y = resids1)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "blue") +
  labs(title = "Residuals vs. PtsAllowedRank",
  x = "Fitted values",
  y = "Residuals")

ggplot(mydata, aes(x = YdsGainedRank, y = resids1)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "blue") +
  labs(title = "Residuals vs. YdsGainedRank",
  x = "Fitted values",
```

```
  y = "Residuals")

ggplot(mydata, aes(x = YdsAllowedRank, y = resids1)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "blue") +
  labs(title = "Residuals vs. YdsAllowedRank",
  x = "Fitted values",
  y = "Residuals")
```



Looking at these residual plots, we observe that there is approximately equal variance because the residuals are approximately equal in average magnitude for the fitted values across each plot. Excluding the few outliers near the bottom of the plots, we note that there is no obvious pattern since the fitted values are scattered mostly randomly across the plots. Therefore, these plots indicate linearity. From this, we understand that first-order, linear predictors should be used in the model.

To identify the best fitting model, we need to identify which predictors to incorporate. To do so, we can evaluate the results from both forward and backward stepwise selection as well as best subsets regression.

```
# fit an empty model with only the response
FitStart <- lm(MoV ~ 1, mydata)

# fit a full model with all predictors
FitAll <- lm(MoV ~ PtsScoredRank + PtsAllowedRank + YdsGainedRank + YdsAllowedRank, mydata)
```

```
# run the stepwise regression with forward selection based on the AIC criterion
step(FitStart, direction = "forward", scope = formula(FitAll))
```

```
## Start:  AIC=227.11
## MoV ~ 1
##
##                   Df Sum of Sq    RSS    AIC
## + PtsScoredRank    1   1487.12 1191.4 181.31
## + YdsGainedRank    1   1152.07 1526.4 195.94
## + PtsAllowedRank   1   1035.07 1643.4 200.29
## + YdsAllowedRank   1    550.61 2127.9 215.53
## <none>                         2678.5 227.11
##
## Step:  AIC=181.31
## MoV ~ PtsScoredRank
##
##                   Df Sum of Sq     RSS    AIC
## + PtsAllowedRank   1    366.47  824.89 161.62
## + YdsAllowedRank   1    308.49  882.88 165.63
## <none>                         1191.36 181.31
## + YdsGainedRank    1      2.33 1189.04 183.20
##
## Step:  AIC=161.63
## MoV ~ PtsScoredRank + PtsAllowedRank
##
##                   Df Sum of Sq    RSS    AIC
## + YdsAllowedRank   1    40.141 784.75 160.68
## <none>                         824.89 161.62
## + YdsGainedRank    1     0.062 824.83 163.62
##
## Step:  AIC=160.68
## MoV ~ PtsScoredRank + PtsAllowedRank + YdsAllowedRank
##
##                  Df Sum of Sq    RSS    AIC
## <none>                        784.75 160.68
## + YdsGainedRank  1    1.1982 783.55 162.59

##
## Call:
## lm(formula = MoV ~ PtsScoredRank + PtsAllowedRank + YdsAllowedRank,
##     data = mydata)
##
## Coefficients:
##    (Intercept)   PtsScoredRank  PtsAllowedRank  YdsAllowedRank
##        12.7631         -0.4735         -0.2508         -0.1655
```

The predictors selected by forward stepwise selection include `PtsScoredRank`, `PtsAllowedRank`, and `YdsAllowedRank`. The regression equation of this relationship is modeled by

$$y = 12.7631 - 0.4735x_1 - 0.2508x_2 - 0.1655x_3$$

where y represents the margin of victory, x_1 represents the season rank in points scored, x_2 represents the season rank in points allowed, and x_3 represents the season rank in yards allowed.

```
# run the stepwise regression with backward selection based on the AIC criterion
step(FitAll, direction = "backward", scope = formula(FitStart))
```

```
## Start:  AIC=162.59
## MoV ~ PtsScoredRank + PtsAllowedRank + YdsGainedRank + YdsAllowedRank
##
##                   Df Sum of Sq     RSS    AIC
## - YdsGainedRank    1     1.198  784.75 160.68
## <none>                          783.55 162.59
## - YdsAllowedRank   1    41.277  824.83 163.62
## - PtsAllowedRank   1    95.303  878.86 167.36
## - PtsScoredRank    1   271.382 1054.93 178.14
##
## Step:  AIC=160.68
## MoV ~ PtsScoredRank + PtsAllowedRank + YdsAllowedRank
##
##                   Df Sum of Sq     RSS    AIC
## <none>                          784.75 160.68
## - YdsAllowedRank   1     40.14  824.89 161.62
## - PtsAllowedRank   1     98.12  882.88 165.63
## - PtsScoredRank    1    853.09 1637.84 202.09

##
## Call:
## lm(formula = MoV ~ PtsScoredRank + PtsAllowedRank + YdsAllowedRank,
##     data = mydata)
##
## Coefficients:
##    (Intercept)   PtsScoredRank  PtsAllowedRank  YdsAllowedRank
##        12.7631         -0.4735         -0.2508         -0.1655
```

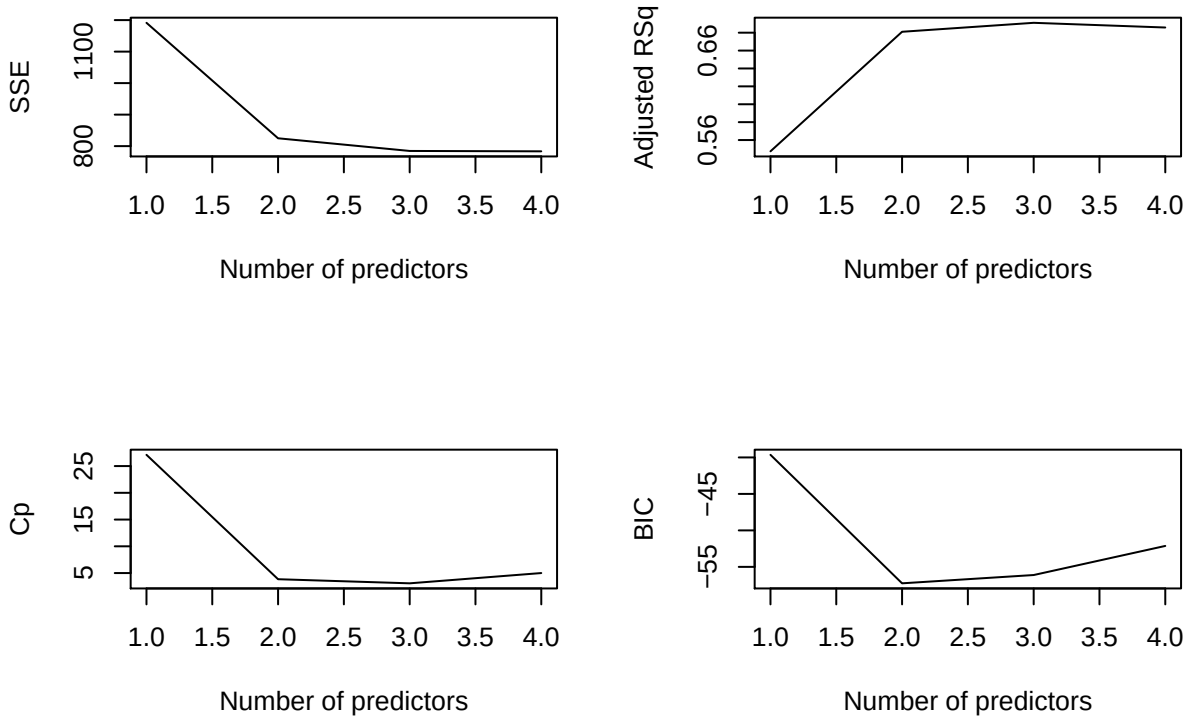
The predictors selected by backward stepwise selection include `PtsScoredRank`, `PtsAllowedRank`, and `YdsAllowedRank`. The regression equation of this relationship is modeled by

$$y = 12.7631 - 0.4735x_1 - 0.2508x_2 - 0.1655x_3$$

where y represents the margin of victory, x_1 represents the season rank in points scored, x_2 represents the season rank in points allowed, and x_3 represents the season rank in yards allowed.

```
# find the best model for each number of predictors (with 3 predictors maximum)
models <- regsubsets(MoV ~ PtsScoredRank + PtsAllowedRank +
                           YdsGainedRank + YdsAllowedRank, mydata, nvmax = 4)
models.sum <- summary(models)

# create four plots within a 2x2 frame to compare the different criteria
par(mfrow = c(2,2))
```

```r
# SSE
plot(models.sum$rss, xlab = "Number of predictors", ylab = "SSE", type = "l")

# R2
plot(models.sum$adjr2, xlab = "Number of predictors", ylab = "Adjusted RSq" , type = "l")

# Mallow's Cp
plot(models.sum$cp, xlab = "Number of predictors", ylab = "Cp", type = "l")

# BIC
plot(models.sum$bic, xlab = "Number of predictors", ylab = "BIC", type = "l")
```




```r
# display the best model for each number of predictors
models.sum$outmat
```

```
##          PtsScoredRank PtsAllowedRank YdsGainedRank YdsAllowedRank
## 1  ( 1 ) "*"           " "            " "           " "
## 2  ( 1 ) "*"           "*"            " "           " "
## 3  ( 1 ) "*"           "*"            " "           "*"
## 4  ( 1 ) "*"           "*"            "*"           "*"
```

From the SSE graph, we notice that using three or more predictors results in the smallest SSE. A smaller SSE is preferrable SSE because it results in a larger R^2.

From the Adjusted R^2 graph, we notice that using three predictors results in the largest adjusted R^2. A larger R^2 is preferrable because it conveys that more of the variance is accounted for by the model.

From the Mallow's C_p graph, we notice that using three predictors results in the smallest C_p. A smaller C_p is preferrable because it conveys that there is less bias introduced into the predicted responses by having an underspecified model.

From the BIC graph, we notice that using two or three predictors results in the smallest BIC. A smaller BIC is preferrable because it depends on the SSE (which we also want to minimize).

From these graphs, the best subsets regression identifies that three predictors should be incorporated in the model.

The predictors selected by best subsets regression include `PtsScoredRank`, `PtsAllowedRank`, and `YdsAllowedRank`. The regression equation of this relationship is modeled by

$$y = 12.7631 - 0.4735x_1 - 0.2508x_2 - 0.1655x_3$$

where y represents the margin of victory, x_1 represents the season rank in points scored, x_2 represents the season rank in points allowed, and x_3 represents the season rank in yards allowed.

From these three methods, we notice that the predictors and the regression equations are the same for all three methods.

We can now check the validity of this model. Since `YdsGainedRank` is no longer used as a predictor, per the Hierarchy Principle, the only interaction effect we can incorporate is between `PtsAllowedRank` and `YdsAllowedRank`.

```
# multiple regression model with 3 predictors and 1 interaction effect
reg2 <- lm(MoV ~ PtsScoredRank + PtsAllowedRank + YdsAllowedRank +
                 PtsAllowedRank * YdsAllowedRank, mydata)

summary(reg2)
```

```
##
## Call:
## lm(formula = MoV ~ PtsScoredRank + PtsAllowedRank + YdsAllowedRank +
##     PtsAllowedRank * YdsAllowedRank, data = mydata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.8454  -0.8006   0.8375   1.7857   5.8606
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   13.947451   1.548390   9.008 2.42e-12 ***
## PtsScoredRank                 -0.459528   0.062654  -7.334 1.18e-09 ***
## PtsAllowedRank                -0.386710   0.162081  -2.386   0.0206 *
## YdsAllowedRank                -0.288266   0.153935  -1.873   0.0665 .
## PtsAllowedRank:YdsAllowedRank  0.009743   0.009383   1.038   0.3037
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.775 on 54 degrees of freedom
## Multiple R-squared:  0.7128, Adjusted R-squared:  0.6915
## F-statistic:  33.5 on 4 and 54 DF,  p-value: 4.778e-14
```

The regression equation of this relationship is modeled by

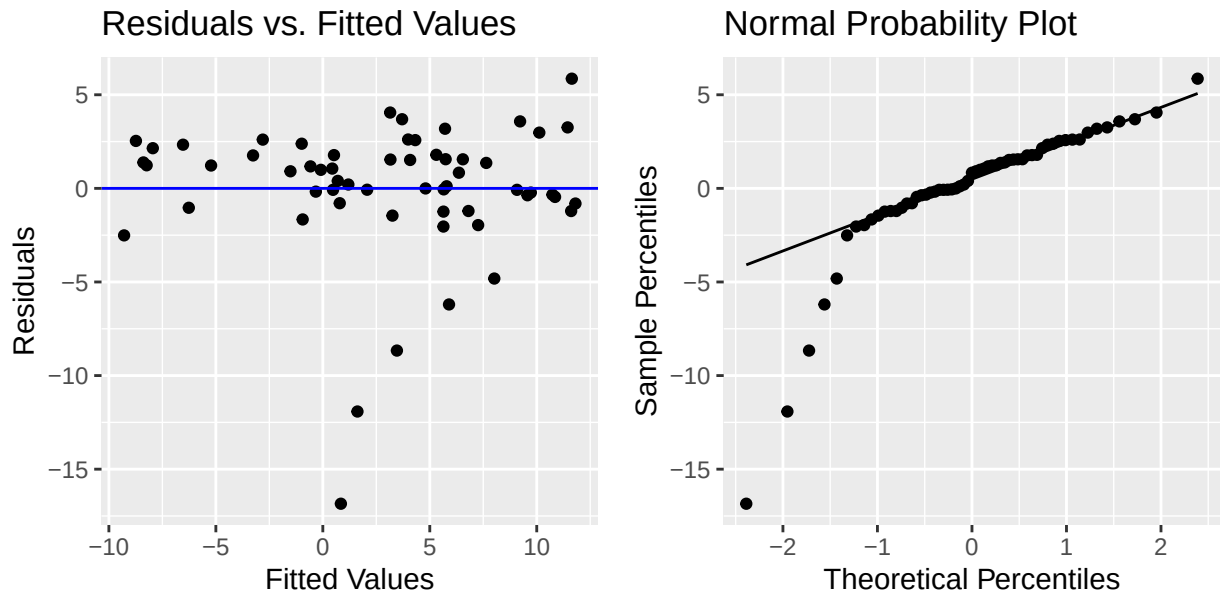
$$y = 13.9475 - 0.4595x_1 - 0.3867x_2 - 0.2882x_3 + 0.0097x_2x_3$$

where y represents the margin of victory, x_1 represents the season rank in points scored, x_2 represents the season rank in points allowed, and x_3 represents the season rank in yards allowed.

From this base regression model, we notice that the R^2 is 0.7128 and the R^2_{adj} is 0.6915. The adjusted coefficient of determination of 0.6915 indicates that 69.15% of the variation in the response variable of `MoV` is accounted for by the predictor variables. In other words, when these predictor variables are considered, the variation in `MoV` is reduced by 69.15%.

We notice that this coefficient of determination is an improvement over the previous model. To further explore the viability of this model, we can check the diagnostics to verify the assumptions.

```
# residuals vs. fitted values
mydata$resids2 <- residuals(reg2)
mydata$predicted2 <- predict(reg2)
ggplot(mydata, aes(x = predicted2, y = resids2)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "blue") +
  labs(title ="Residuals vs. Fitted Values",
       x = "Fitted Values",
       y = "Residuals")

# normal probability plot
ggplot(mydata, aes(sample = resids2)) +
  stat_qq() +
  stat_qq_line() +
  labs(title = "Normal Probability Plot",
       x = "Theoretical Percentiles",
       y = "Sample Percentiles")
```



From this residual plot, we observe that there is approximately equal variance because the residuals are approximately equal in average magnitude for the fitted values across the plot. Excluding the few outliers

near the bottom of the plot, we note that there is no obvious pattern since the fitted values are scattered mostly randomly across the plot. Therefore, this plot indicates linearity.

From this normal probability plot, we can observe that the data points form an approximately straight line and line up mostly along the line shown in the plot. This indicates that the normal distribution is a good model because the plot shows no significant deviation from a normal distribution of error terms. Since there is deviation from the line around the extremes (specifically at the lower extreme), this plot indicates heavy tails. Otherwise, the assumption that the residuals are normally distributed is approximately met.

```
# calculate the Variance Inflation Factor for each predictor
vif(reg2)
```

```
##                  PtsScoredRank                 PtsAllowedRank
##                       1.241950                       6.106687
##                 YdsAllowedRank PtsAllowedRank:YdsAllowedRank
##                       4.563227                      11.670645
```

Looking at the VIF for each of the predictors in this model, we notice that all but one predictor has a high VIF (close to or greater than 5). To account for this, we can center the predictors to reduce multicollinearity.

```
# center the predictors
mydata <- mydata %>%
          mutate(PtsScoredRank.c = PtsScoredRank - mean(PtsScoredRank),
                 PtsAllowedRank.c = PtsAllowedRank - mean(PtsAllowedRank),
                 YdsGainedRank.c = YdsGainedRank - mean(YdsGainedRank),
                 YdsAllowedRank.c = YdsAllowedRank - mean(YdsAllowedRank))

# fit the regression model with centered predictors
reg3 <- lm(MoV ~ PtsScoredRank.c + PtsAllowedRank.c + YdsAllowedRank.c +
                 PtsAllowedRank.c * YdsAllowedRank.c, mydata)

summary(reg3)
```

```
##
## Call:
## lm(formula = MoV ~ PtsScoredRank.c + PtsAllowedRank.c + YdsAllowedRank.c +
##     PtsAllowedRank.c * YdsAllowedRank.c, data = mydata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.8454  -0.8006   0.8375   1.7857   5.8606
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        2.635055   0.588456   4.478 3.94e-05 ***
## PtsScoredRank.c                   -0.459528   0.062654  -7.334 1.18e-09 ***
## PtsAllowedRank.c                  -0.283174   0.100524  -2.817  0.00676 **
## YdsAllowedRank.c                  -0.177300   0.099274  -1.786  0.07972 .
## PtsAllowedRank.c:YdsAllowedRank.c  0.009743   0.009383   1.038  0.30373
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.775 on 54 degrees of freedom
```

```
## Multiple R-squared:  0.7128, Adjusted R-squared:  0.6915
## F-statistic:  33.5 on 4 and 54 DF,  p-value: 4.778e-14
```

The regression equation of this relationship is modeled by

$$y = 2.6351 - 0.4595x_1 - 0.2832x_2 - 0.1773x_3 + 0.0097x_2x_3$$

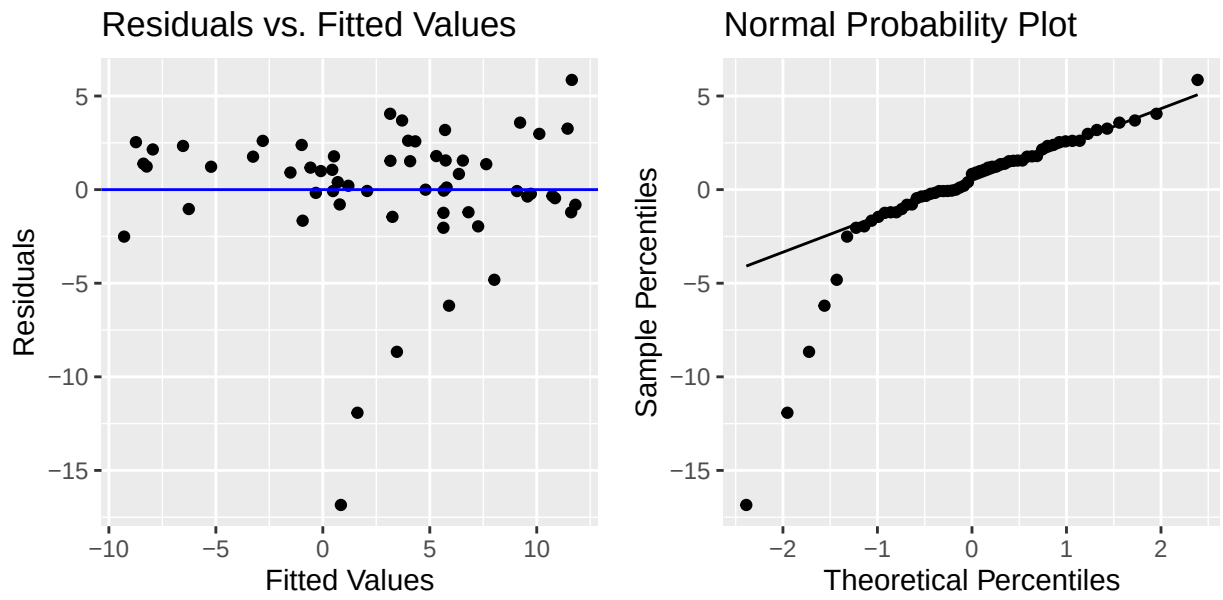
where y represents the margin of victory, x_1 represents the season rank in points scored, x_2 represents the season rank in points allowed, and x_3 represents the season rank in yards allowed.

From this base regression model, we notice that the R^2 is 0.7128 and the R^2_{adj} is 0.6915. The adjusted coefficient of determination of 0.6915 indicates that 69.15% of the variation in the response variable of `MoV` is accounted for by the predictor variables. In other words, when these predictor variables are considered, the variation in `MoV` is reduced by 69.15%.

The coefficient of determination is the same as the previous model because centering the predictors is a translation on all of the data points, so every data point is shifted by the same amount and the variance is unaffected.

To verify the assumptions, we can check the diagnostics for this updated model.

```
# residuals vs. fitted values
mydata$resids3 <- residuals(reg3)
mydata$predicted3 <- predict(reg3)
ggplot(mydata, aes(x = predicted3, y = resids3)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "blue") +
  labs(title ="Residuals vs. Fitted Values",
       x = "Fitted Values",
       y = "Residuals")

# normal probability plot
ggplot(mydata, aes(sample = resids3)) +
  stat_qq() +
  stat_qq_line() +
  labs(title = "Normal Probability Plot",
       x = "Theoretical Percentiles",
       y = "Sample Percentiles")
```



From this residual plot, we observe that there is approximately equal variance because the residuals are approximately equal in average magnitude for the fitted values across the plot. Excluding the few outliers near the bottom of the plot, we note that there is no obvious pattern since the fitted values are scattered mostly randomly across the plot. Therefore, this plot indicates linearity.

From this normal probability plot, we can observe that the data points form an approximately straight line and line up mostly along the line shown in the plot. This indicates that the normal distribution is a good model because the plot shows no significant deviation from a normal distribution of error terms. Since there is deviation from the line around the extremes (specifically at the lower extreme), this plot indicates heavy tails. Otherwise, the assumption that the residuals are normally distributed is approximately met.

```
# updated VIF
vif(reg3)
```

```
##                  PtsScoredRank.c                  PtsAllowedRank.c
##                         1.241950                         2.348999
##                YdsAllowedRank.c PtsAllowedRank.c:YdsAllowedRank.c
##                         1.897890                         1.312197
```

After centering the predictors, we notice that the VIF for each of the predictors has decreased significantly and are all between 1 and 2. From this, we understand that centering the predictors helped resolve the issue of multicollinearity.

Therefore, by removing outliers, analyzing relationships between predictors, using forward and backward stepwise selection and best subsets regression, and centering predictors to reduce multicollinearity, the best fit model is represented by the equation

$$y = 2.6351 - 0.4595x_1 - 0.2832x_2 - 0.1773x_3 + 0.0097x_2x_3$$

where y represents the margin of victory, x_1 represents the season rank in points scored, x_2 represents the season rank in points allowed, and x_3 represents the season rank in yards allowed.