Midterm

STAT 5400

Time: Nov 4, 2024. 6:30 - 8:30 PM

Submit your exam solutions in the form of a pdf and rmd file. Give detailed interpretation on every result you present. Show and only show relevant code. Organize your report.

Problem 1. Data import and visualization Install and load the `worldfootballR` package.

1. Use the R function `fb_season_team_stats` in the R package `worldfootballR` to access squad performance data for the English Premier League. Specify the arguments as follows: `country = "ENG"`, `gender = "M"`,`season_end_year = "2023"`, `tier = "1st"`, and `stat_type = "standard"`. The document of the R function `fb_season_team_stats` can be found at https://rdrr.io/github/JaseZiv/worldfootballR/man/fb_season_team_stats.html.

```
library(worldfootballR)

df <- fb_season_team_stats(
  country = "ENG",
  gender = "M",
  season_end_year = "2023",
  tier = "1st",
  stat_type = "standard"
  )
```

The extracted data contains 40 rows and 37 columns, capturing various performance metrics for the 20 Premier League teams. Each team has two records: one for *home games* and one for *away games*. In soccer, a *home game* refers to a match played at the team's own stadium; an *away game*, on the other hand, is played at the opponent's stadium, where the team does not have these advantages. For example, the first row in this dataset shows the team Arsenal's performance in their home games, while the 21st row records Arsenal's performance in away games.

To answer this question, please display the first six rows and the following three columns of the data:

- Column **5**: `Squad`
- Column **6**: `Team_or_Opponent` (`team` for home games, and `opponent` for away games).
- Column **14**: `Gls` (Total Goals)

```
cat("Printing first 6 rows and 3 specified columns:\n")
```

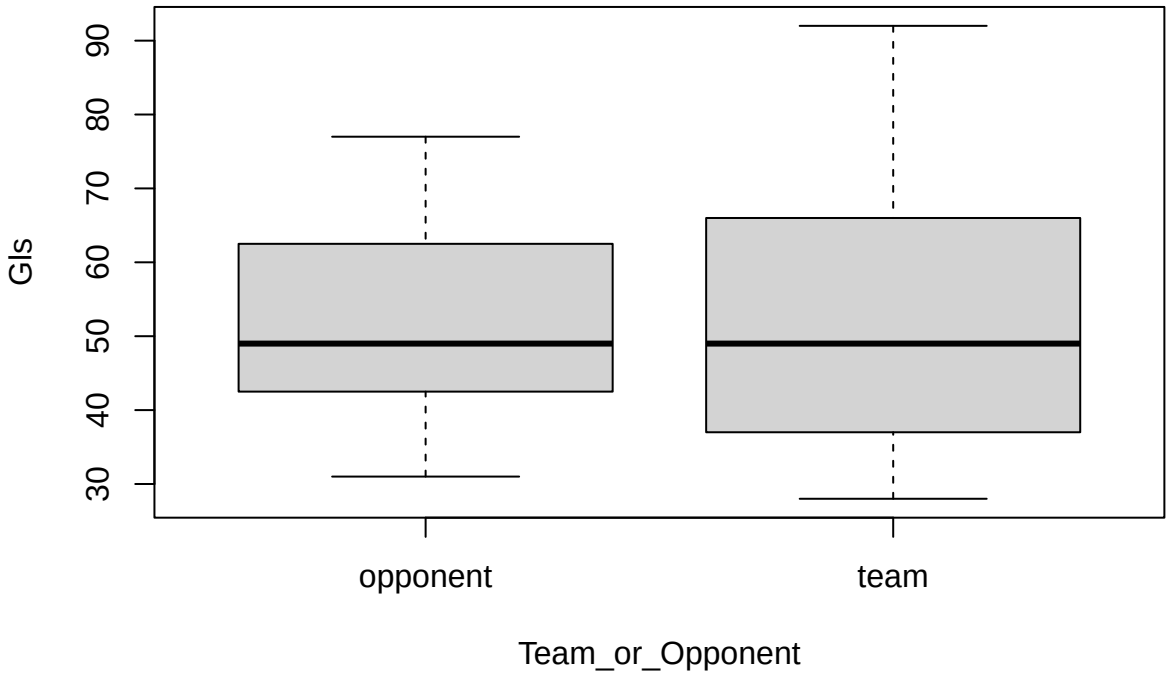
```
## Printing first 6 rows and 3 specified columns:
```

```r
df[1:6, c('Squad', 'Team_or_Opponent', 'Gls')]
```

```
##         Squad Team_or_Opponent Gls
## 1     Arsenal             team  84
## 2 Aston Villa             team  49
## 3 Bournemouth             team  37
## 4   Brentford             team  56
## 5    Brighton             team  68
## 6     Chelsea             team  37
```

2. Create side-by-side boxplots for the number of goals scored for all the teams; one boxplot for home games and the other for away games.

```r
par(mfrow=c(1,1))
boxplot(Gls ~ Team_or_Opponent, df)
```




```r
#boxplot(Gls ~ Team_or_Opponent, df[which(df$Team_or_Opponent == 'team'),])
#boxplot(Gls ~ Team_or_Opponent, df[which(df$Team_or_Opponent == 'opponent'),])
```

Teams appear to have a very slight home ground advantage

3. Calculate the total number of goals for each team. For example, the total number of goals for Arsenal would be the sum of goals in home and away games: $53 + 35 = 88$. Calculate the mean and standard deviation of the total number of goals across all the teams.

```r
library(purrr)

df$Squad <-unlist(map(df$Squad, function(sq) gsub('vs ', '', sq)))

# Total number of Goals
aggregate(df$Gls, by=list(Category=df$Squad), FUN=sum)
```

```
##           Category   x
## 1          Arsenal 126
## 2      Aston Villa  91
## 3      Bournemouth 105
## 4        Brentford  99
## 5         Brighton 115
## 6          Chelsea  81
## 7   Crystal Palace  84
## 8          Everton  89
## 9           Fulham 103
## 10    Leeds United 122
## 11  Leicester City 112
## 12       Liverpool 116
## 13 Manchester City 124
## 14  Manchester Utd  98
## 15   Newcastle Utd  95
## 16 Nott'ham Forest 104
## 17     Southampton 104
## 18       Tottenham 130
## 19        West Ham  95
## 20          Wolves  85
```

```r
# Mean of all team goals
mean(df$Gls)
```

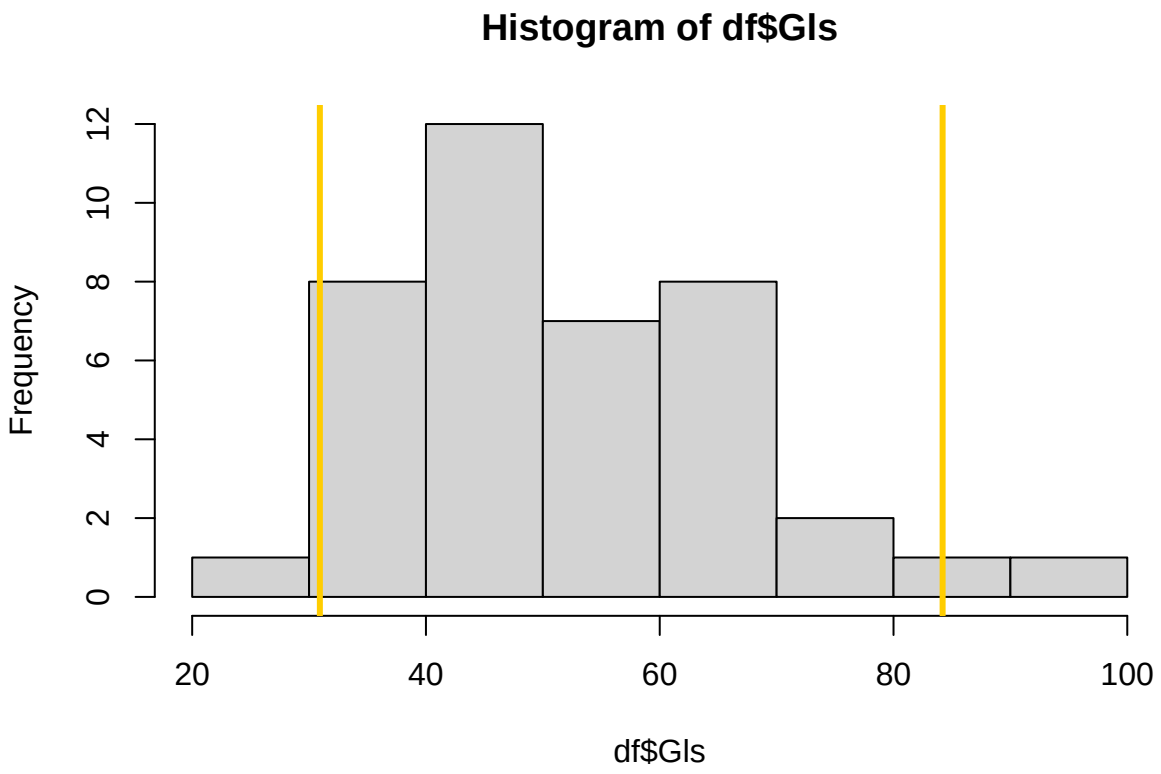
```
## [1] 51.95
```

```r
# SD of all team goals
sd(df$Gls)
```

```
## [1] 15.24156
```

4. Plot a histogram of the total number of goals. Mark the 2.5th and 97.5th quantiles with two vertical lines in the color of *Iowa Gold* (https://brand.uiowa.edu/color).

```r
hist(df$Gls)

abline(v=quantile(df$Gls, .025), col='#FFCD00', lwd=3)
abline(v=quantile(df$Gls, .975), col='#FFCD00', lwd=3)
```



Goals distribution appears to follow a exponential distribution

Problem 2. Random number generators Let $X_1, X_2, \ldots, X_k$ be a random sample from Uniform distribution between 0 and 1. It is known that $Z = X_{\max} - X_{\min}$ follows the Beta$(k-1, 2)$ distribution, where $X_{\max} = \max\{X_1, X_2, \ldots, X_k\}$ and $X_{\min} = \min\{X_1, X_2, \ldots, X_k\}$.

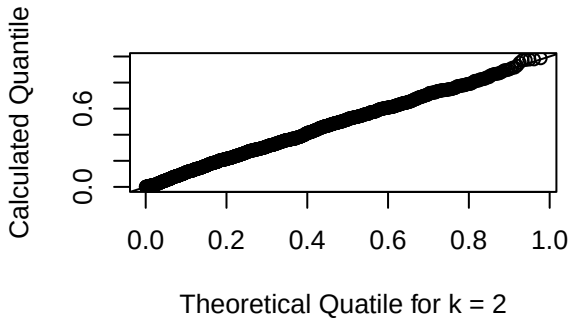
1. Based on the above fact, write an R function `mybeta`, which takes two arguments `n` and `k`, outputing a sequence of n numbers independently from the Beta$(k-1, 2)$ distribution.

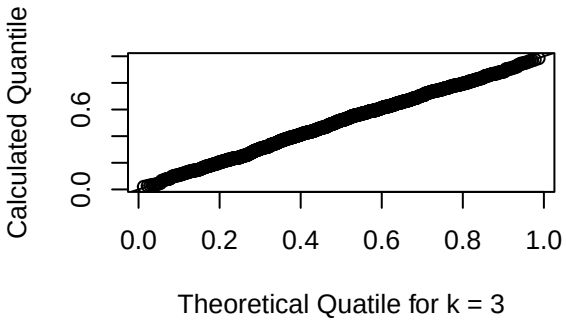
```
mybeta <- function(n = 1000,k) {
  u <- matrix(runif(n*k), n, k)
  return(apply(u, 1, function(row) max(row) - min(row)))
}
```

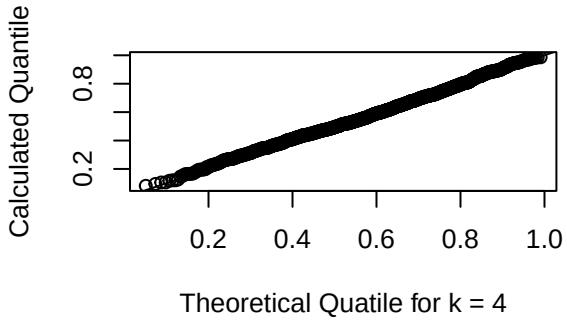
2. Use $k = 2, 3, 4$ as examples. Have Q-Q plots to check whether the samples conform with the corresponding Beta distributions. You may use the built-in function in R to give the inverse CDF function of Beta distribution.

```
n<- 1000

par(mfrow = c(2,2))

for(k in 2:4) {
  plot(
    qbeta(ppoints(n),k-1,2),
    sort(mybeta(n, k)),
    xlab = paste("Theoretical Quatile for k =",k),
    ylab="Calculated Quantile"
```

```
    )
  abline(0,1)
}

par(mfrow = c(1,1))
```







mybeta appears to be a good appropriater for the beta function for Beta(k - 1, 2)

Problem 3. Sampling distributions Suppose $\hat{\theta}$ is an estimator of a population parameter θ. The bias is defined as $\mathrm{E}(\hat{\theta} - \theta)$, and the mean squared error (MSE) is defined as $\mathrm{E}(\hat{\theta} - \theta)^2$.

Suppose $X_1, \ldots, X_{15}$ is a random sample from the $t(2)$ distribution. The simulated data can be generated using the R function `rt`. We consider the trimmed-k mean to estimate the population mean, where the trimmed mean is the average of all the sample observations except for the k largest and k smallest ones. For example, the trimmed-2 mean is the average of $X_{(3)}, X_{(4)}, \ldots, X_{(n-3)}, X_{(n-2)}$, where $X_{(j)}$ is the jth order statistics.

1. For each $k = 1, 2, \ldots, 5$, estimate the bias, variance, and MSE of the trimmed-k mean using simulations with 10^4 replicates.

```
B = 10000
vals <- rt(B,2)

arr <- data.frame(names = c('mean_vals', 'bias', 'variance'))

for (k in 1:5) {
```

```
trimmedk <- function(k, vals) {
  row = list()

  mean_vals <- mean(vals,trim = k/length(vals))
 return(mean_vals)
}

bias <- mean(trimmedk(15, vals) - mean(vals))
mse <- mean((trimmedk(15, vals) - mean(vals))^2)
variance <- var(trimmedk(15, vals))

arr[nrow(arr) + 1,] = c(bias,mse, variance)

}
```

2. Plot the MSE of the trimmed-k mean against $k = 1, 2, \ldots, 5$.

Problem 4. Bootstrap Consider the following airconditioning data set:

```
dat1 <- c(3, 5, 7, 18, 43, 85, 91, 98, 100, 130, 230, 487)
```

1. Suppose this data set is drawn from an underlying distribution F with population mean μ_F. Estimate μ_F using the sample mean, and calculate both the bootstrap and jackknife estimates of the bias.

```
set.seed(5400)

# Sample Mean
muhat <- mean(dat1)
B <- 200

# Generate B bootstraps sample and replications
boot.rep <- rep(NA, 200)
for (b in seq(B)) {
  boot.sample <- sample(dat1, replace=TRUE)
  boot.rep[b] <- mean(boot.sample)
}

# bootstrap estimate of bias
boot.bias <- mean(boot.rep - muhat)

cat('Bootstrap est. of bias is: ',boot.bias)
```

```
## Bootstrap est. of bias is:  -1.182917
```

```
# Jackknife
n <- length(dat1)

# jackknife estimate of bias
jack.r <- rep(NA, n)
for (i in seq(n)) {
  jack.r[i] <- mean(dat1[-i])
```

```
}
jack.bias <- (n-1) * (mean(jack.r) - muhat)

cat('Bootstrap est. of bias is: ',jack.bias)
```

```
## Bootstrap est. of bias is:  0
```

2. In terms of the MSE of the estimator, do you think the bias is large given the standard error?

```
(sd(boot.rep))
```

```
## [1] 38.10996
```

```
(sqrt((n - 1) / n * sum((jack.r - mean(jack.r))^2)))
```

```
## [1] 39.32681
```

Bias values are very small compared to the standard error

3. Now, consider another independently drawn airconditioning data set:

```
dat2 <- c(2, 3, 4, 12, 25, 37, 55, 61, 75, 120, 138, 249)
```

This second data set is drawn from an underlying distribution G with population mean μ_G. The two airconditioning data sets are drawn independently.

Use bootstrap to test:

$$H_0 : \mu_F = \mu_G \quad \text{vs} \quad H_0 : \mu_F > \mu_G.$$

```
set.seed(5400)

# Sample Mean
muhat2 <- mean(dat2)

# Generate B bootstraps sample and replications
boot.rep <- rep(NA, 200)
for (b in seq(B)) {
  boot.sample1 <- sample(dat1, replace=TRUE)
  boot.sample2 <- sample(dat2, replace=TRUE)

  boot.rep[b] <- t.test(boot.sample1, boot.sample2, alternative = "greater")
}
```

** Always interpret your results and findings.**