Untitled

Caoxin Mao

3/16/2022

```r
library(readxl)
library(dplyr)
```

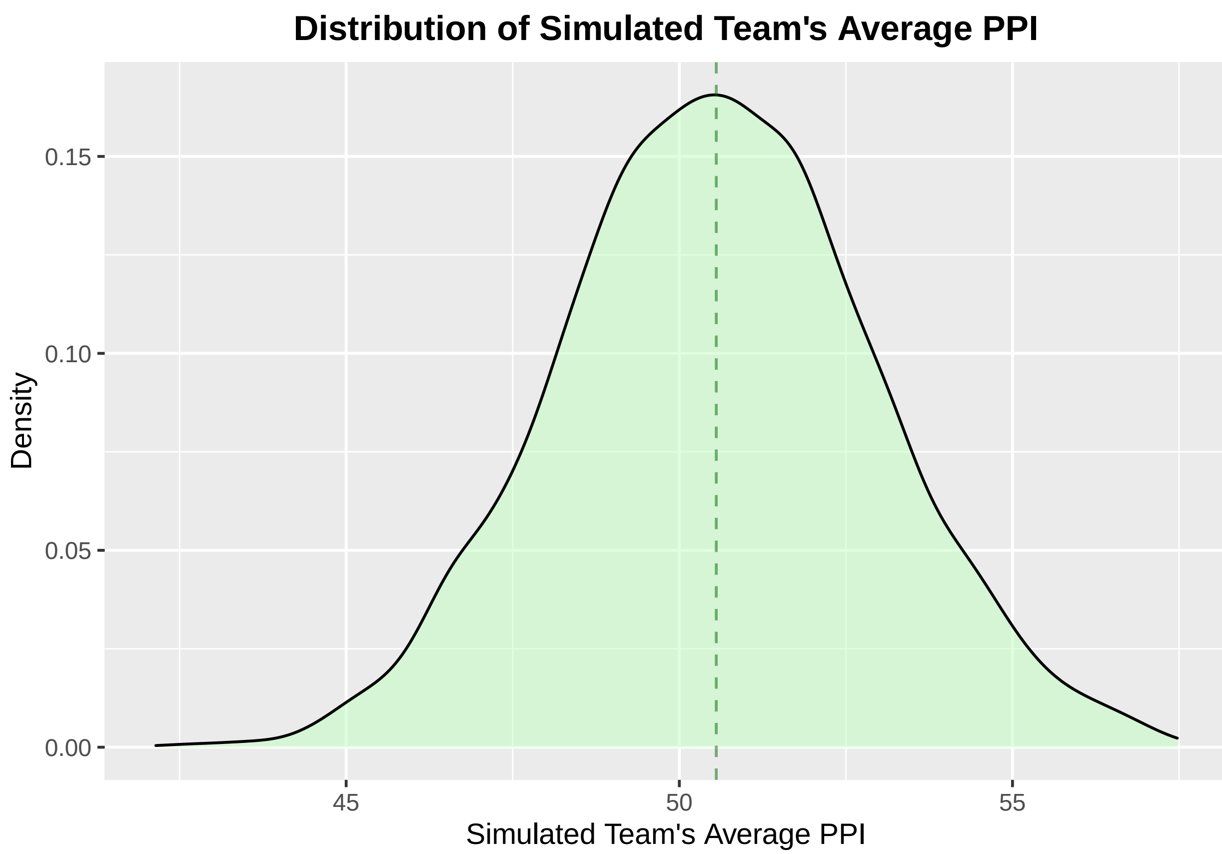
```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(ggplot2)
lea_noGK <- read_excel('Poisson Loves Exponential 2022-student-research-case-study-player-data 202234.xl
                       sheet = "league without GK")
lea_GK <- read_excel('Poisson Loves Exponential 2022-student-research-case-study-player-data 202234.xls
                     sheet = "league GK")
tour_noGK <- read_excel('Poisson Loves Exponential 2022-student-research-case-study-player-data 202234.
                        sheet = "tournament without GK")
tour_GK <- read_excel('Poisson Loves Exponential 2022-student-research-case-study-player-data 202234.xls
                      sheet = "tournament GK")

PPIsc <- read_excel('Poisson Loves Exponential 2022-student-research-case-study-player-data 202234.xlsx
                    sheet = "PPI scheme")

Salary <- read_excel('Poisson Loves Exponential 2022-student-research-case-study-player-data 202234.xls
                     sheet = "2021 Salaries")

inflation <- read_excel('Poisson Loves Exponential 2022-student-research-case-study-economic-data.xlsx'
                        sheet = "Rarita Inflation Rates")

reexpop <- read_excel('Poisson Loves Exponential 2022-student-research-case-study-football-soccer-data.
                      sheet = "reexpop data")
```

Create PPI function used to calculate PPI

```r
PPI <- function(x){
  nr <- nrow(x)
  nc <- ncol(x)
  if (is.element("GK",x$Pos)){
    n_cri <- 3
  } else {
    n_cri <- 13
  }
  n<-nc-n_cri+1

  ### single criteria PPI
  for (j in c(n:nc)) {
    x[,j] <- order(x[,j])/nr*100
  }

  ### overall PPI
  overall_PPI <- c()
  for (i in c(1:nr)) {
    pos_index <- which(names(PPIsc)==x$Pos[i])
    overall_PPI[i] <- as.numeric(as.matrix(x[i,n:nc])%*%as.matrix(PPIsc[1:n_cri,pos_index]))
  }

  ### delete single criteria keep the information and overall PPI
  x <- x[,-(n:nc)]
  x$overall_PPI <- overall_PPI

  ### delete complicated player using highest overall PPI
  del_index_all <- c()
  dup_player <- x$Player[duplicated(x$Player)]
  if (length(dup_player) > 0) {
    for (i in dup_player) {
      dup_index <- which(x$Player==i)
      bigger_index <- dup_index[which(x[dup_index,n] == max(x[dup_index,n]))]
      del_index <- dup_index[-which(dup_index == bigger_index)]
      del_index_all <- append(del_index_all,del_index)
    }
    x <- x[-del_index_all,]
  }

  ### add highest salary to each player
  player_salary <- c()
  for (p in 1:nrow(x)) {
    player_index <- which(Salary$`Player Name` == x$Player[p])
    salary <- max(Salary$`Annualized Salary`[player_index])
    if (x$Nation[p] != "Rarita"){
      salary <- (1+0.1)*salary ### adjust the foreign player salary with 10% loan rate
    }
    player_salary <- append(player_salary,salary)
  }
  x$salary <- player_salary
  x$salaryefficiency <- x$overall_PPI*(10^6)/x$salary

  ### return result
```

```r
  return(x)
}


############################################# calculate the PPI
options(warn=-1)
lea_GK_PPI <- PPI(lea_GK)
lea_noGK_PPI <- PPI(lea_noGK)
tour_GK_PPI <- PPI(tour_GK)
tour_noGK_PPI <- PPI(tour_noGK)
options(warn=1)


############################################# find benchmark distribution to figure out the probablity
can_FW <- lea_noGK_PPI %>%
  filter(Pos == "FW" | Pos=="FWMF" | Pos=="FWDF") %>%
  filter(Nation=="Sobianitedrucy"|Nation=="People's Land of Maneau"|Nation=="Nganion"|Nation=="Mico"|
           Nation=="Southern Ristan"|Nation=="Dosqaly")

can_MF <- lea_noGK_PPI %>%
  filter(Pos == "MF" | Pos=="MFFW" | Pos=="MFDF")%>%
  filter(Nation=="Sobianitedrucy"|Nation=="People's Land of Maneau"|Nation=="Nganion"|Nation=="Mico"|
           Nation=="Southern Ristan"|Nation=="Dosqaly")

can_DF <- lea_noGK_PPI %>%
  filter(Pos == "DF" | Pos=="DFFW" | Pos=="DFMF")%>%
  filter(Nation=="Sobianitedrucy"|Nation=="People's Land of Maneau"|Nation=="Nganion"|Nation=="Mico"|
           Nation=="Southern Ristan"|Nation=="Dosqaly")

simed_benchmark <- c()


set.seed(10)
for (i in c(1:5000)) {
  sim_FW <- sum(sample(can_FW$overall_PPI,6,replace = FALSE))
  sim_MF <- sum(sample(can_MF$overall_PPI,6,replace = FALSE))
  sim_DF <- sum(sample(can_DF$overall_PPI,8,replace = FALSE))
  sim_GK <- sum(sample(tour_GK_PPI$overall_PPI,3,replace = FALSE))
  sim_res <- sum(c(sim_FW,sim_MF,sim_DF,sim_GK))/23
  simed_benchmark <- append(simed_benchmark, sim_res)
}

simed_benchmark <- data.frame(sim_team_score = simed_benchmark)

p <- ggplot(data = simed_benchmark, mapping = aes(x = sim_team_score)) +
  geom_density(stat="density",fill="darkseagreen1", alpha = 0.5) +
  geom_vline(xintercept = mean(simed_benchmark$sim_team_score),
             linetype = 2, alpha = 0.5, color = 'darkgreen') +
  ggtitle("Distribution of Simulated Team's Average PPI") +
  labs(x = "Simulated Team's Average PPI", y = "Density") +
  theme(plot.title = element_text(face = "bold",hjust=0.5))
p
```




```r
shapiro.test(simed_benchmark$sim_team_score)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  simed_benchmark$sim_team_score
## W = 0.9994, p-value = 0.09749
```

Team Selection

```r
mu <- mean(simed_benchmark$sim_team_score)
sigma <- sd(simed_benchmark$sim_team_score)


############################################# team selection
team_FW <- lea_noGK_PPI %>%
  filter(Pos == "FW" | Pos=="FWMF" | Pos=="FWDF") %>%
  filter(salaryefficiency>quantile(salaryefficiency,0.5)) %>%
  filter(overall_PPI>quantile(overall_PPI,0.8)) %>%
  filter(overall_PPI>quantile(overall_PPI,0.97) | Nation =="Rarita")

team_MF <- lea_noGK_PPI %>%
  filter(Pos == "MF" | Pos=="MFFW" | Pos=="MFDF") %>%
```

```r
  filter(salaryefficiency>quantile(salaryefficiency,0.5)) %>%
  filter(overall_PPI>quantile(overall_PPI,0.8)) %>%
  filter(Nation =="Rarita")

team_DF <- lea_noGK_PPI %>%
  filter(Pos == "DF" | Pos=="DFFW" | Pos=="DFMF") %>%
  filter(salaryefficiency>quantile(salaryefficiency,0.5)) %>%
  filter(overall_PPI>quantile(overall_PPI,0.8)) %>%
  filter(Nation =="Rarita")

team_GK <- lea_GK_PPI %>%
  filter(salaryefficiency>quantile(salaryefficiency,0.5)) %>%
  filter(overall_PPI>quantile(overall_PPI,0.8)) %>%
  filter(Nation =="Rarita")

team <- data.frame(
  "Name"=c(team_FW$Player,team_MF$Player,team_DF$Player,team_GK$Player),
  "Nation"=c(team_FW$Nation,team_MF$Nation,team_DF$Nation,team_GK$Nation),
  "Position"=c(team_FW$Pos,team_MF$Pos,team_DF$Pos,team_GK$Pos),
  "Age"=c(team_FW$Age,team_MF$Age,team_DF$Age,team_GK$Age),
  "PPI"=c(team_FW$overall_PPI,team_MF$overall_PPI,team_DF$overall_PPI,team_GK$overall_PPI),
  "Salary"=c(team_FW$salary,team_MF$salary,team_DF$salary,team_GK$salary))
team
```

```
##             Name                     Nation Position Age      PPI   Salary
## 1      B. Maturu People's Land of Maneau       FW  28 76.81596 24607000
## 2       F. Adiru                  Dosqaly     FWMF  16 75.33938 28435000
## 3    Z. Nakiwala                   Rarita       FW  24 67.77797 24480000
## 4  A. Kyarikunda                   Rarita       FW  30 63.61714  8870000
## 5       B. Quaye                   Rarita       FW  22 64.27649 28510000
## 6   E. Nakanjako                   Rarita     MFFW  20 63.79485 10750000
## 7     M. Muhindo                   Rarita       MF  26 66.14521 28640000
## 8     D. Mattila                   Rarita       MF  26 64.18404 18500000
## 9        F. Chin                   Rarita       MF  22 76.60025  1340000
## 10     D. Kimuli                   Rarita       MF  28 67.37526  7580000
## 11      P. Rabiu                   Rarita     MFFW  27 67.82440  7280000
## 12       Y. Cheu                   Rarita       MF  18 66.56268  6190000
## 13     F. Yunusa                   Rarita       DF  26 69.41790 18150000
## 14     A. Núñez                  Rarita     DFMF  20 66.35796 22730000
## 15     C. Baluka                   Rarita       DF  20 68.86745  7760000
## 16       T. Audu                   Rarita       DF  27 61.84593  1360000
## 17     V. Sultan                   Rarita     DFMF  20 63.77374  5080000
## 18       R. Tsao                   Rarita       DF  26 63.12748  6530000
## 19      D. Naula                   Rarita       DF  24 62.75095  6560000
## 20         E. Ow                   Rarita       DF  26 66.82862  4780000
## 21    F. Ithungu                   Rarita       GK  27 75.91696  1530000
## 22      Y. Draru                   Rarita       GK  36 86.95502 11390000
## 23  X. Tumushabe                   Rarita       GK  31 78.54671  3390000
```

```r
# write.csv(team, "Team.csv")
pnorm(mean(team$PPI),mu,sigma)
```

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

Salary Projection

```r
t_inf <- lm(inflation$`Annual Inflation Rate`~inflation$T)
#summary(t_inf)
future_time_point <- c(30:39)
pre_inf <- t_inf$coefficients[1]+t_inf$coefficients[2]*future_time_point
future_salary <- sum(team$Salary)*cumprod(1+pre_inf)
future_salary
```

```
##  [1] 288941153 293151398 297057670 300645805 303902620 306815998 309374954
##  [8] 311569716 313391773 314833943
```

Revenue Projection

```r
t_re <- lm(reexpop$Revenue~reexpop$Time)
#summary(t_re)
ftp_re <- c(6:15)
pre_re <- t_re$coefficients[1]+t_re$coefficients[2]*ftp_re
pre_re
```

```
##  [1] 458.3087 468.2875 478.2663 488.2452 498.2240 508.2028 518.1817 528.1605
##  [9] 538.1393 548.1182
```

Expense Projection

```r
t_ex <- lm(reexpop$Expense~reexpop$Time)
#summary(t_ex)
ftp_ex <- c(6:15)
pre_ex <- t_ex$coefficients[1]+t_ex$coefficients[2]*ftp_ex
pre_ex
```

```
##  [1] 407.8440 428.2138 448.5837 468.9535 489.3233 509.6932 530.0630 550.4328
##  [9] 570.8027 591.1725
```

Population Projection

```r
t_pop <- lm(reexpop$`Rarita Pop`~reexpop$Time)
#summary(t_pop)
ftp_pop <- c(6:15)
pre_pop <- t_pop$coefficients[1]+t_pop$coefficients[2]*ftp_pop
pre_pop
```

```
##  [1] 12714051 12777270 12840488 12903706 12966925 13030143 13093361 13156580
##  [9] 13219798 13283016
```

Interest Projection

```r
t_int <- lm(reexpop$`1-year spot rate`~reexpop$Time)
#summary(t_int)
ftp_int <- c(6:15)
pre_int <- t_int$coefficients[1]+t_int$coefficients[2]*ftp_int
pre_int
```

```
##  [1] 0.039834 0.044157 0.048480 0.052803 0.057126 0.061449 0.065772 0.070095
##  [9] 0.074418 0.078741
```