Major Leagues - Caio Vigo Pereira

Loading the data

I choose to analyze the soccer datasets.

```
my_path<-"C:/Users/Caio Laptop/OneDrive - The University of Kansas/Documents/PhD/11. Courses/19. EECS 73
setwd(my_path)

spi_matches<-read.csv(paste(my_path,"/Datasets/spi_matches.csv", sep=""),header=T)
spi_global_rankings_intl<-read.csv(paste(my_path,"/Datasets/spi_global_rankings_intl.csv", sep=""),heade
spi_global_rankings<-read.csv(paste(my_path,"/Datasets/spi_global_rankings.csv", sep=""),header=T)
```

Information from the source

SPI Ratings This file contains links to the data behind our Club Soccer Predictions and Global Club Soccer Rankings. spi_matches.csv contains match-by-match SPI ratings and forecasts back to 2016. spi__global__rankings.csv contains current SPI ratings and rankings for men's club teams. spi__global__rankings__intl.csv contains current SPI ratings and rankings for men's international teams.

Analyzing my 3 datasets

```
names(spi_matches)
```

```
##  [1] "date"        "league_id"   "league"      "team1"       "team2"
##  [6] "spi1"        "spi2"        "prob1"       "prob2"       "probtie"
## [11] "proj_score1" "proj_score2" "importance1" "importance2" "score1"
## [16] "score2"      "xg1"         "xg2"         "nsxg1"       "nsxg2"
## [21] "adj_score1"  "adj_score2"
```

```
head(spi_matches)
```

```
##         date league_id                   league         team1
## 1 2016-08-12      1843           French Ligue 1        Bastia
## 2 2016-08-12      1843           French Ligue 1     AS Monaco
## 3 2016-08-13      2411 Barclays Premier League     Hull City
## 4 2016-08-13      2411 Barclays Premier League       Burnley
## 5 2016-08-13      2411 Barclays Premier League Middlesbrough
## 6 2016-08-13      2411 Barclays Premier League   Southampton
##                 team2  spi1  spi2  prob1  prob2 probtie proj_score1
## 1 Paris Saint-Germain 51.16 85.68 0.0463 0.8380  0.1157        0.91
## 2            Guingamp 68.85 56.48 0.5714 0.1669  0.2617        1.82
## 3      Leicester City 53.57 66.81 0.3459 0.3621  0.2921        1.16
## 4        Swansea City 58.98 59.74 0.4482 0.2663  0.2854        1.37
## 5          Stoke City 56.32 60.35 0.4380 0.2692  0.2927        1.30
## 6             Watford 69.49 59.33 0.5759 0.1874  0.2367        1.91
##   proj_score2 importance1 importance2 score1 score2  xg1  xg2 nsxg1 nsxg2
## 1        2.36        32.4        67.7      0      1 0.97 0.63  0.43  0.45
## 2        0.86        53.7        22.9      2      2 2.45 0.77  1.75  0.42
## 3        1.24        38.1        22.2      2      1 0.85 2.77  0.17  1.25
## 4        1.05        36.5        29.1      0      1 1.24 1.84  1.71  1.56
## 5        1.01        33.9        32.5      1      1 1.40 0.55  1.13  1.06
## 6        1.05        34.1        30.7      1      1 1.05 0.22  1.52  0.41
##   adj_score1 adj_score2
## 1       0.00       1.05
## 2       2.10       2.10
## 3       2.10       1.05
## 4       0.00       1.05
## 5       1.05       1.05
## 6       1.05       1.05
```

```r
dim(spi_matches)
```

```
## [1] 20879    22
```

```r
typeof(spi_matches)
```

```
## [1] "list"
```

```r
str(spi_matches)
```

```
## 'data.frame':    20879 obs. of  22 variables:
##  $ date       : Factor w/ 839 levels "2016-08-12","2016-08-13",..: 1 1 2 2 2 2 2 2 2 2 ...
##  $ league_id  : int  1843 1843 2411 2411 2411 2411 2411 2411 1843 2411 ...
##  $ league     : Factor w/ 37 levels "Argentina Primera Division",..: 13 13 4 4 4 4 4 4 13 4 ...
##  $ team1      : Factor w/ 698 levels "1. FC Heidenheim 1846",..: 78 50 319 120 406 581 213 180 103 39
##  $ team2      : Factor w/ 698 levels "1. FC Heidenheim 1846",..: 473 295 369 613 604 682 634 685 593
##  $ spi1       : num  51.2 68.8 53.6 59 56.3 ...
##  $ spi2       : num  85.7 56.5 66.8 59.7 60.4 ...
##  $ prob1      : num  0.0463 0.5714 0.3459 0.4482 0.438 ...
##  $ prob2      : num  0.838 0.167 0.362 0.266 0.269 ...
##  $ probtie    : num  0.116 0.262 0.292 0.285 0.293 ...
##  $ proj_score1: num  0.91 1.82 1.16 1.37 1.3 1.91 1.47 1.35 1.39 2.69 ...
##  $ proj_score2: num  2.36 0.86 1.24 1.05 1.01 1.05 1.38 1.14 1.14 0.48 ...
##  $ importance1: num  32.4 53.7 38.1 36.5 33.9 34.1 31.9 43.6 37.9 73 ...
##  $ importance2: num  67.7 22.9 22.2 29.1 32.5 30.7 48 34.6 44.2 27 ...
##  $ score1     : int  0 2 2 0 1 1 1 0 3 2 ...
##  $ score2     : int  1 2 1 1 1 1 1 1 2 1 ...
##  $ xg1        : num  0.97 2.45 0.85 1.24 1.4 1.05 0.73 1.11 1.03 2.14 ...
##  $ xg2        : num  0.63 0.77 2.77 1.84 0.55 0.22 1.11 0.68 1.84 1.25 ...
##  $ nsxg1      : num  0.43 1.75 0.17 1.71 1.13 1.52 0.88 0.84 1.1 1.81 ...
##  $ nsxg2      : num  0.45 0.42 1.25 1.56 1.06 0.41 1.81 1.6 2.26 0.92 ...
##  $ adj_score1 : num  0 2.1 2.1 0 1.05 1.05 1.05 0 3.12 2.1 ...
##  $ adj_score2 : num  1.05 2.1 1.05 1.05 1.05 1.05 1.05 1.05 2.1 1.05 ...
```

```r
names(spi_global_rankings_intl)
```

```
## [1] "rank"   "name"   "confed" "off"    "def"    "spi"
```

```r
head(spi_global_rankings_intl)
```

```
##   rank      name   confed  off  def   spi
## 1    1    Brazil CONMEBOL 3.11 0.29 92.96
## 2    2     Spain     UEFA 3.46 0.48 92.54
## 3    3   Belgium     UEFA 3.06 0.54 89.10
## 4    4    France     UEFA 2.84 0.46 88.57
## 5    5   Germany     UEFA 2.96 0.56 87.93
## 6    6 Argentina CONMEBOL 2.57 0.49 85.53
```

```
dim(spi_global_rankings_intl)
```

```
## [1] 213   6
```

```
typeof(spi_global_rankings_intl)
```

```
## [1] "list"
```

```
str(spi_global_rankings_intl)
```

```
## 'data.frame':    213 obs. of  6 variables:
##  $ rank  : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ name  : Factor w/ 213 levels "Afghanistan",..: 28 175 19 68 74 8 61 150 133 203 ...
##  $ confed: Factor w/ 6 levels "AFC","CAF","CONCACAF",..: 4 6 6 6 6 4 6 6 6 4 ...
##  $ off   : num  3.11 3.46 3.06 2.84 2.96 2.57 2.32 2.38 2.55 2.3 ...
##  $ def   : num  0.29 0.48 0.54 0.46 0.56 0.49 0.51 0.56 0.68 0.54 ...
##  $ spi   : num  93 92.5 89.1 88.6 87.9 ...
```

```
names(spi_global_rankings)
```

```
## [1] "rank"      "prev_rank" "name"      "league"    "off"       "def"      
## [7] "spi"
```

```
head(spi_global_rankings)
```

```
##   rank prev_rank                name                   league  off  def
## 1    1         1     Manchester City  Barclays Premier League 2.92 0.20
## 2    2         3           Barcelona Spanish Primera Division 3.12 0.38
## 3    3         4         Real Madrid Spanish Primera Division 2.99 0.38
## 4    4         2       Bayern Munich        German Bundesliga 2.94 0.40
## 5    5         6            Juventus            Italy Serie A 2.66 0.29
## 6    6         7 Paris Saint-Germain           French Ligue 1 3.09 0.49
##     spi
## 1 93.78
## 2 92.41
## 3 91.75
## 4 90.93
## 5 90.72
## 6 90.70
```

```
dim(spi_global_rankings)
```

```
## [1] 628   7
```

```
typeof(spi_global_rankings)
```

```
## [1] "list"
```

```
str(spi_global_rankings)
```

```
## 'data.frame':    628 obs. of  7 variables:
##  $ rank     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ prev_rank: int  1 3 4 2 6 7 5 8 9 10 ...
##  $ name     : Factor w/ 628 levels "1. FC Heidenheim 1846",..: 357 64 461 70 302 428 338 57 136 573
##  $ league   : Factor w/ 35 levels "Argentina Primera Division",..: 4 28 28 16 18 13 4 28 4 4 ...
##  $ off      : num  2.92 3.12 2.99 2.94 2.66 3.09 2.66 2.21 2.52 2.42 ...
##  $ def      : num  0.2 0.38 0.38 0.4 0.29 0.49 0.3 0.26 0.45 0.52 ...
##  $ spi      : num  93.8 92.4 91.8 90.9 90.7 ...
```

Loading some packages

```
library(stargazer)
```

```
##
## Please cite as:
##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
library(Amelia) # for missmap() function
```

```
## Loading required package: Rcpp
## ##
## ## Amelia II: Multiple Imputation
## ## (Version 1.7.5, built: 2018-05-07)
## ## Copyright (C) 2005-2018 James Honaker, Gary King and Matthew Blackwell
## ## Refer to http://gking.harvard.edu/amelia/ for more information
## ##
```

Taking a look on the Descriptive Statistics

```
stargazer(spi_matches, type='text', align=TRUE, digits=2)
```

```
##
## ====================================================================
## Statistic     N      Mean   St. Dev.  Min  Pctl(25) Pctl(75)  Max
## --------------------------------------------------------------------
## league_id   20,879 2,141.44  738.39  1,818  1,849    2,160   5,641
## spi1        20,879   46.57   18.62   5.23   33.04    59.48   96.57
## spi2        20,879   46.53   18.61   4.97   33.04    59.41   96.78
## prob1       20,879   0.45     0.16   0.03    0.35     0.54    0.98
## prob2       20,879   0.29     0.14   0.00    0.20     0.36    0.88
## probtie     20,879   0.26     0.05   0.00    0.24     0.28    0.45
## proj_score1 20,879   1.52     0.43   0.25    1.24     1.72    4.03
## proj_score2 20,879   1.14     0.42   0.20    0.88     1.36    3.42
## importance1 10,515   30.79   25.35   0.00   10.90    44.60    100.00
## importance2 10,515   30.12   25.03   0.00   10.50    43.60    100.00
## score1      14,315   1.54     1.28   0.00    1.00     2.00    8.00
## score2      14,315   1.17     1.14   0.00    0.00     2.00    8.00
## xg1         8,664    1.47     0.83   0.00    0.85     1.94    7.04
## xg2         8,664    1.12     0.72   0.00    0.58     1.50    6.20
## nsxg1       8,664    1.40     0.65   0.00    0.95     1.74    6.58
## nsxg2       8,664    1.12     0.57   0.00    0.72     1.42    5.92
## adj_score1  8,664    1.55     1.26   0.00    1.05     2.10    7.97
## adj_score2  8,664    1.17     1.12   0.00    0.00     2.10    6.76
## --------------------------------------------------------------------
```

```
stargazer(spi_global_rankings, type='text', align=TRUE, digits=2)
```

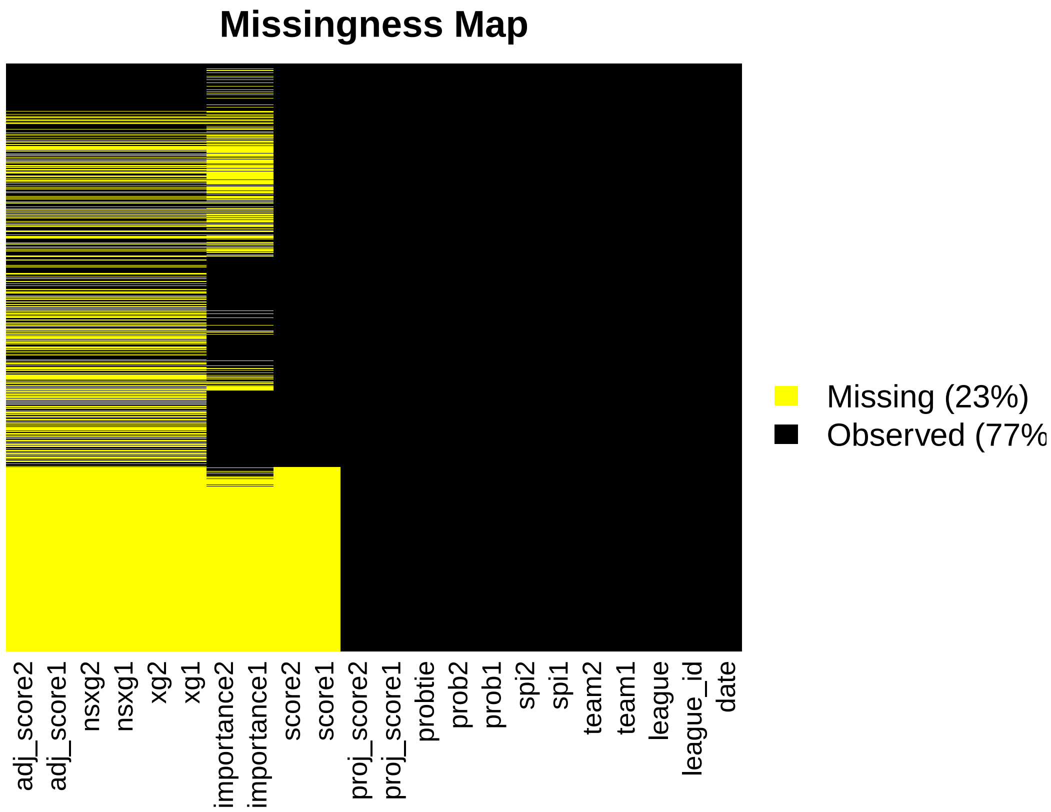
```
##
## ==========================================================
## Statistic  N   Mean  St. Dev. Min  Pctl(25) Pctl(75)  Max
## ----------------------------------------------------------
## rank      628 314.50  181.43   1    157.8    471.2    628
## prev_rank 628 314.50  181.43   1    157.8    471.2    628
## off       628  1.26   0.49   0.20   0.96     1.53   3.12
## def       628  1.39   0.44   0.20   1.09     1.68   2.84
## spi       628 42.99   18.08   4.97  29.98    55.66   93.78
## -----------------------------------------------------------
```

```
stargazer(spi_global_rankings_intl, type='text', align=TRUE, digits=2)
```

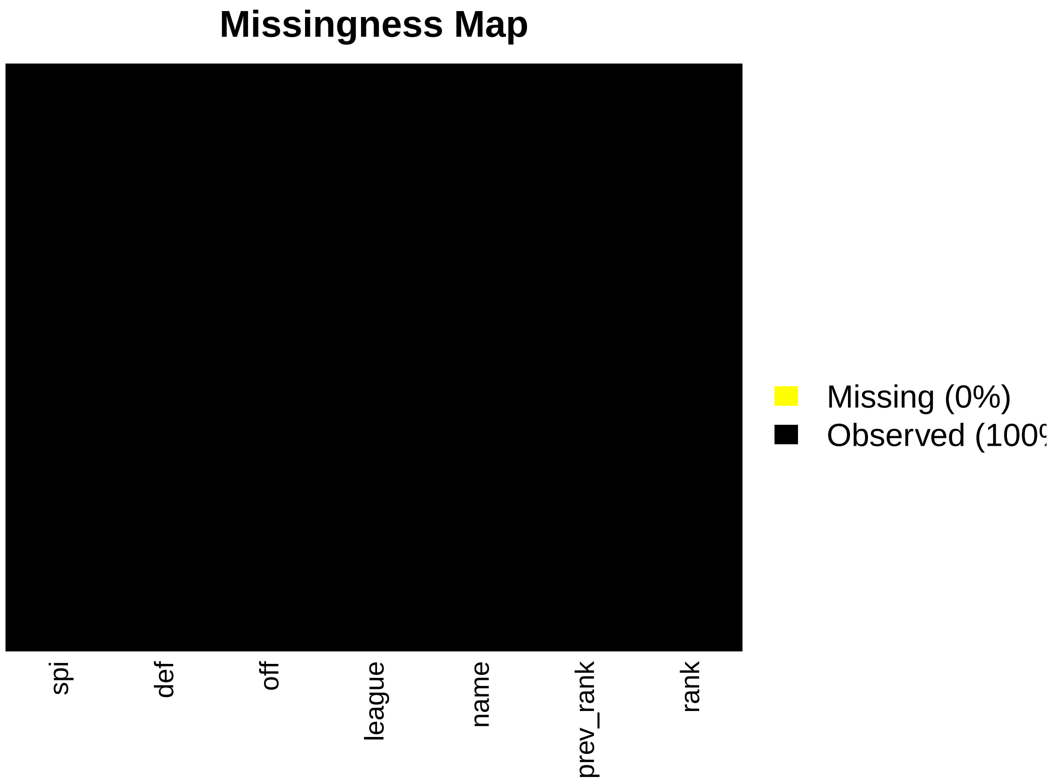
```
##
## ===========================================================
## Statistic  N   Mean  St. Dev. Min  Pctl(25) Pctl(75)  Max
## -----------------------------------------------------------
## rank      213 107.00  61.63    1      54      160      213
## off       213  1.17   0.65   0.20   0.67     1.57   3.46
## def       213  1.64   1.10   0.29   0.93     1.89   6.08
## spi       213 39.91   24.45   0.26  19.84    59.64   92.96
## -----------------------------------------------------------
```

Checking for any NA's in the dataframe.

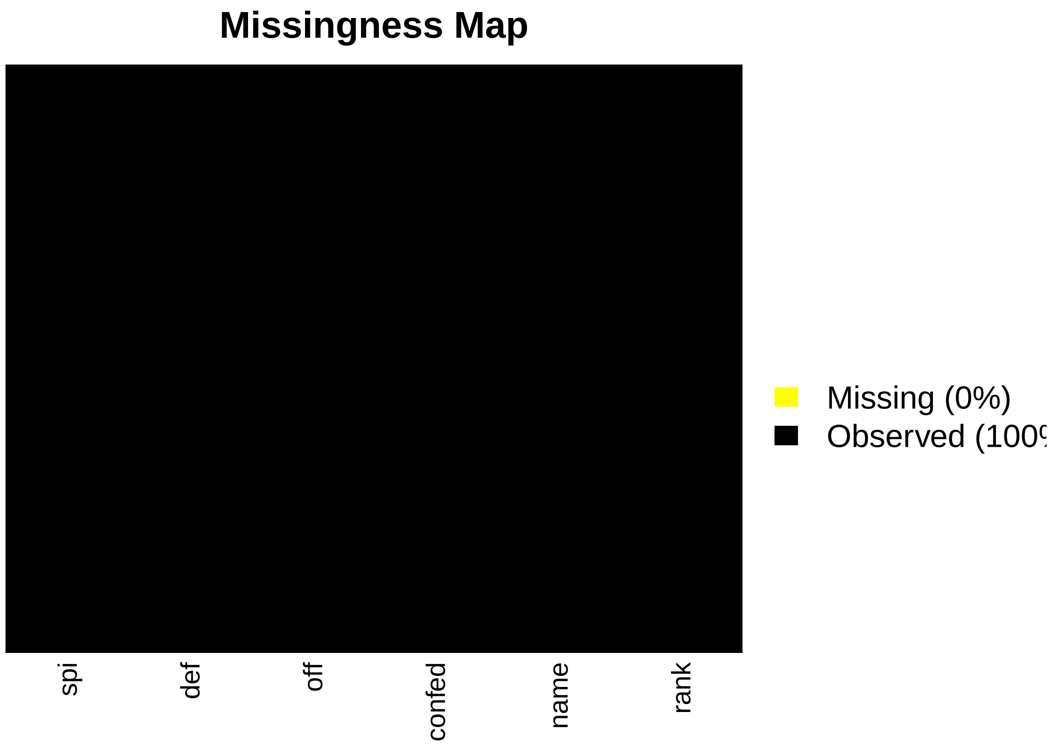
```
missmap(spi_matches,col=c('yellow','black'),y.at=1,y.labels='',legend=TRUE)
```



```
missmap(spi_global_rankings,col=c('yellow','black'),y.at=1,y.labels='',legend=TRUE)
```



```
missmap(spi_global_rankings_intl,col=c('yellow','black'),y.at=1,y.labels='',legend=TRUE)
```

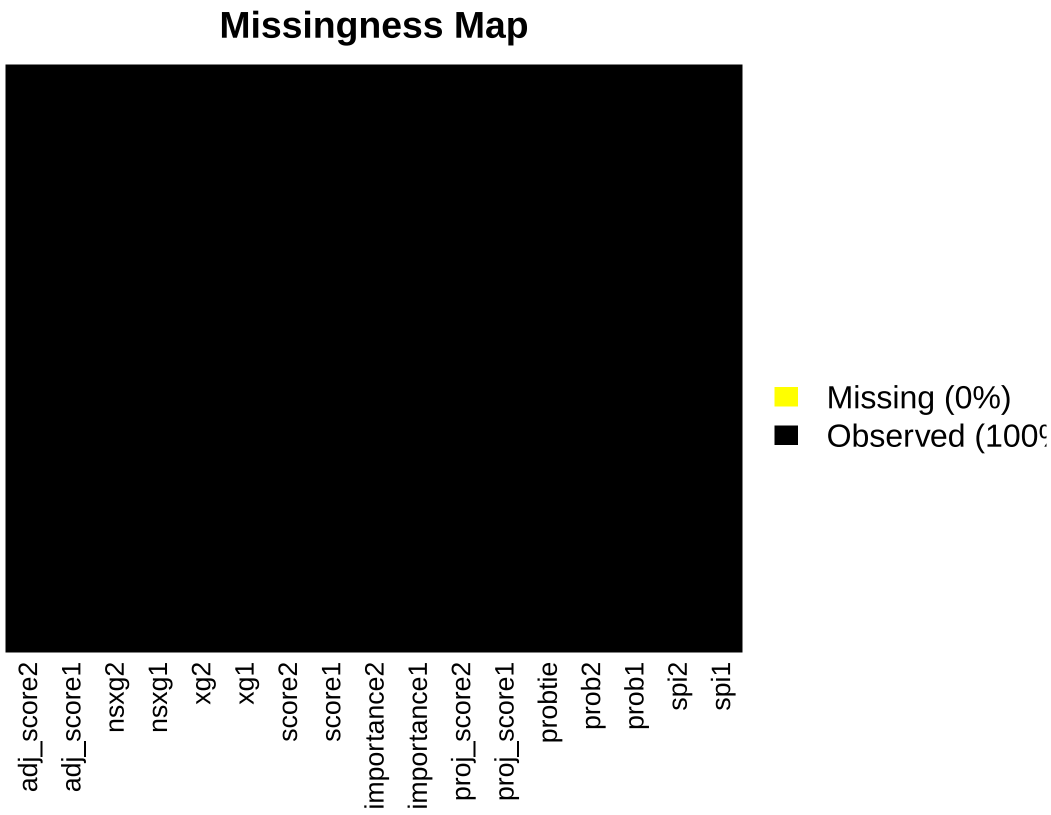


Replacing missing values with the mean

```r
spi_matches<-spi_matches[,6:22]
for(i in 1:ncol(spi_matches)){
  spi_matches[is.na(spi_matches[,i]), i] <- mean(spi_matches[,i], na.rm = TRUE)
}
```

Checking if there is any missing value after the changes

```r
missmap(spi_matches,col=c('yellow','black'),y.at=1,y.labels='',legend=TRUE)
```



```r
#set a seed
set.seed(999)

library(caTools)
#Split the data , `split()` assigns a booleans to a new column based on the SplitRatio specified.

split <- sample.split(spi_matches,SplitRatio =0.75)


train <- subset(spi_matches,split==TRUE)
test <- subset(spi_matches,split==FALSE)
```

```r
team_1_model <- lm(score1 ~ proj_score1 + importance1 + xg1 + nsxg1 + spi1 + prob1, data=train)
team_2_model <- lm(score2 ~ proj_score2 + importance2 + xg2 + nsxg2 + spi2 + prob2, data=train)
# summary(team_1_model)
# summary(team_2_model)
stargazer(team_1_model, type='text', align=TRUE, digits=2)
```

```
##
## ===============================================
##                          Dependent variable:
##                      ---------------------------
##                              score1
## -----------------------------------------------
## proj_score1                  0.35***
##                              (0.04)
##
```

```
## importance1                   0.0003
##                              (0.0004)
##
## xg1                           0.93***
##                               (0.02)
##
## nsxg1                        -0.40***
##                               (0.02)
##
## spi1                          -0.0003
##                              (0.0005)
##
## prob1                         0.37***
##                               (0.11)
##
## Constant                       0.03
##                               (0.04)
##
## -----------------------------------------------
## Observations                  14,739
## R2                             0.24
## Adjusted R2                    0.24
## Residual Std. Error      0.92 (df = 14732)
## F Statistic          773.89*** (df = 6; 14732)
## ===============================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
```

```
stargazer(team_2_model, type='text', align=TRUE, digits=2)
```

```
##
## ===============================================
##                         Dependent variable:
##                     ---------------------------
##                               score2
## -----------------------------------------------
## proj_score2                   0.28***
##                               (0.04)
##
## importance2                   0.0000
##                              (0.0004)
##
## xg2                           0.99***
##                               (0.02)
##
## nsxg2                        -0.33***
##                               (0.02)
##
## spi2                          -0.0001
##                              (0.0004)
##
## prob2                         0.33**
##                               (0.13)
##
## Constant                       0.02
##                               (0.03)
```

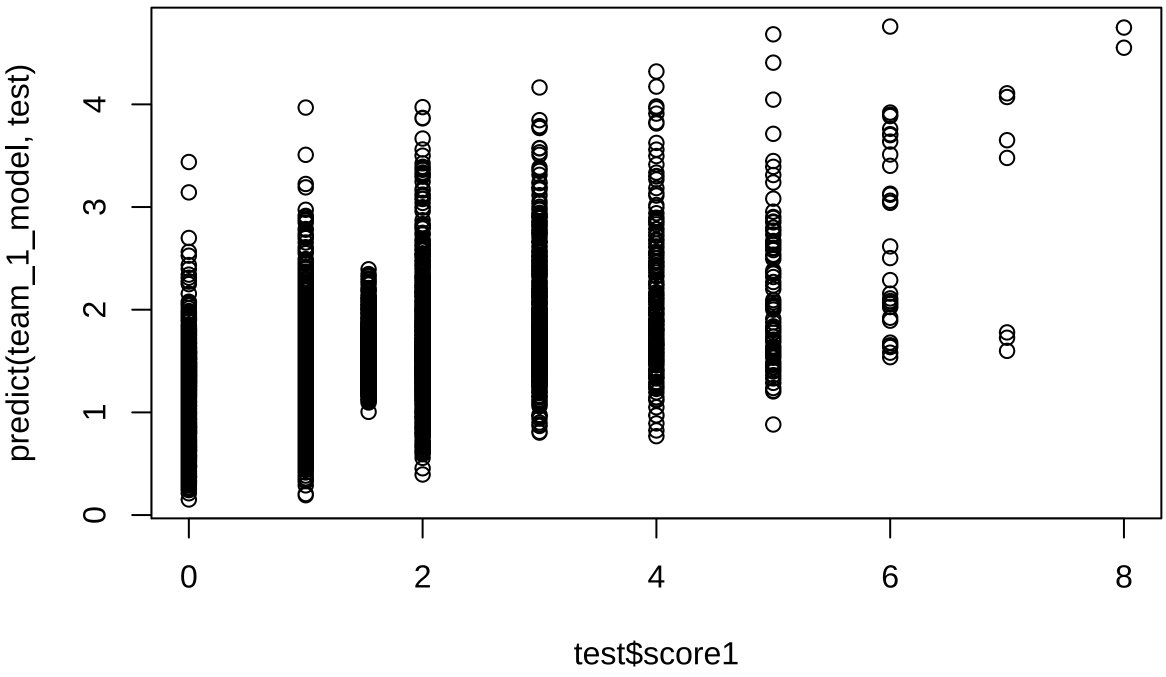
```
##
## -----------------------------------------------
## Observations                    14,739
## R2                               0.24
## Adjusted R2                      0.24
## Residual Std. Error        0.82 (df = 14732)
## F Statistic            775.62*** (df = 6; 14732)
## ===============================================
## Note:                 *p<0.1; **p<0.05; ***p<0.01
```

```
# test$predicted.medv <- predict(team_1_model,test)
# test<- na.omit(test$predicted.medv)
```

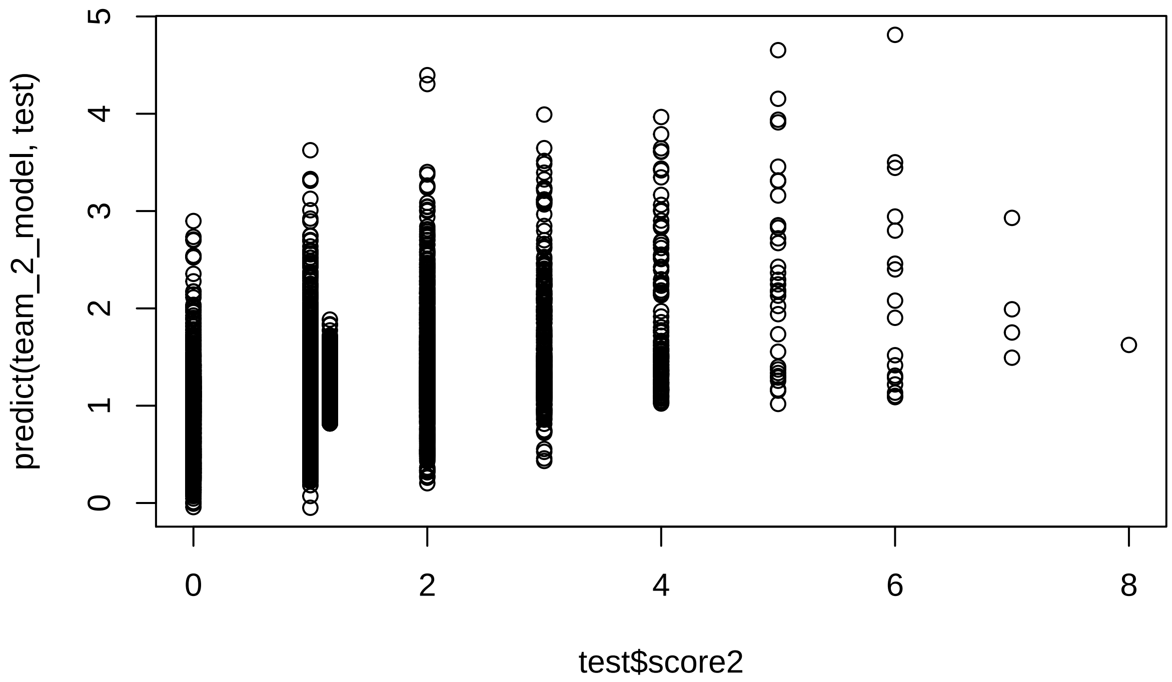
Predictions plots

```
predict_team_1_model <- predict(team_1_model,test)
predict_team_2_model <- na.omit(predict(team_2_model,test))

plot(test$score1,predict(team_1_model,test))
```



```
plot(test$score2,predict(team_2_model,test))
```



```
error <- test$score1-predict_team_1_model
rmse <- sqrt(mean(error)^2)
rmse
```

```
## [1] 0.008974758
```

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Ctrl+Alt+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Ctrl+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.