Analysis of FIFA 18 Player Dataset Based on Player Attributes

Team Steam: Jaemin Kim, Siwook Yong, Sang Min Lee

December 15, 2018

Introduction

Inspired by the game and our interests in football, our team decided to use the FIFA 18 player dataset to perform various statistical analysis. Data was publicly available at Kaggle (https://www.kaggle.com/thec03u5/fifa-18-demo-player-dataset) with complete information of every player's stats and attributes. This dataset provides information for over 17,000 players in FIFA database. It includes various traits of each player such as name, age, nationality, preferred positions, acceleration, ball control, dribbling, reaction, and 67 other variables. With the ever increasing popularity of football, FIFA series gained their reputation as well. It established solid number of fans and gamers, and each sequel that is released annually improved previous editions with new features overall and updated player stats. We wanted to see how closely these in-game player stats match player values in real life, and how the overall ratings in FIFA 18 are scored based on individual stats. This paper aims to lay foundation on football player stats analysis with the following objectives:

- Categorization of positions based on examining scores of player stats and performing clustering analysis for exploratory data analysis to find patterns and grouping in data.
- Player stats modeling through testing several regression analysis (including linear regression, Ridge/Lasso Regression, and regression spline) to find the best fitting model.
- Explore the use of such statistics in our dataset and discover new findings and/or prediction, such as prediction of FIFA FIFPro World XI, for players in FIFA 18.

This paper is structured with 5 subdivided sections. Section 2 presents a review of the literature and clarifies the intended contributions and references of this paper. Section 3 introduces the summary statistics and data visualization of our FIFA 18 dataset. Section 4 presents our proposed analysis and statistical learning tasks. Section 5 concludes our study with summary of scientific findings, and discusses any potential improvements in our analysis.

Literature Review

World Cup, Olympics, and most importantly, tournaments and leagues among professional football clubs draw attention of many football fans and public audiences worldwide. Derived from the name of Fédération Internationale de Football Association(FIFA), which is an international sports organization that governs over majority of soccer events and occasions worldwide, Electronic Arts(EA Sports) has been releasing the well-known FIFA series for consoles/PC annually since 1994 to meet the interests and demands of football fans, creating a whole new genre in game industry. One of the reasons this series is loved by so many football fans worldwide is that each sequel keeps track of the most up-to-date player stats and team stats every year and closely reflects this data into their game engine, allowing gamers and soccer fans to have the very realistic sensation of controlling players and managing professional clubs in the game.

FIFA 18, the edition used in our analysis, was released worldwide on 29 September 2017 for Microsoft Windows, PlayStation 3, PlayStation 4, Xbox 360, Xbox One and Nintendo Switch, and it is the 25th instalment in the FIFA series with Cristiano Ronaldo as cover player. The game features 52 fully licensed stadiums from 12 countries, including new stadiums, plus 30 generic fields for a total of 82. All 20 Premier League stadium are represented in the series. Most importantly, it has all the updated player and club information up to this year's play records and match results. The player rating data always becomes a hot topic within gaming communities shortly after the release of each game in the series. It contains over 700 clubs and around 18,000 players, summing up to millions of data points in total that makes up this amazing game.

As part of our objectives, we included the prediction of FIFA FIFPro World XI using the in-game player stats data. FIFA FIFPro World XI is the best eleven football players of the year. Unlike the Ballon d'Or prize, which is voted for by national team captains and coaches, as well as select journalists, the FIFPro XI is made up solely of names chosen by over 50,000 professional footballers from a total of 70 countries. A list of 55 players, consisting of 5 goalkeepers, 20 defenders, 15 midfielders, and 15 forwards are nominated initially. For the finalists, voters must select 1 goalkeeper, 4 defenders, 3 midfielders and 3 forwards. FIFPro only makes a distinction between the four lines, not between each position, and it simply asks for the best goalkeeper, 4 best defenders, 3 best midfielders and 3 best attackers.

Summary Statistics

Before producing any summary report and proceeding further on our analysis, data cleaning was initially performed. We went through all 75 variables in the raw data, and chose variables that are not related or needed in our analysis.

Column Names

```
##  [1] "X"                   "Name"                "Age"
##  [4] "Photo"               "Nationality"         "Flag"
##  [7] "Overall"             "Potential"           "Club"
## [10] "Club.Logo"           "Value"               "Wage"
## [13] "Special"             "Acceleration"        "Aggression"
## [16] "Agility"             "Balance"             "Ball.control"
## [19] "Composure"           "Crossing"            "Curve"
## [22] "Dribbling"           "Finishing"           "Free.kick.accuracy"
## [25] "GK.diving"           "GK.handling"         "GK.kicking"
## [28] "GK.positioning"      "GK.reflexes"         "Heading.accuracy"
## [31] "Interceptions"       "Jumping"             "Long.passing"
## [34] "Long.shots"          "Marking"             "Penalties"
## [37] "Positioning"         "Reactions"           "Short.passing"
## [40] "Shot.power"          "Sliding.tackle"      "Sprint.speed"
## [43] "Stamina"             "Standing.tackle"     "Strength"
## [46] "Vision"              "Volleys"             "CAM"
## [49] "CB"                  "CDM"                 "CF"
## [52] "CM"                  "ID"                  "LAM"
## [55] "LB"                  "LCB"                 "LCM"
## [58] "LDM"                 "LF"                  "LM"
## [61] "LS"                  "LW"                  "LWB"
## [64] "Preferred.Positions" "RAM"                 "RB"
## [67] "RCB"                 "RCM"                 "RDM"
## [70] "RF"                  "RM"                  "RS"
## [73] "RW"                  "RWB"                 "ST"
```

The ones we chose to remove are: *X(preset row number), Photo, Flag, Club.Logo, Special, ID, and Preferred.Positions.* Then, we had to make the formats of our variables constant. Some variables such as Wage had currency symbol along with numerical values causing it to be a factor variable instead of a numerical value. Also, some of the player attributes had minor updates for their scores. For example, a player named Malcom has a dribbling score of "84+1", and this caused the whole column to be non-numerical.

To resolve this issue, we created a function called **numextract<- function(string){str_extract(string, "[[:digit:]]+")}** from *stringr* library to extract just the numbers, and made all columns into numerical values by using apply and as.numerical function. Next, we found out that there are missing values for some attributes, resulting in NA. These NA values were changed to 0 to avoid NA coercion error in R when performing functions specifically for numerical values. Lastly, we removed players with Wage==0 since they are marked as inactive players who either retired or are currently not playing in any club.

```
##                      Age     Overall   Potential        Wage Acceleration
## median       25.00000000 66.00000000 71.00000000   4.0000000   67.0000000
## mean         25.10940055 66.23278633 71.21197767  11.7084532   64.6076806
## SE.mean       0.03461759  0.05257716  0.04585853   0.1742205    0.1120500
## CI.mean.0.95  0.06785387  0.10305638  0.08988721   0.3414892    0.2196290
## var          21.25083256 49.02036195 37.29258831 538.2456972  222.6415314
## std.dev       4.60986253  7.00145427  6.10676578  23.2001228   14.9211773
## coef.var      0.18359110  0.10570980  0.08575476   1.9814849    0.2309505
```

```
##               Age Overall Potential   Wage Acceleration
## nbr.val     17733   17733     17733  17733        17733
## nbr.null        0       0         0      0            0
## nbr.na          0       0         0      0            0
## min            16      46        46      1           11
## max            47      94        94    565           96
## range          31      48        48    564           85
## sum        445265 1174506   1262802 207626      1145688
```

From looking at the summary statistics, we can see that, for example, the average age is 25.14454146, and the variance of Overall score is 48.83165959. Also, we were able to confirm that we have equal number of observations for all columns and that there is no NAs in our dataset that may cause problems in our analysis. * (We only displayed 5 variables in the above statistics to prevent wasting spaces. The whole summary statistics is shown in Appendix.) **(APX(1))**

One last thing to do before performing regression analysis to find the best fitting model for our data was partitioning our data into four different positions: Forwards(OFF), Midfielders(MID), Defense(DEF), and Goal Keepers(GK). Since there were 26 specific position data such as *CAM*, *RF*, and *ST* included in our original dataset, we used these variables to determine the four fundamental positions as such:

- Forwards: *CF, LF, LS, RF, RS, ST, LW, RW*
- MID: *CM, LCM, LM, RCM, RM, CAM, LAM, RAM, CDM, LDM, RDM*
- DEF: *LB, LCB, RB, RCB, LWB, RWB, CB*
- GK: Players with 0 values for specific sub-positions.

Using these guidelines we divided our positions, and finally split the data into train and test data in ratio of 75 Train:25 Test. The number of observations in each set is described below in the tables.

Full Data

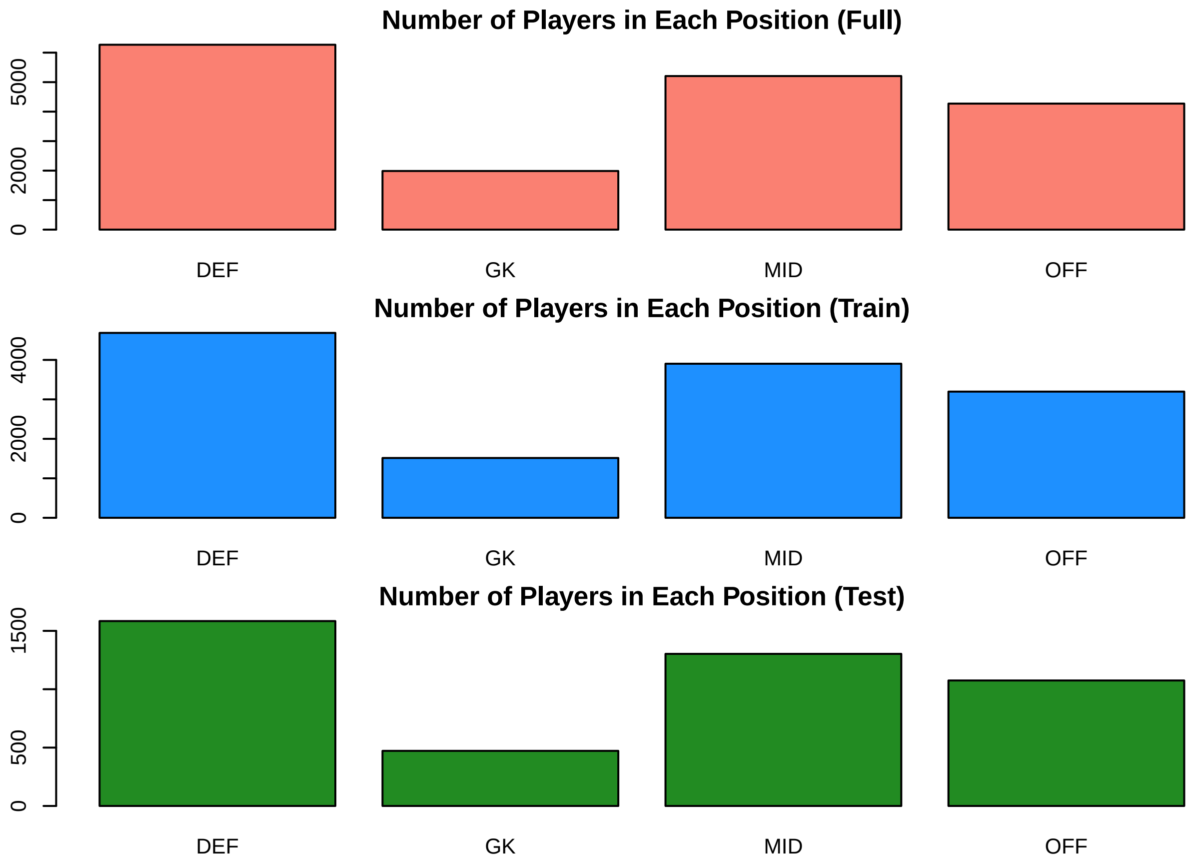
```
##  DEF   GK  MID  OFF
## 6268 1987 5206 4272
```

Train Data

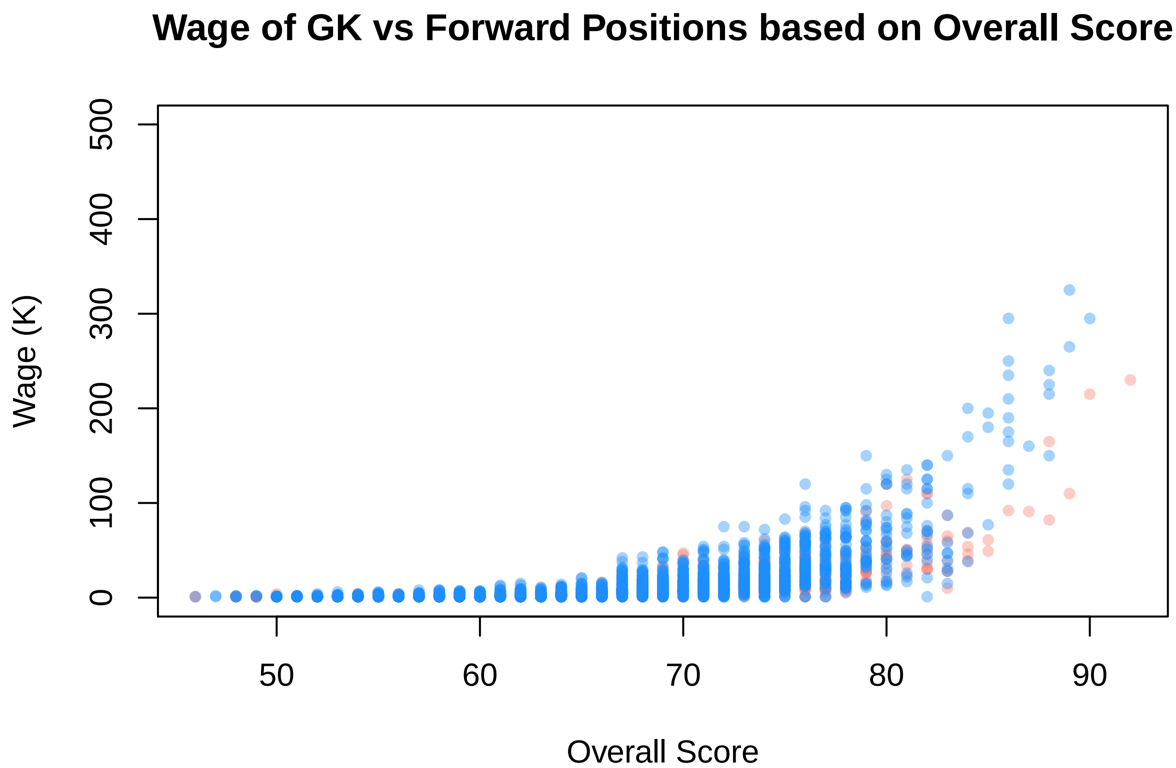
```
##  DEF   GK  MID  OFF
## 4684 1515 3903 3197
```

Test Data

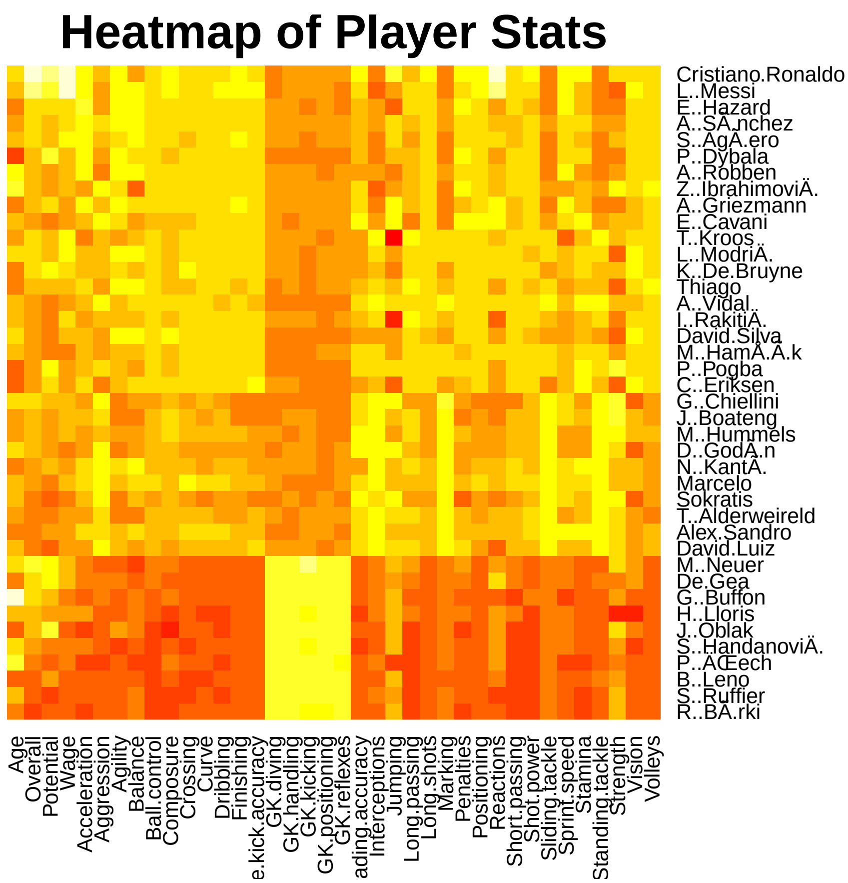
```
## DEF   GK  MID  OFF
## 1584  472 1303 1075
```



- Both from the tables and plots, we can visually see that the train and test data were split evenly in ratio. For full data before splitting, there are 6268 DEF, 1987 GK, 5206 MID, and 4272 OFF positions. After splitting, train data has 4684 DEF, 1515 GK, 3903 MID, and 3197 OFF, and test data has 1584 DEF, 472 GK, 1303 MID, 1075 OFF positions. We will be using train data to perform various analysis in this paper to find the best fitting model, and the chosen model will later be compared to the test data to see if the model truly fits our data well for both train and test data. Then, we will use this model with some of the player stats to predict the FIFA FIFPro World XI.



- This is a visualization to show, for example, if there are any difference in the wage based on positions of football players. For the comparison, two categories, goal keeper and forward positions, were drawn into one plot with different colors. As a result, we were able to find out that for players with relatively low overall Scores, there may not be a big difference in wages they receive. However, for players with high overall scores, more players in forward positions seem to get higher wages than defense position players.



- Lastly, we wanted to visually check what player stats are more important and valued for each of the four positions we divided. In order to create a clear data visualization, top 10 players with highest overall scores from each position were selected. Then, these 40 players were put into Heatmap function, which allowed us to clearly see the differences in scores for each attribute: In the y-axis, from the top, players are ordered as forward, midfielder, defense, and goal keeper. If you see closely into the heatmap, there are separation by colors that is quite visible for each attribute of each position. These are just few of the information we could visibly obtain from the heatmap:

1. Forward positions, including Cristiano Ronaldo and Lionel Messi, have relatively high Overall scores, Acceleration, Balance, Agility, and Penalties.

2. Midfielders, such as Pogba and David Silva, have better Interceptions, Long-passing, Aggression, and Sliding Tackles than forward positions while sharing similar range of scores for other stats.

3. For defend players like Boateng and Marcelo, they have superior stats in Aggression, Interceptions, Sliding Tackle, Standing Tackle, and Strength than any other positions.

4. Goal keepers generally have low scores in most of the stats compared to players in other positions, but they have a few criteria they are specialized in. All of the goal keepers have exceptional GK.diving, GK.handling, GK.kicking, GK.positioning, and GK.reflexes as shown in Heatmap as yellow rectangle area at lower part of the map.

- (Hierarchical clustering option is added in **APX (2)**)

After exploring our data with various examples and visualizations, we confirmed the data describes its values very well for our purposes. With these cleaned data that are divided into four different positions and then into train and test data, the data seemed adequate to perform statistical analysis.

Proposed Analysis

Linear Regression

In order to achieve our goal of finding the best fitting model, we first tried to fit linear regression to our model as it is the most fundamental regression used in various fields to model data. We began with fitting all variables into a linear model using lm() function to see how it generally fits our data in summary. There were number of variables that are larger than our 0.05 cut off value, therefore not significant. To find a model that represents our data better, we performed variable selection using stepwise selection via step() function, then we compared the AIC and BIC values for test goodness of fit compared to the initial model.

For linear regression analysis, only train data for forward position was used to reduce the time it takes to process large models. Although it is highly unlikely that linear regression will be the best fitting model, if it turns out that it is indeed the best model, we could later try to fit this model to three other positions as they share the same variables and relatively similar regressions originated from the same data.

```
#Linear Reg.
fit1=lm(Overall ~ Age+Wage+Acceleration+Aggression+Agility+Balance+Ball.control+Composure+Crossing+Curve
summary(fit1)
```

```
##
## Call:
## lm(formula = Overall ~ Age + Wage + Acceleration + Aggression +
##     Agility + Balance + Ball.control + Composure + Crossing +
##     Curve + Dribbling + Finishing + Free.kick.accuracy + GK.diving +
##     GK.handling + GK.kicking + GK.positioning + GK.reflexes +
##     Heading.accuracy + Interceptions + Jumping + Long.passing +
##     Long.shots + Marking + Penalties + Positioning + Reactions +
##     Short.passing + Shot.power + Sliding.tackle + Sprint.speed +
##     Stamina + Standing.tackle + Strength + Vision + Volleys,
##     data = train_off)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.2453 -0.6970  0.0112  0.7011  4.9797
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         4.6502036  0.3338450  13.929  < 2e-16 ***
## Age                 0.0079757  0.0061880   1.289   0.1975
## Wage                0.0098915  0.0008908  11.104  < 2e-16 ***
## Acceleration        0.0355028  0.0043505   8.161 4.76e-16 ***
## Aggression          0.0045844  0.0017887   2.563   0.0104 *
## Agility            -0.0023739  0.0032069  -0.740   0.4592
## Balance            -0.0057360  0.0027066  -2.119   0.0341 *
## Ball.control        0.1550723  0.0062953  24.633  < 2e-16 ***
## Composure           0.0278962  0.0037501   7.439 1.30e-13 ***
## Crossing            0.0009501  0.0027343   0.347   0.7283
## Curve               0.0016852  0.0028091   0.600   0.5486
## Dribbling           0.0848108  0.0055345  15.324  < 2e-16 ***
## Finishing           0.1308136  0.0052450  24.941  < 2e-16 ***
## Free.kick.accuracy  0.0030735  0.0023406   1.313   0.1892
## GK.diving           0.0081901  0.0063832   1.283   0.1996
## GK.handling        -0.0027549  0.0065116  -0.423   0.6723
## GK.kicking          0.0037522  0.0062532   0.600   0.5485
## GK.positioning      0.0072951  0.0065398   1.115   0.2647
```

```
## GK.reflexes         0.0044535  0.0064178   0.694   0.4878
## Heading.accuracy    0.0297474  0.0029956   9.931  < 2e-16 ***
## Interceptions      -0.0018753  0.0025594  -0.733   0.4638
## Jumping            -0.0047640  0.0020468  -2.328   0.0200 *
## Long.passing       -0.0077318  0.0032853  -2.353   0.0187 *
## Long.shots          0.0279678  0.0036658   7.629 3.10e-14 ***
## Marking            -0.0088685  0.0034543  -2.567   0.0103 *
## Penalties          -0.0054764  0.0030296  -1.808   0.0708 .
## Positioning         0.1454771  0.0047460  30.653  < 2e-16 ***
## Reactions           0.0768915  0.0040509  18.981  < 2e-16 ***
## Short.passing       0.0778228  0.0047314  16.448  < 2e-16 ***
## Shot.power          0.0749804  0.0040288  18.611  < 2e-16 ***
## Sliding.tackle     -0.0038630  0.0038270  -1.009   0.3129
## Sprint.speed        0.0443515  0.0041427  10.706  < 2e-16 ***
## Stamina             0.0002689  0.0023350   0.115   0.9083
## Standing.tackle     0.0049082  0.0036588   1.341   0.1799
## Strength            0.0333027  0.0024347  13.678  < 2e-16 ***
## Vision              0.0163039  0.0034630   4.708 2.61e-06 ***
## Volleys             0.0035902  0.0033112   1.084   0.2783
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.12 on 3160 degrees of freedom
## Multiple R-squared:  0.9741, Adjusted R-squared:  0.9738
## F-statistic:  3302 on 36 and 3160 DF,  p-value: < 2.2e-16
```

```r
#Stepwise Selection
step(fit1, trace = 0, direction = "both")
```

```
##
## Call:
## lm(formula = Overall ~ Age + Wage + Acceleration + Aggression +
##     Balance + Ball.control + Composure + Dribbling + Finishing +
##     Free.kick.accuracy + GK.diving + Heading.accuracy + Jumping +
##     Long.passing + Long.shots + Marking + Penalties + Positioning +
##     Reactions + Short.passing + Shot.power + Sprint.speed + Strength +
##     Vision, data = train_off)
##
## Coefficients:
##        (Intercept)                 Age                Wage
##           4.664428            0.009344            0.009894
##       Acceleration          Aggression             Balance
##           0.034794            0.004664           -0.006334
##       Ball.control           Composure           Dribbling
##           0.155366            0.028546            0.085126
##          Finishing  Free.kick.accuracy           GK.diving
##           0.131353            0.003985            0.009659
##   Heading.accuracy             Jumping        Long.passing
##           0.030359           -0.005124           -0.007563
##         Long.shots             Marking           Penalties
##           0.028450           -0.009205           -0.005280
##        Positioning           Reactions       Short.passing
##           0.145596            0.076983            0.078153
##         Shot.power        Sprint.speed            Strength
##           0.076585            0.044230            0.033208
```

```
##            Vision
##          0.016449
```

```
##
## Call:
## lm(formula = Overall ~ Wage + Acceleration + Aggression + Balance +
##     Ball.control + Composure + Dribbling + Finishing + Heading.accuracy +
##     Jumping + Long.shots + Marking + Positioning + Reactions +
##     Short.passing + Shot.power + Sprint.speed + Strength + Vision,
##     data = train_off)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.3401 -0.6773  0.0084  0.7126  5.0772
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      4.7404103  0.2945237  16.095  < 2e-16 ***
## Wage             0.0095310  0.0008769  10.870  < 2e-16 ***
## Acceleration     0.0344616  0.0041847   8.235 2.59e-16 ***
## Aggression       0.0055334  0.0017165   3.224  0.00128 **
## Balance         -0.0062559  0.0025690  -2.435  0.01494 *
## Ball.control     0.1555861  0.0062233  25.000  < 2e-16 ***
## Composure        0.0293285  0.0036847   7.960 2.38e-15 ***
## Dribbling        0.0849271  0.0054167  15.679  < 2e-16 ***
## Finishing        0.1290108  0.0050363  25.616  < 2e-16 ***
## Heading.accuracy 0.0297626  0.0028757  10.350  < 2e-16 ***
## Jumping         -0.0049665  0.0020049  -2.477  0.01329 *
## Long.shots       0.0294185  0.0034916   8.425  < 2e-16 ***
## Marking         -0.0098185  0.0020112  -4.882 1.10e-06 ***
## Positioning      0.1475037  0.0046443  31.760  < 2e-16 ***
## Reactions        0.0776067  0.0039936  19.433  < 2e-16 ***
## Short.passing    0.0733982  0.0041462  17.703  < 2e-16 ***
## Shot.power       0.0767796  0.0038781  19.798  < 2e-16 ***
## Sprint.speed     0.0437162  0.0040613  10.764  < 2e-16 ***
## Strength         0.0333292  0.0023563  14.145  < 2e-16 ***
## Vision           0.0156976  0.0033528   4.682 2.96e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.121 on 3177 degrees of freedom
## Multiple R-squared:  0.9739, Adjusted R-squared:  0.9738
## F-statistic:  6247 on 19 and 3177 DF,  p-value: < 2.2e-16
```

```r
BIC(fit1)
```

```
## [1] 10067.16
```

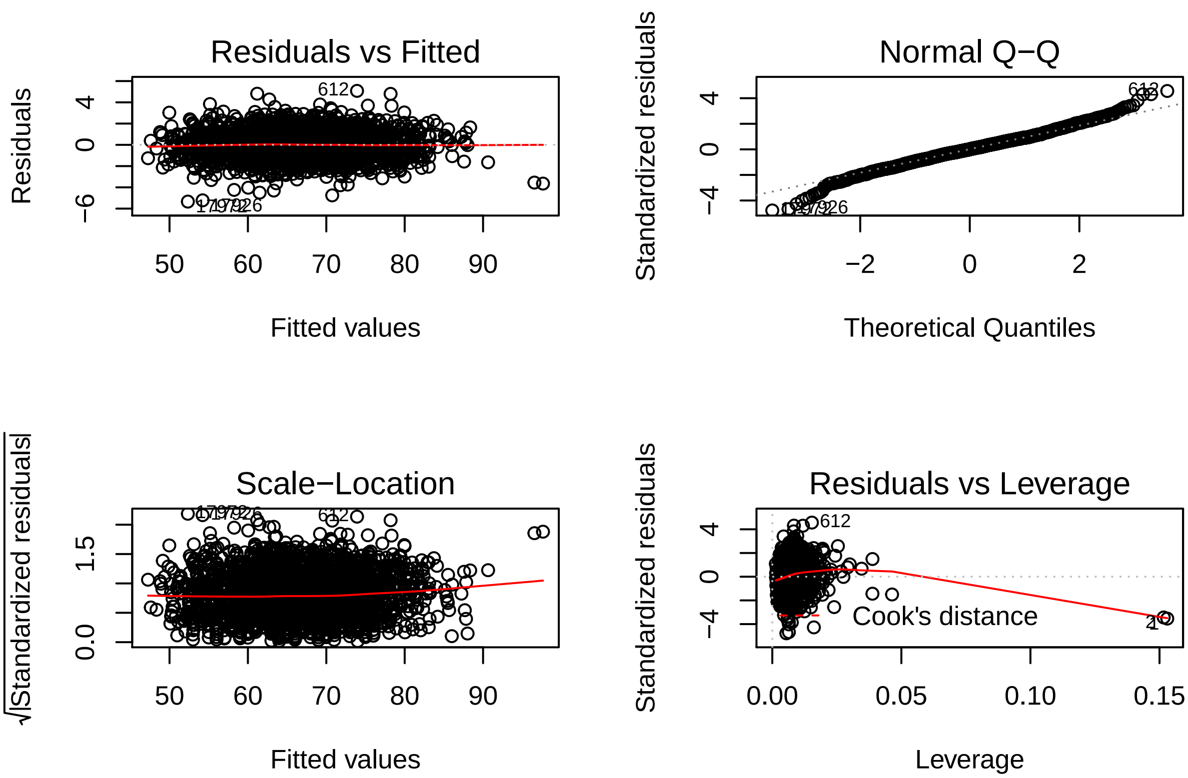
```r
BIC(fit3)
```

```
## [1] 9951.592
```

After performing variable selection via stepwise selection to find the smallest but the most effective model, we looked into each variable and left only the ones that are significant, or less than alpha=0.05. After confirming the signficance of each variable in our fit3 model, we compared the BIC values of the initial fit1 model and fit3. Since we pursue the goal of minimizing BIC value in this specific selection, we can say that fit3 model is a better fitting model than fit1, even though there was only slight reduction in BIC values.

However, we still face a major issue when doing a lack of fit test. Observing the F statistics of our model, we discovered that the value is exceptionally large with the value of 6247 with p-value of $< 2.2\text{e-}16$. This only signifies that there is significant lack of fit in our model. To furthur explore this model, we generated diagnostic plots to identify any peculiarities.

```
par(mfrow=c(2,2))
plot(fit3)
```




```
## [1] 1.248452
```

Seeing the four diagnostic plots displayed, our model does not seem that bad after all. The Residuals vs Fitted plot show that the model meets the regression assumptions well, Normal Q-Q plot shows normality except both ends, Scale-Location plot displays roughly flat line of curve. However, the Residuals vs Leverage plot shows two influential points, 1 and 2, that seems to be over Cook's distance and located far from cluster of points. These points can be removed to stabilize our model, but first we will go through other regressions to see if we can have a better fitting model.

Ridge Regression

Ridge regression is a regression method which adds a penalty by the tuning parameter called λ(lambda) that is chosen by cross-validation. The overall concept of ridge regression is making the fit small by minimizing the residual sum of squares and adding the shrinkage penalty. The shrinkage penalty computed by λ times the sum of squares of the coefficients, so coefficients that become large are the ones that get penalized. As the value of λ increases, the bias increases and the variance decreases. Now, let's see how the model of each position changes by ridge regression.

Coefficient estimate by Ridge Regression

```
##      best.lambda.off best.lambda.mid best.lambda.def best.lambda.gk
## [1,]            15.5            23.8              38            1.1
```

```
##                    Forward Midfielder Defender Goalkeeper
##                     3.5207     2.4675   3.7157     0.9251
## Acceleration        0.0362     0.0216   0.0146     0.0034
## Aggression          0.0045     0.0093   0.0361     0.0011
## Agility            -0.0031     0.0132   0.0004    -0.0005
## Balance            -0.0055    -0.0155  -0.0097    -0.0009
## Ball.control        0.1571     0.2148   0.0492    -0.0007
## Composure           0.0295     0.0400   0.0240     0.0023
## Crossing            0.0020     0.0559   0.0081    -0.0066
## Curve               0.0016    -0.0054   0.0093     0.0031
## Dribbling           0.0853     0.1081  -0.0111    -0.0023
## Finishing           0.1370     0.0265   0.0070    -0.0008
## Free.kick.accuracy  0.0031    -0.0103   0.0010     0.0023
## GK.diving           0.0058     0.0118  -0.0035     0.2109
## GK.handling        -0.0032    -0.0023  -0.0039     0.2109
## GK.kicking          0.0027     0.0114   0.0002     0.0512
## GK.positioning      0.0052     0.0012  -0.0058     0.2105
## GK.reflexes         0.0046     0.0056   0.0009     0.2083
## Heading.accuracy    0.0297     0.0131   0.0645    -0.0012
## Interceptions      -0.0021     0.0135   0.1232     0.0006
## Jumping            -0.0045     0.0061   0.0075     0.0004
## Long.passing       -0.0069     0.0265  -0.0046    -0.0017
## Long.shots          0.0277     0.0225  -0.0075    -0.0053
## Marking            -0.0106     0.0157   0.1478    -0.0022
## Penalties          -0.0027     0.0100  -0.0044     0.0021
## Positioning         0.1462     0.0241  -0.0035     0.0017
## Reactions           0.0808     0.1202   0.0904     0.1084
## Short.passing       0.0779     0.1668   0.0550     0.0016
## Shot.power          0.0747     0.0102   0.0069     0.0012
## Sliding.tackle     -0.0030    -0.0262   0.1238    -0.0021
## Sprint.speed        0.0446     0.0171   0.0379    -0.0012
## Stamina             0.0008     0.0474   0.0246    -0.0001
## Standing.tackle     0.0045     0.0058   0.1514     0.0023
## Strength            0.0339     0.0119   0.0477    -0.0002
## Vision              0.0194     0.0621  -0.0129     0.0003
## Volleys             0.0048    -0.0231  -0.0030     0.0005
```

Conclusion: The best λ is the value that produces the smallest GCV(generalized cross-validation). The coefficient of the variables have been computed using the best λ value and the output shows that the variables that are important to estimate the Overall for each position tend to have a higher coefficient value than the others. However, the problem of ridge regression is that it does not exclude the non-significant variables. Instead, it leaves it with an extremely small coefficient value.

LASSO regression

LASSO regression is a regression method which the penalty is computed by the sum of the absolute values of the coefficients. It shrinks the coefficient values to zero which is similar to ridge regression. However, the good part of LASSO regression is that unlike ridge regression, as the λ values increases, it excludes the variables that the coefficients become zero which leaves only the necessary variables to the final model.

Therefore, LASSO regression can be described as a method that can perform both shrinkage and variable selection. Now, let's see how LASSO shrinks the model for our data and which variables are left on the final model for each position.

Variable selection by LASSO

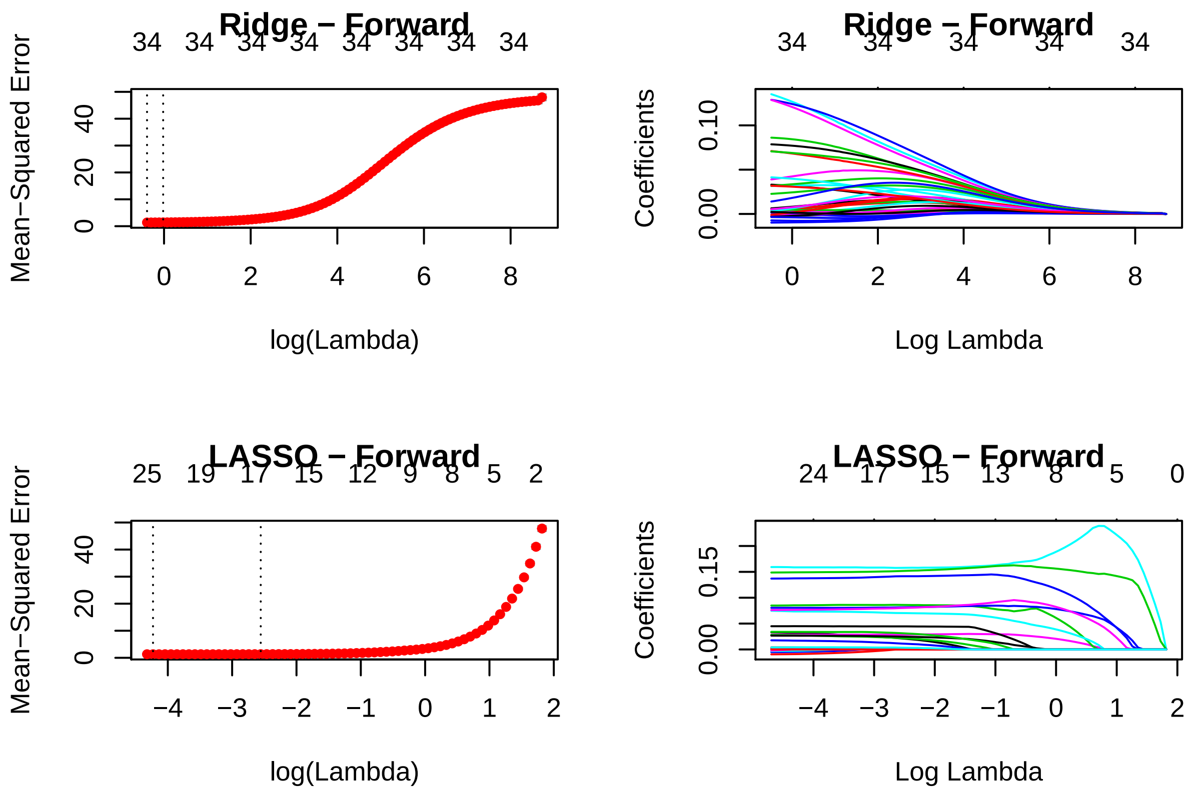
```
## 35 x 4 sparse Matrix of class "dgCMatrix"
##                      Forward Midfielder Defender Goalkeeper
## (Intercept)           3.8464     3.6528   3.9550     1.5384
## Acceleration          0.0276     0.0145   0.0058     .
## Aggression            0.0009     0.0054   0.0338     .
## Agility               .          0.0007   .          .
## Balance               .          .       -0.0016     .
## Ball.control          0.1583     0.2238   0.0427     .
## Composure             0.0289     0.0325   0.0211     .
## Crossing              .          0.0472   .          .
## Curve                 .          .        0.0038     .
## Dribbling             0.0855     0.1079   .          .
## Finishing             0.1419     0.0215   .          .
## Free.kick.accuracy    .          .        .          .
## GK.diving             .          .        .          0.2105
## GK.handling           .          .        .          0.2102
## GK.kicking            .          .        .          0.0468
## GK.positioning        .          .        .          0.2103
## GK.reflexes           .          .        .          0.2075
## Heading.accuracy      0.0272     0.0115   0.0643     .
## Interceptions         .          0.0106   0.1216     .
## Jumping               .          0.0027   0.0047     .
## Long.passing          .          0.0147   .          .
## Long.shots            0.0222     0.0193   .          .
## Marking              -0.0016     .        0.1529     .
## Penalties             .          .        .          .
## Positioning           0.1481     0.0239   .          .
## Reactions             0.0814     0.1251   0.0894     0.1063
## Short.passing         0.0715     0.1685   0.0475     .
## Shot.power            0.0812     0.0004   0.0011     .
## Sliding.tackle        .          .        0.1243     .
## Sprint.speed          0.0418     0.0143   0.0392     .
## Stamina               .          0.0528   0.0226     .
## Standing.tackle       .          .        0.1521     .
## Strength              0.0323     0.0161   0.0490     .
## Vision                0.0131     0.0573  -0.0035     .
## Volleys               0.0043    -0.0006   .          .
```

As we expected, only the important variables for the Overall of each position are left over. We can easily notice that the ouput is quite accurate by looking at the Goalkeeper column. All of the GK skill variables are left on the final model for the goalkeeper Overall and the only non-GK skill which is Reactions is also an important variable for the goalkeeper, because how fast they react to the ball coming is an essential part of their goalkeeping skill.

MSE Comparison

Mean Squared Error(MSE), also known as the prediction error, is the mean squared difference between the estimated values and the actual values. The formula for the MSE is $MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)$. Lower MSE means that the prediction is more accurate, so we will now examine which regression among ridge and LASSO is better in estimating the Overall of the players for each position using the train and test data..

Forward Position




```
##      Position     Ridge MSE            LASSO MSE
## [1,] "Forward"    "1.41573038377561"   "1.35285133431696"
## [2,] "Midfielder" "3.18781288283597"   "3.16431769082251"
## [3,] "Defender"   "1.76592913603406"   "1.77251890806939"
## [4,] "Goalkeeper" "0.177825541163576" "0.122562095555412"
```

Conclusion: LASSO regression seems to be the better prediction method than ridge regression, because it had a smaller MSE in three out of the four predictions. However, considering that the difference between the MSE of the two regressions is very small, we can conclude that both regressions work fine for predicting the Overall of the soccer players.

- Only the MSE plots and Ridge/Lasso regression plots for forward position are displayed here. Plots for rest of the position will be inlcuded in **(APX (3)**)

Spline

Conclusion

After performing several different regression analyses and comparing models from each of the regression, we used Mean Squared Error(MSE) for our method to decide which model is the best fitting model. Upon achieving all regression models and finding the MSE values, we were able to draw a conclusion that LASSO regression model is the one that fits our data the most. According to the MSE table we created below, we can see that linear model is actually quite good, most fitting model for Forward and Midfielder positions, but its MSE increases exponentially when it is compared for Defender and Goalkeeper positions among other regression models. Therefore, overall LASSO regression shows the most consistent, and yet, low MSE values throughout all positions, making it the best fitting model.

```
##      Position     Linear MSE         Ridge MSE
## [1,] "Forward"    "1.24845237065937" "1.41573038377561"
## [2,] "Midfielder" "3.05832215922678" "3.18781288283597"
## [3,] "Defender"   "2.9894586793159"  "1.76592913603406"
## [4,] "Goalkeeper" "12.2807791259053" "0.177825541163576"
##      LASSO MSE
## [1,] "1.35285133431696"
## [2,] "3.16431769082251"
## [3,] "1.77251890806939"
## [4,] "0.122562095555412"
```

As for predicting the next FIFA FIFPro World XI, we used data selection method to extract the top players from each position using the variable *potential*. Then, we were able to draw top 10 players for each position in order of high to low values.

```
##                   OFF             MID            DEF            GK
## 1  Cristiano.Ronaldo   K..De.Bruyne      R..Varane G..Donnarumma
## 2             Neymar       P..Pogba   Sergio.Ramos      J..Oblak
## 3          K..MbappÃ.  Marco.Asensio      N..KantÃ.      M..Neuer
## 4            L..Messi    M..Verratti   G..Chiellini        De.Gea
## 5           P..Dybala     C..Eriksen      E..Bailly   T..Courtois
## 6          L..SuÃ.rez Bernardo.Silva     A..Laporte       T..Horn
## 7         O..DembÃ.lÃ.       T..Lemar       N..SÃ.le     G..Buffon
## 8       Gabriel.Jesus       T..Kroos     Marquinhos M..ter.Stegen
## 9     R..Lewandowski         Thiago A..Christensen      G..Rulli
## 10          E..Hazard           Isco     M..de.Ligt      Ederson
```

Following the number of nominees for each position for this year's FIFA FIFPro World XI, next year's nominees will be:

- Forward: Cristiano.Ronaldo, Neymar, K..MbappÃ.
- Midfielder: K..De.Bruyne, P..Pogba, Marco.Asensio
- Defense: R..Varane, Sergio.Ramos, N..KantÃ., G..Chiellini
- Goalkeeper: G..Donnarumma

Since the World XI are voted subjectively by 50,000 professional football players worldwide, the actual nominees may vary from our results as they are not purely judged by player attributes, but also include popularity, reputation, and out-of-sports activites. For potential improvements for our analyses, we believe there may exist regression models that suit FIFA 18 dataset better than LASSO regression. Some weighted analysis such as kernel distributions could be used to improve the accuracy and fitness of models.

Appendix

(APX 1) Summary Statistics of All Variables

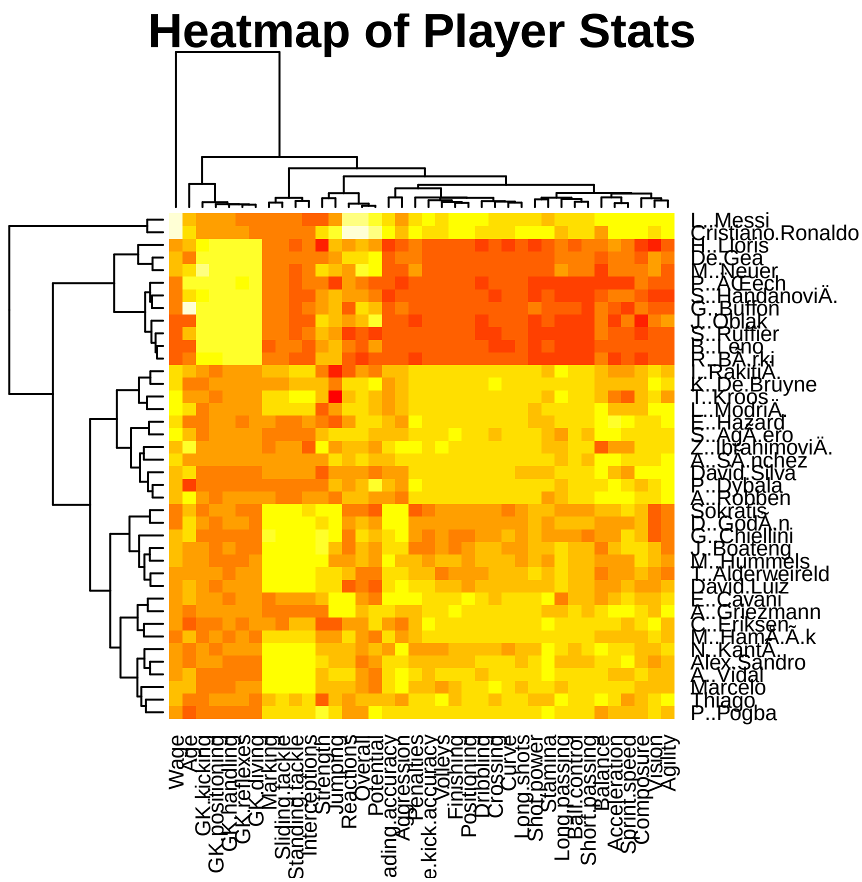
```
##                      Age     Overall   Potential        Wage Acceleration
## median       25.00000000 66.00000000 71.00000000   4.0000000   67.0000000
## mean         25.10940055 66.23278633 71.21197767  11.7084532   64.6076806
## SE.mean       0.03461759  0.05257716  0.04585853   0.1742205    0.1120500
## CI.mean.0.95  0.06785387  0.10305638  0.08988721   0.3414892    0.2196290
## var          21.25083256 49.02036195 37.29258831 538.2456972  222.6415314
## std.dev       4.60986253  7.00145427  6.10676578  23.2001228   14.9211773
## coef.var      0.18359110  0.10570980  0.08575476   1.9814849    0.2309505
##               Aggression     Agility     Balance Ball.control    Composure
## median        59.0000000  65.0000000  66.0000000   63.0000000  60.00000000
## mean          55.8373654  63.3917555  63.8170078   58.1688942  57.93419049
## SE.mean        0.1310700   0.1110035   0.1059302    0.1258468   0.09678235
## CI.mean.0.95   0.2569101   0.2175778   0.2076336    0.2466720   0.18970286
## var          304.6414934 218.5022262 198.9857427  280.8448659 166.10186106
## std.dev       17.4539822  14.7818208  14.1062306   16.7584267  12.88805110
## coef.var       0.3125861   0.2331821   0.2210419    0.2880995   0.22246019
##                 Crossing       Curve   Dribbling   Finishing
## median        54.0000000  49.0000000  61.0000000  48.0000000
## mean          49.8061806  47.3606835  55.1224835  45.3512660
## SE.mean        0.1383977   0.1384760   0.1424248   0.1460763
## CI.mean.0.95   0.2712731   0.2714266   0.2791666   0.2863239
## var          339.6567134 340.0411160 359.7110290 378.3920011
## std.dev       18.4297779  18.4402038  18.9660494  19.4523007
## coef.var       0.3700299   0.3893568   0.3440710   0.4289252
##              Free.kick.accuracy   GK.diving GK.handling  GK.kicking
## median               42.0000000  11.0000000  11.0000000  11.0000000
## mean                 43.1969774  16.6869678  16.4667569  16.3426944
## SE.mean               0.1318926   0.1330990   0.1272511   0.1241439
## CI.mean.0.95          0.2585225   0.2608871   0.2494245   0.2433341
## var                 308.4773831 314.1463658 287.1475103 273.2956489
## std.dev              17.5635242  17.7241746  16.9454274  16.5316560
## coef.var              0.4065915   1.0621567   1.0290689   1.0115624
##              GK.positioning GK.reflexes Heading.accuracy Interceptions
## median           11.0000000  11.0000000       56.0000000    52.0000000
## mean             16.4496701  16.8192635       52.3553826    46.5290137
## SE.mean           0.1280568   0.1352571        0.1305055     0.1553531
## CI.mean.0.95      0.2510039   0.2651171        0.2558036     0.3045073
## var             290.7954168 324.4159564      302.0229178   427.9787008
## std.dev          17.0527246  18.0115506       17.3788066    20.6876461
## coef.var          1.0366606   1.0708882        0.3319393     0.4446182
##                    Jumping Long.passing  Long.shots     Marking   Penalties
## median         66.00000000   56.0000000  51.0000000  48.0000000  50.0000000
## mean           64.93334461   52.4694073  47.2575424  44.0634974  48.9982518
## SE.mean         0.08938792    0.1164191   0.1445837   0.1618855   0.1185647
## CI.mean.0.95    0.17520907    0.2281928   0.2833981   0.3173114   0.2323985
## var           141.69023310  240.3425814 370.6983309 464.7273010 249.2833829
## std.dev        11.90337066   15.5029862  19.2535278  21.5575347  15.7887106
## coef.var        0.18331676    0.2954671   0.4074170   0.4892379   0.3222301
##              Positioning   Reactions Short.passing  Shot.power
## median        55.0000000 62.00000000    62.0000000  59.0000000
```

```
## mean          49.6961033 61.89257317    58.3413974  55.7149946
## SE.mean        0.1457858  0.06896257     0.1117357   0.1302920
## CI.mean.0.95   0.2857545  0.13517337     0.2190130   0.2553851
## var          376.8885236 84.33523302   221.3942691 301.0356183
## std.dev       19.4136170  9.18342164    14.8793235  17.3503780
## coef.var       0.3906467  0.14837679     0.2550389   0.3114131
##              Sliding.tackle Sprint.speed     Stamina Standing.tackle
## median           51.0000000   67.0000000  66.0000000      54.0000000
## mean             45.5464388   64.8474031  63.2771669      47.4183161
## SE.mean           0.1612511    0.1100176   0.1194722       0.1639058
## CI.mean.0.95      0.3160680    0.2156453   0.2341772       0.3212714
## var             461.0922066  214.6380032 253.1138464     476.3989923
## std.dev          21.4730577   14.6505291  15.9095520      21.8265662
## coef.var          0.4714542    0.2259231   0.2514264       0.4602982
##                  Strength      Vision     Volleys pos
## median        66.00000000  55.0000000  44.0000000  NA
## mean          65.31573902  53.0377263  43.2660576  NA
## SE.mean        0.09460772   0.1077997   0.1330724  NA
## CI.mean.0.95   0.18544038   0.2112979   0.2608350  NA
## var          158.72136122 206.0710445 314.0207952  NA
## std.dev       12.59846662  14.3551748  17.7206319  NA
## coef.var       0.19288562   0.2706597   0.4095735  NA
```

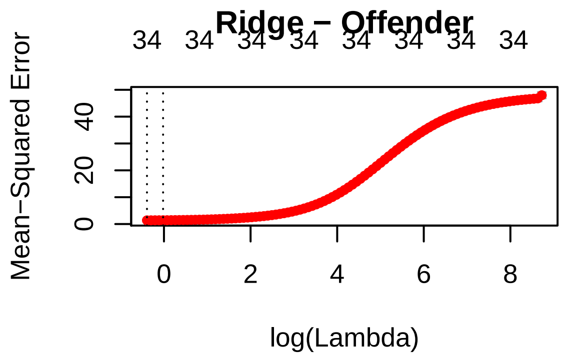
```
##              Age Overall Potential   Wage Acceleration Aggression Agility
## nbr.val    17733   17733     17733  17733        17733      17733   17733
## nbr.null       0       0         0      0            0          0       0
## nbr.na         0       0         0      0            0          0       0
## min           16      46        46      1           11         11      14
## max           47      94        94    565           96         96      96
## range         31      48        48    564           85         85      82
## sum       445265 1174506   1262802 207626      1145688     990164 1124126
##          Balance Ball.control Composure Crossing  Curve Dribbling
## nbr.val    17733        17733     17733    17733  17733     17733
## nbr.null       0            0         0        0      0         0
## nbr.na         0            0         0        0      0         0
## min           11            8         5        5      6         2
## max           96           95        96       91     92        97
## range         85           87        91       86     86        95
## sum      1131667      1031509   1027347   883213 839847    977487
##          Finishing Free.kick.accuracy GK.diving GK.handling GK.kicking
## nbr.val      17733              17733     17733       17733      17733
## nbr.null         0                  0         0           0          0
## nbr.na           0                  0         0           0          0
## min              2                  4         1           1          1
## max             95                 93        91          91         95
## range           93                 89        90          90         94
## sum         804214             766012    295910      292005     289805
##          GK.positioning GK.reflexes Heading.accuracy Interceptions Jumping
## nbr.val           17733       17733            17733         17733   17733
## nbr.null              0           0                0             0       0
## nbr.na                0           0                0             0       0
## min                   1           1                4             4      13
## max                  91          90               94            92      95
## range                90          89               90            88      82
```

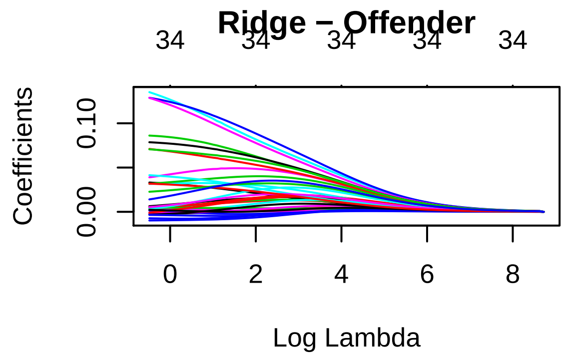
```
## sum             291702      298256           928418        825099 1151463
##          Long.passing Long.shots Marking Penalties Positioning Reactions
## nbr.val         17733      17733   17733     17733       17733     17733
## nbr.null            0          0       0         0           0         0
## nbr.na              0          0       0         0           0         0
## min                 7          3       4         5           2        28
## max                93         92      92        92          95        96
## range              86         89      88        87          93        68
## sum            930440     838018  781378    868886      881261   1097541
##          Short.passing Shot.power Sliding.tackle Sprint.speed Stamina
## nbr.val          17733      17733          17733        17733   17733
## nbr.null             0          0              0            0       0
## nbr.na               0          0              0            0       0
## min                 10          3              4           11      12
## max                 92         94             91           96      95
## range               82         91             87           85      83
## sum            1034568     987994         807675      1149939 1122094
##          Standing.tackle Strength Vision Volleys pos
## nbr.val            17733    17733  17733   17733  NA
## nbr.null               0        0      0       0  NA
## nbr.na                 0        0      0       0  NA
## min                    4       12     10       4  NA
## max                   92       98     94      91  NA
## range                 88       86     84      87  NA
## sum               840869  1158244 940518  767237  NA
```

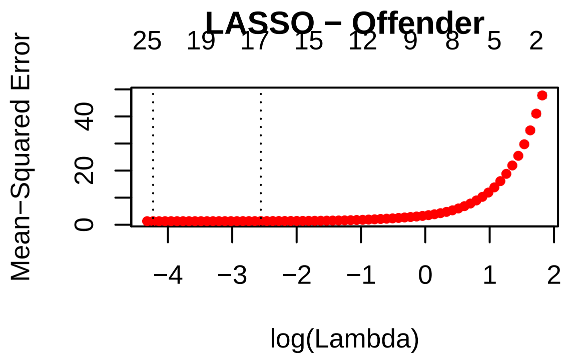
(APX 2) Heatmap with Hierarchical Cluster Option

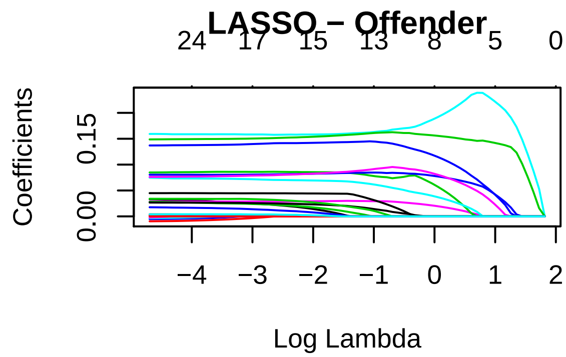


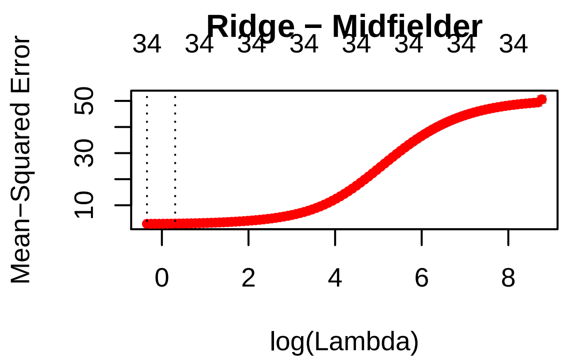
(APX 3) MSE and Ridge/LASSo Plots for All Positions

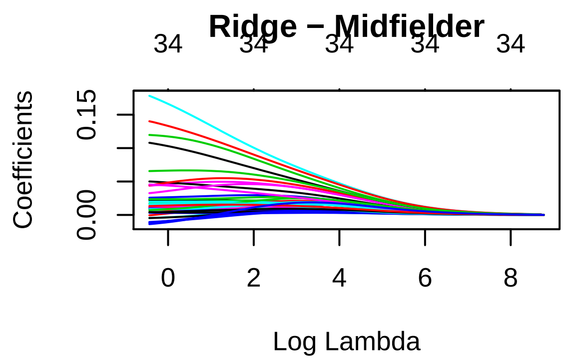


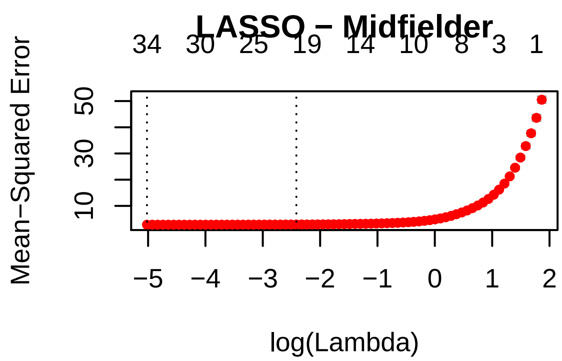


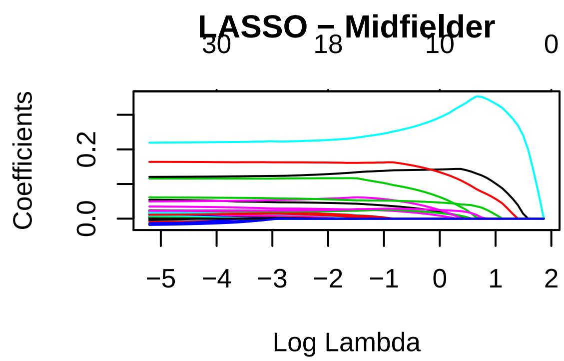


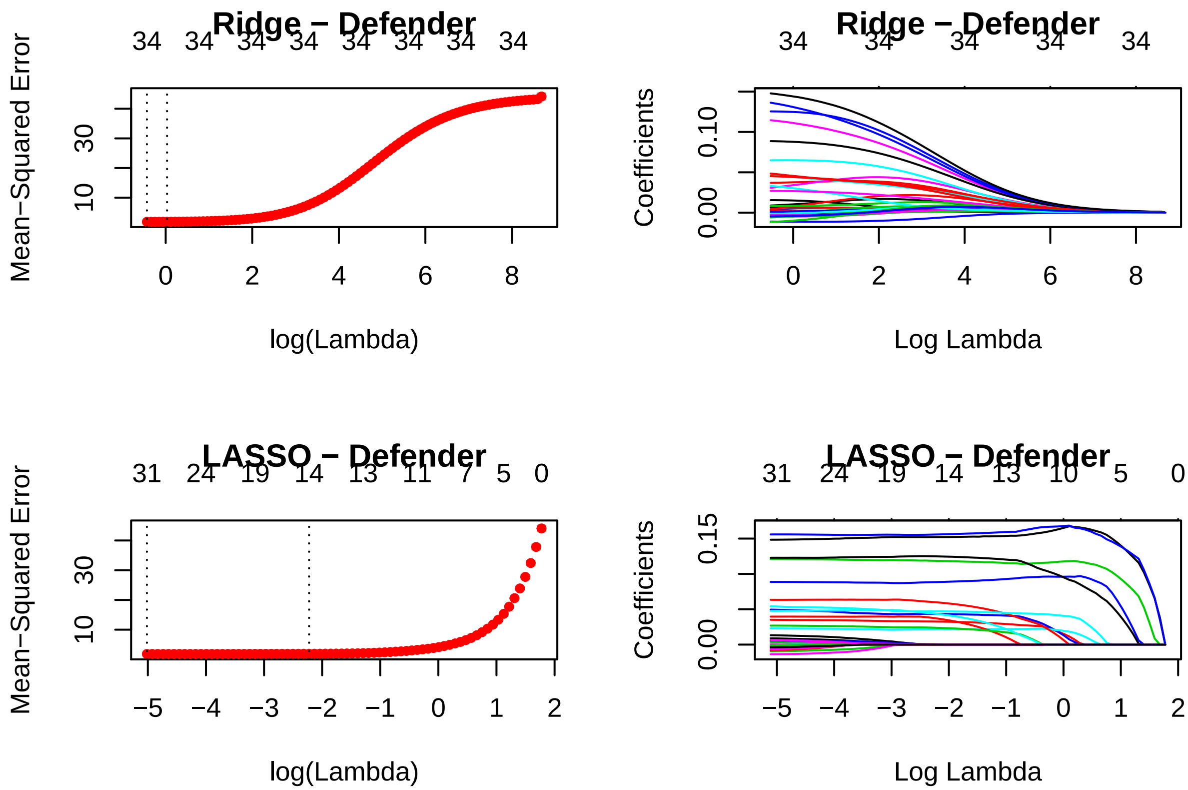




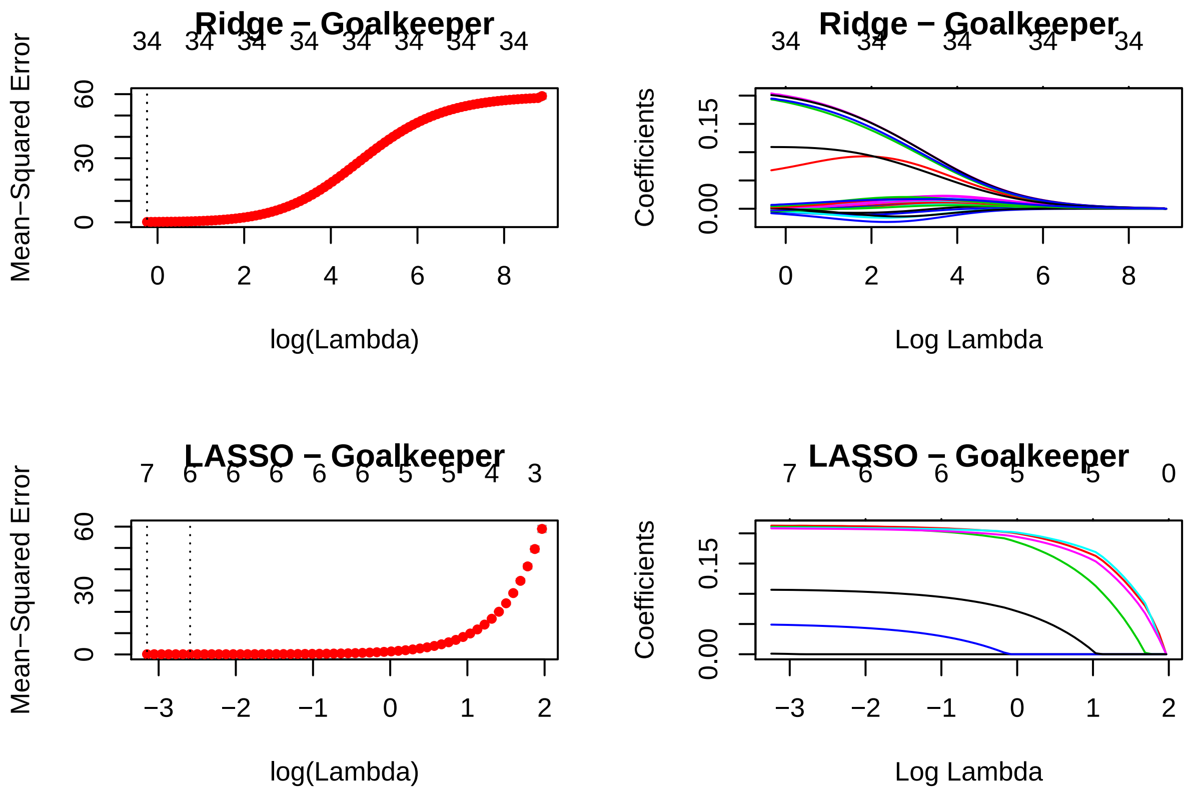








Ridge − Defender
34 34 34 34 34 34 34 34
Mean−Squared Error
10
30
0 2 4 6 8
log(Lambda)
Ridge − Defender
34 34 34 34 34
Coefficients
0.00 0.10
0 2 4 6 8
Log Lambda
LASSO − Defender
31 24 19 14 13 11 7 5 0
Mean−Squared Error
10
30
−5 −4 −3 −2 −1 0 1 2
log(Lambda)
LASSO − Defender
31 24 19 14 13 10 5 0
Coefficients
0.00 0.15
−5 −4 −3 −2 −1 0 1 2
Log Lambda




```
##      Position     Ridge MSE            LASSO MSE
## [1,] "Offender"   "1.41573038377561"   "1.35285133431696"
## [2,] "Midfielder" "3.18781288283597"   "3.16431769082251"
## [3,] "Defender"   "1.76592913603406"   "1.77251890806939"
## [4,] "Goalkeeper" "0.177825541163576"  "0.122562095555412"
```