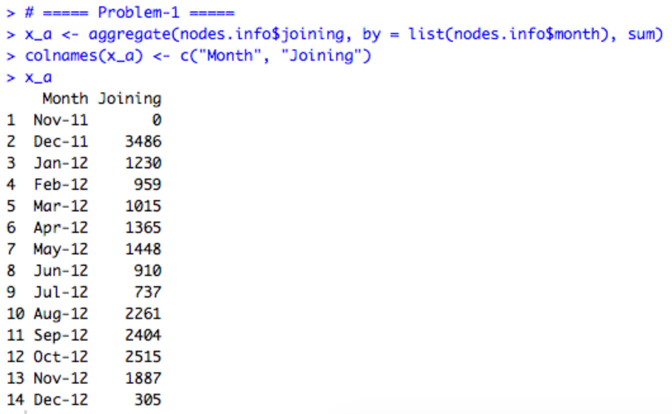
CS688 Web Analytics and Text Mining
Homework#6

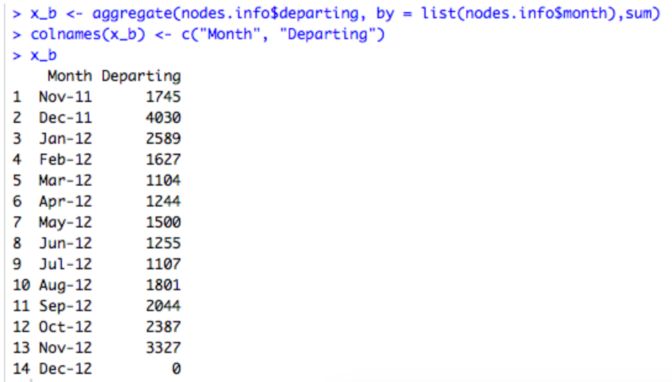
Problem#1

(a) Using aggregate function, compute the data frame for the total players joining each month. Name the columns as *Month* and *Joining*.



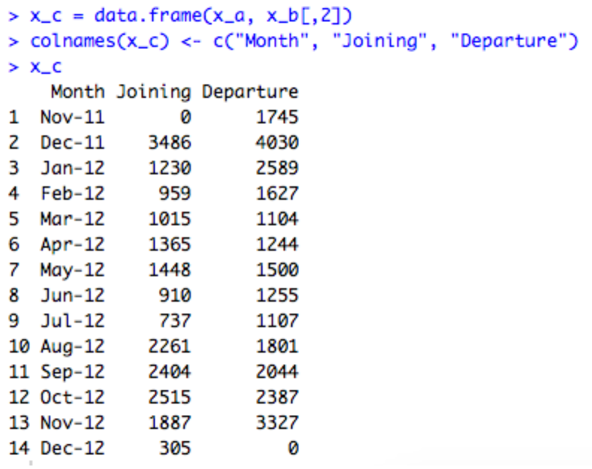
```
> # ===== Problem-1 =====
> x_a <- aggregate(nodes.info$joining, by = list(nodes.info$month), sum)
> colnames(x_a) <- c("Month", "Joining")
> x_a
    Month Joining
1  Nov-11       0
2  Dec-11    3486
3  Jan-12    1230
4  Feb-12     959
5  Mar-12    1015
6  Apr-12    1365
7  May-12    1448
8  Jun-12     910
9  Jul-12     737
10 Aug-12    2261
11 Sep-12    2404
12 Oct-12    2515
13 Nov-12    1887
14 Dec-12     305
```

(b) Using the aggregate function, compute the data frame for the total players departing each month. Names the columns as *Month* and *Departing*.



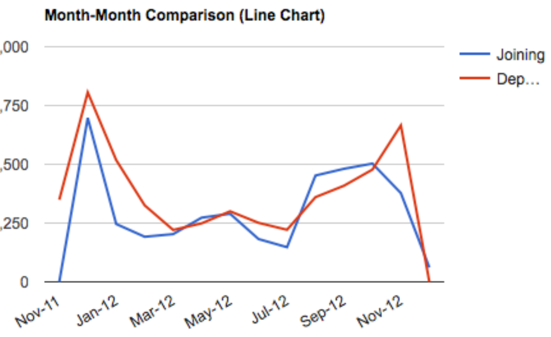
```
> x_b <- aggregate(nodes.info$departing, by = list(nodes.info$month),sum)
> colnames(x_b) <- c("Month", "Departing")
> x_b
    Month Departing
1  Nov-11      1745
2  Dec-11      4030
3  Jan-12      2589
4  Feb-12      1627
5  Mar-12      1104
6  Apr-12      1244
7  May-12      1500
8  Jun-12      1255
9  Jul-12      1107
10 Aug-12      1801
11 Sep-12      2044
12 Oct-12      2387
13 Nov-12      3327
14 Dec-12         0
```

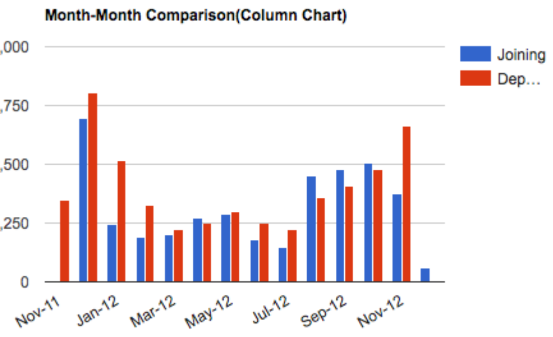
(c) Merge the two data frames by *Month* column with sort option as FALSE



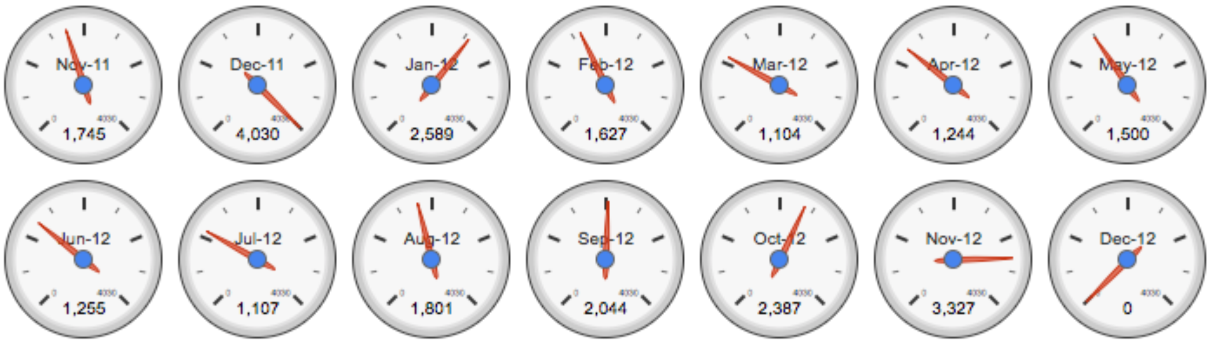
```
> x_c = data.frame(x_a, x_b[,2])
> colnames(x_c) <- c("Month", "Joining", "Departure")
> x_c
    Month Joining Departure
1  Nov-11       0      1745
2  Dec-11    3486      4030
3  Jan-12    1230      2589
4  Feb-12     959      1627
5  Mar-12    1015      1104
6  Apr-12    1365      1244
7  May-12    1448      1500
8  Jun-12     910      1255
9  Jul-12     737      1107
10 Aug-12    2261      1801
11 Sep-12    2404      2044
12 Oct-12    2515      2387
13 Nov-12    1887      3327
14 Dec-12     305         0
```

(d) Show month-by-month comparison of the above numbers using the Google Line chart and Google Column chart. Merge the two into a single chart.



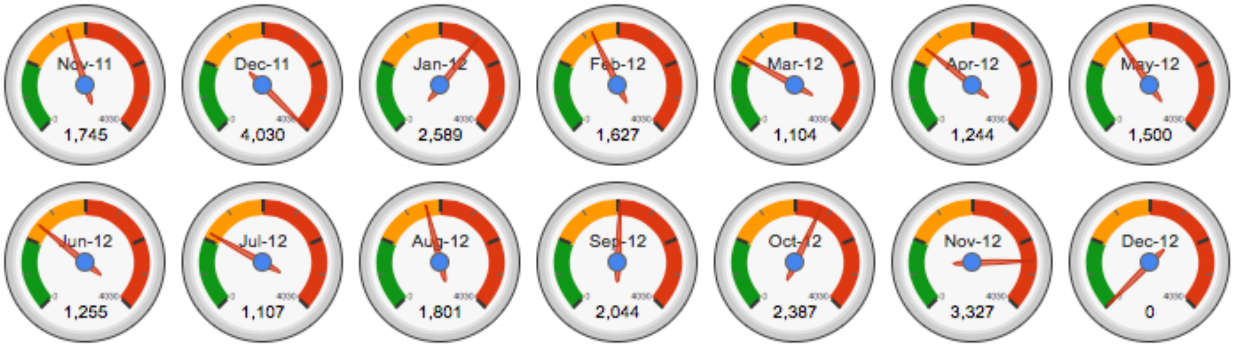


(e) Show the Google Gauge chart with default options for the monthly departing data. Use the range from 0 to 4030.



Data: data.frame(x_c$Month, x_c$Departure) • Chart ID: GaugeIDe9429adb34d • googleVis-0.6.1
R version 3.3.1 (2016-06-21) • Google Terms of Use • Documentation and Data Policy

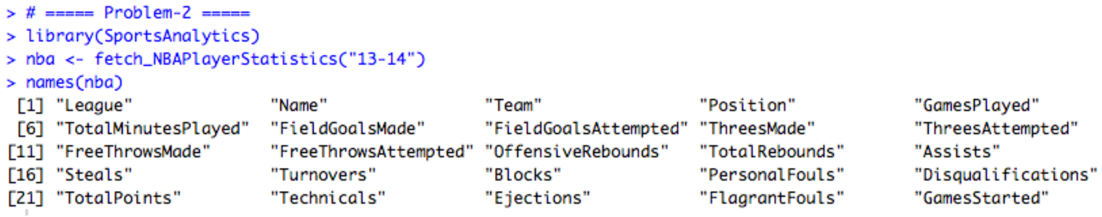
(f) Show the Google Gauge chart for the monthly departing data with the green range 0 – 1000, yellow range 1000 – 2000, and the red range 2000 – 4030.



Data: data.frame(x_c$Month, x_c$Departure) • Chart ID: GaugeIDe944ad6999b • googleVis-0.6.1
R version 3.3.1 (2016-06-21) • Google Terms of Use • Documentation and Data Policy

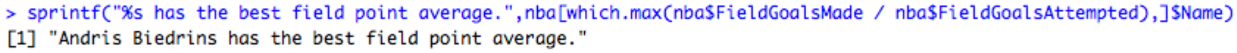
Problem#2

(a) Retrieve the NBA data for the 13-14 season.



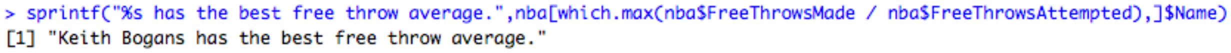
```
> # ===== Problem-2 =====
> library(SportsAnalytics)
> nba <- fetch_NBAPlayerStatistics("13-14")
> names(nba)
 [1] "League"             "Name"               "Team"               "Position"           "GamesPlayed"
 [6] "TotalMinutesPlayed" "FieldGoalsMade"     "FieldGoalsAttempted" "ThreesMade"         "ThreesAttempted"
[11] "FreeThrowsMade"     "FreeThrowsAttempted" "OffensiveRebounds"  "TotalRebounds"      "Assists"
[16] "Steals"             "Turnovers"          "Blocks"             "PersonalFouls"      "Disqualifications"
[21] "TotalPoints"        "Technicals"         "Ejections"          "FlagrantFouls"      "GamesStarted"
```

(b) Which player has the best field point percentage?



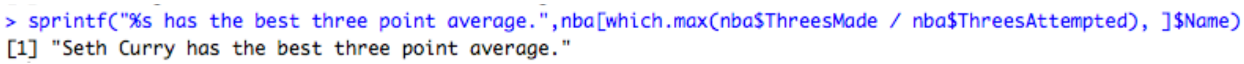
```
> sprintf("%s has the best field point average.",nba[which.max(nba$FieldGoalsMade / nba$FieldGoalsAttempted),]$Name)
[1] "Andris Biedrins has the best field point average."
```

(c) Which player has the best free throw percentage?



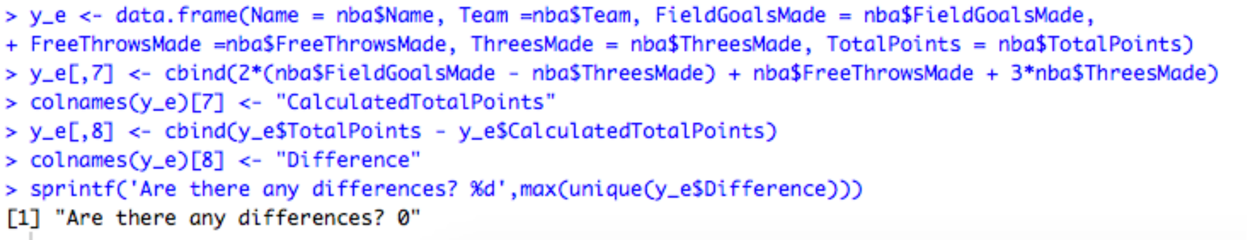
```
> sprintf("%s has the best free throw average.",nba[which.max(nba$FreeThrowsMade / nba$FreeThrowsAttempted),]$Name)
[1] "Keith Bogans has the best free throw average."
```

(d) Which player has the best three point percentage?



```
> sprintf("%s has the best three point average.",nba[which.max(nba$ThreesMade / nba$ThreesAttempted), ]$Name)
[1] "Seth Curry has the best three point average."
```

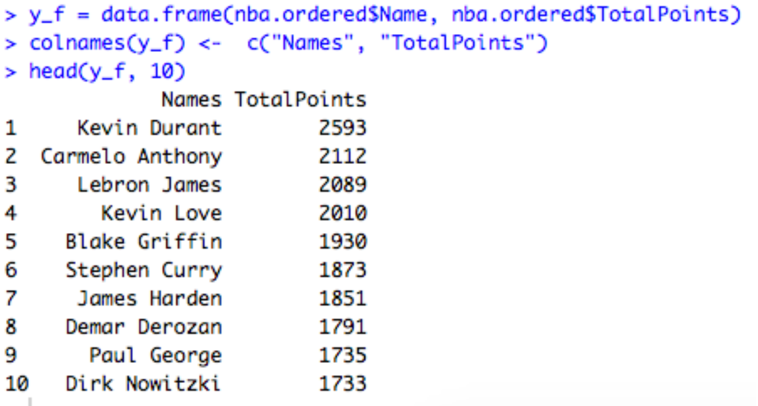
(e) Do you suspect any error in the TotalPoints column in the dataset?



```
> y_e <- data.frame(Name = nba$Name, Team =nba$Team, FieldGoalsMade = nba$FieldGoalsMade,
+ FreeThrowsMade =nba$FreeThrowsMade, ThreesMade = nba$ThreesMade, TotalPoints = nba$TotalPoints)
> y_e[,7] <- cbind(2*(nba$FieldGoalsMade - nba$ThreesMade) + nba$FreeThrowsMade + 3*nba$ThreesMade)
> colnames(y_e)[7] <- "CalculatedTotalPoints"
> y_e[,8] <- cbind(y_e$TotalPoints - y_e$CalculatedTotalPoints)
> colnames(y_e)[8] <- "Difference"
> sprintf('Are there any differences? %d',max(unique(y_e$Difference)))
[1] "Are there any differences? 0"
```

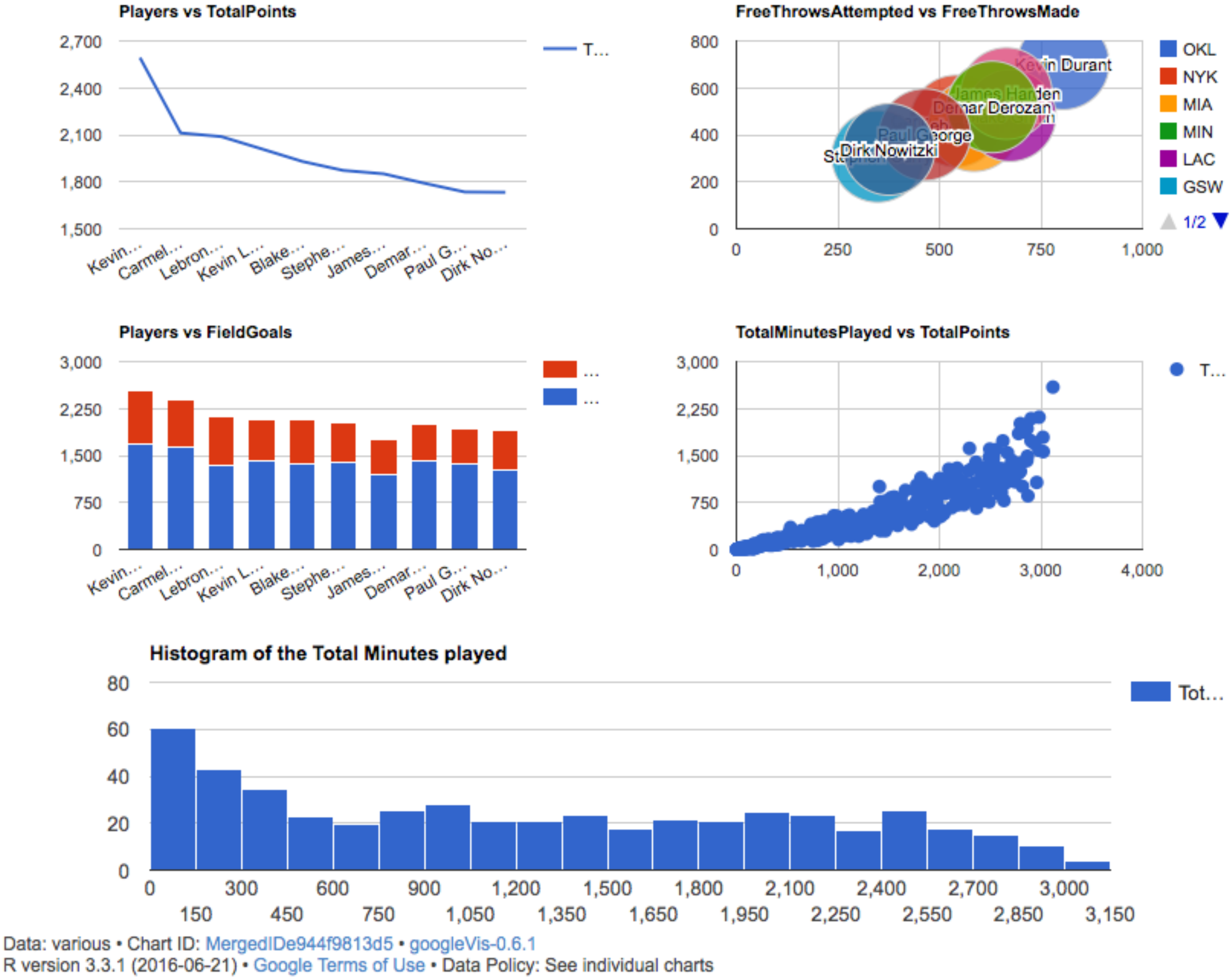
0 -> means no error and 1 -> means error exists in the data

(f) Show the top 10 players in terms of TotalPoints, arranged from the highest to lowest.



```
> y_f = data.frame(nba.ordered$Name, nba.ordered$TotalPoints)
> colnames(y_f) <-  c("Names", "TotalPoints")
> head(y_f, 10)
             Names TotalPoints
1     Kevin Durant        2593
2  Carmelo Anthony        2112
3      Lebron James        2089
4        Kevin Love        2010
5    Blake Griffin        1930
6    Stephen Curry        1873
7     James Harden        1851
8    Demar Derozan        1791
9      Paul George        1735
10   Dirk Nowitzki        1733
```

(g) Use at least 5 Google charts (your choice) to show relevant data from this dataset.

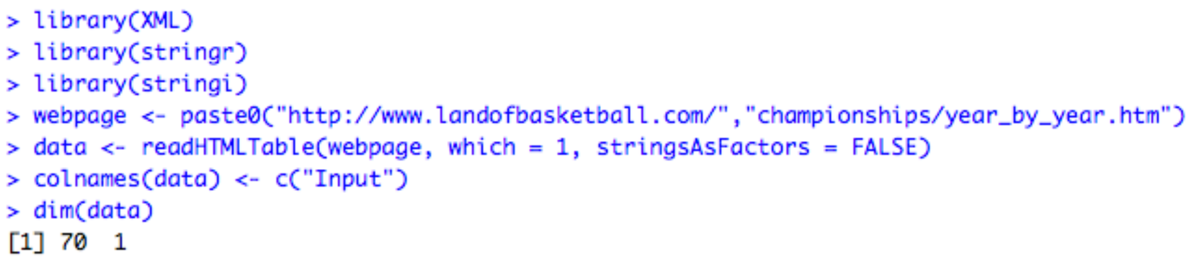


Data: various • Chart ID: MergedIDe944f9813d5 • googleVis-0.6.1
R version 3.3.1 (2016-06-21) • Google Terms of Use • Data Policy: See individual charts

Problem#3

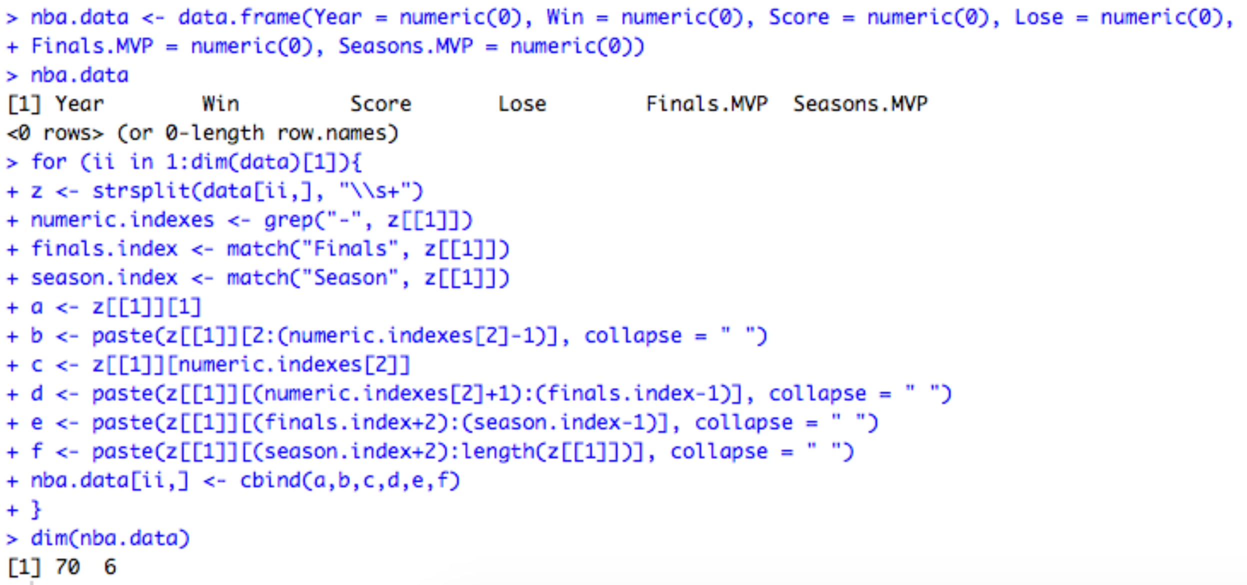
The NBA championship data from the landofbasketball.com

Data preparation:



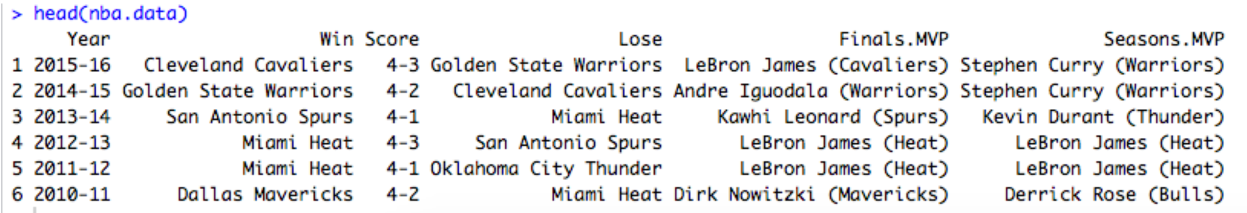
```
> library(XML)
> library(stringr)
> library(stringi)
> webpage <- paste0("http://www.landofbasketball.com/","championships/year_by_year.htm")
> data <- readHTMLTable(webpage, which = 1, stringsAsFactors = FALSE)
> colnames(data) <- c("Input")
> dim(data)
[1] 70  1
```

Converted into data frame of 70 rows and 6 columns



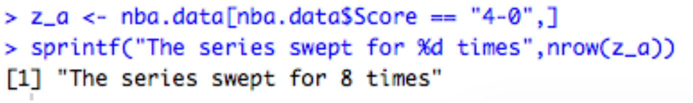
```
> nba.data <- data.frame(Year = numeric(0), Win = numeric(0), Score = numeric(0), Lose = numeric(0),
+ Finals.MVP = numeric(0), Seasons.MVP = numeric(0))
> nba.data
[1] Year        Win         Score       Lose        Finals.MVP  Seasons.MVP
<0 rows> (or 0-length row.names)
> for (ii in 1:dim(data)[1]){
+ z <- strsplit(data[ii,], "\\s+")
+ numeric.indexes <- grep("-", z[[1]])
+ finals.index <- match("Finals", z[[1]])
+ season.index <- match("Season", z[[1]])
+ a <- z[[1]][1]
+ b <- paste(z[[1]][2:(numeric.indexes[2]-1)], collapse = " ")
+ c <- z[[1]][numeric.indexes[2]]
+ d <- paste(z[[1]][(numeric.indexes[2]+1):(finals.index-1)], collapse = " ")
+ e <- paste(z[[1]][(finals.index+2):(season.index-1)], collapse = " ")
+ f <- paste(z[[1]][(season.index+2):length(z[[1]])], collapse = " ")
+ nba.data[ii,] <- cbind(a,b,c,d,e,f)
+ }
> dim(nba.data)
[1] 70  6
```

Processed dataframe.



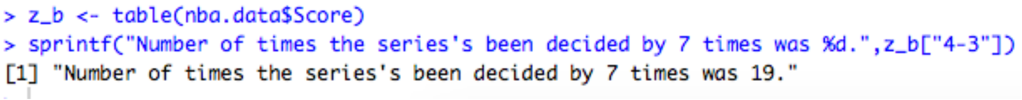
```
> head(nba.data)
     Year                   Win Score                  Lose                Finals.MVP              Seasons.MVP
1 2015-16   Cleveland Cavaliers   4-3 Golden State Warriors  LeBron James (Cavaliers) Stephen Curry (Warriors)
2 2014-15 Golden State Warriors   4-2   Cleveland Cavaliers Andre Iguodala (Warriors) Stephen Curry (Warriors)
3 2013-14     San Antonio Spurs   4-1            Miami Heat     Kawhi Leonard (Spurs)   Kevin Durant (Thunder)
4 2012-13            Miami Heat   4-3     San Antonio Spurs       LeBron James (Heat)      LeBron James (Heat)
5 2011-12            Miami Heat   4-1 Oklahoma City Thunder       LeBron James (Heat)      LeBron James (Heat)
6 2010-11      Dallas Mavericks   4-2            Miami Heat Dirk Nowitzki (Mavericks)     Derrick Rose (Bulls)
```

(a) How many times was the series swept, i.e., decided by the series score 4-0?



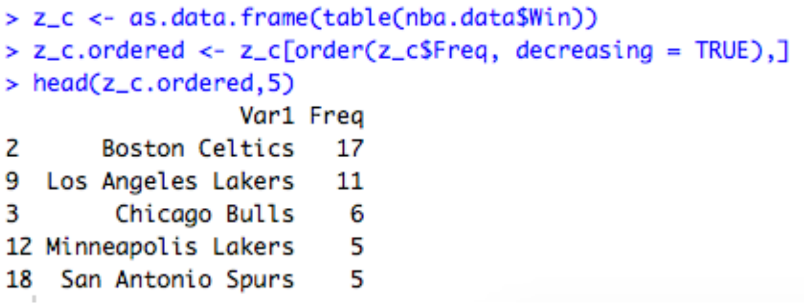
```
> z_a <- nba.data[nba.data$Score == "4-0",]
> sprintf("The series swept for %d times",nrow(z_a))
[1] "The series swept for 8 times"
```

(b) How many times was the series decided by game 7? (Series score 4-3)



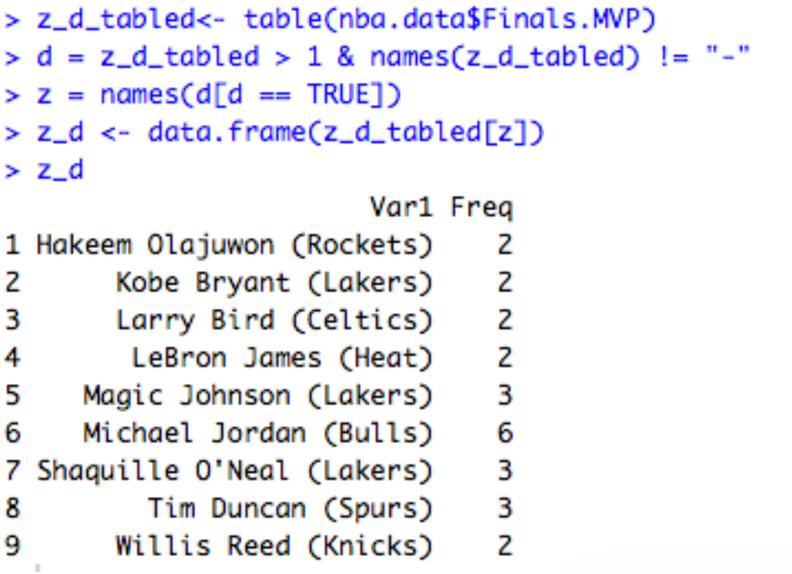
```
> z_b <- table(nba.data$Score)
> sprintf("Number of times the series's been decided by 7 times was %d.",z_b["4-3"])
[1] "Number of times the series's been decided by 7 times was 19."
```

(c) Show 5 teams that have the most wins in descending order.



```
> z_c <- as.data.frame(table(nba.data$Win))
> z_c.ordered <- z_c[order(z_c$Freq, decreasing = TRUE),]
> head(z_c.ordered,5)
                  Var1 Freq
2       Boston Celtics   17
9   Los Angeles Lakers   11
3        Chicago Bulls    6
12 Minneapolis Lakers     5
18   San Antonio Spurs    5
```

(d) Create a subset of the lecture data frame with championship data from the last championship to the 1968 season. Using the split data column from the lecture example, add a new column showing the Finals MVP. Show the players who won the FinalsMVP award more than once.



```
> z_d_tabled<- table(nba.data$Finals.MVP)
> d = z_d_tabled > 1 & names(z_d_tabled) != "-"
> z = names(d[d == TRUE])
> z_d <- data.frame(z_d_tabled[z])
> z_d
                      Var1 Freq
1 Hakeem Olajuwon (Rockets)    2
2       Kobe Bryant (Lakers)    2
3       Larry Bird (Celtics)    2
4         LeBron James (Heat)    2
5     Magic Johnson (Lakers)    3
6     Michael Jordan (Bulls)    6
7 Shaquille O'Neal (Lakers)    3
8         Tim Duncan (Spurs)    3
9       Willis Reed (Knicks)    2
```

R-codes:

```
library(RCurl)
library(RJSONIO)
library(googleVis)
cat("\014")
webpage <- paste("http://powerful-meadow-8588.herokuapp.com/data/12months_departures_joiners.json", sep = "")
data <- fromJSON(getURL(webpage))
names(data)
data$nodes[[1]]

nodes.info <- do.call("rbind", lapply(data$nodes, data.frame))
head(nodes.info)

# ===== Problem-1 =====
# 1-a
x_a <- aggregate(nodes.info$joining, by = list(nodes.info$month), sum)
colnames(x_a) <- c("Month", "Joining")
x_a

# 1-b
x_b <- aggregate(nodes.info$departing, by = list(nodes.info$month),sum)
colnames(x_b) <- c("Month", "Departing")
x_b

# 1-c
x_c = data.frame(x_a, x_b[,2])
colnames(x_c) <- c("Month", "Joining", "Departure")
x_c

# 1-d
line.chart <- gvisLineChart(x_c, xvar = "Month", yvar = c("Joining", “Departure"), options = list(title = "Month-Month
                        Comparison (Line Chart)”, width = 500, height = 300))
column.chart <- gvisColumnChart(x_c, xvar = "Month", yvar = c("Joining", “Departure"), options = list(title = "Month-Month
                        Comparison(Column Chart)”, width = 500, height = 300))

merged.chart <- gvisMerge(line.chart, column.chart, horizontal = TRUE)
plot(merged.chart)

# 1-e
gauge.chart <- gvisGauge(data.frame(x_c$Month,x_c$Departure), options = list(title = "Monthly-departing data (Gauge Chart)",
                                  min = 0, max = 4030, width = 700, height = 300))
plot(gauge.chart)

# 1-f
gauge.chart.colored <- gvisGauge(data.frame(x_c$Month,x_c$Departure), options = list(title = "Monthly-departing data (Gauge
                                    Chart)”, min = 0, max = 4030, greenFrom = 0, greenTo = 1000, yellowFrom = 1000,
                                    yellowTo = 2000, redFrom = 2000, redTo = 4030, width = 700, height = 300))
plot(gauge.chart.colored)


# ===== Problem-2 =====
library(SportsAnalytics)

# 2-a
nba <- fetch_NBAPlayerStatistics("13-14")
names(nba)

# 2-b
sprintf("%s has the best field point average.",nba[which.max(nba$FieldGoalsMade / nba$FieldGoalsAttempted),]$Name)

# 2-c
sprintf("%s has the best free throw average.",nba[which.max(nba$FreeThrowsMade / nba$FreeThrowsAttempted),]$Name)
```

```
# 2-d
sprintf("%s has the best three point average.",nba[which.max(nba$ThreesMade / nba$ThreesAttempted), ]$Name)

# 2-e
y_e <- data.frame(Name = nba$Name, Team =nba$Team, FieldGoalsMade = nba$FieldGoalsMade,
            FreeThrowsMade =nba$FreeThrowsMade, ThreesMade = nba$ThreesMade, TotalPoints = nba$TotalPoints)
y_e[,7] <- cbind(2*(nba$FieldGoalsMade - nba$ThreesMade) + nba$FreeThrowsMade + 3*nba$ThreesMade)
colnames(y_e)[7] <- "CalculatedTotalPoints"
y_e[,8] <- cbind(y_e$TotalPoints - y_e$CalculatedTotalPoints)
colnames(y_e)[8] <- "Difference"
sprintf('Are there any differences? %d',max(unique(y_e$Difference)))

# 2-f
nba.ordered <- nba[order(nba$TotalPoints, decreasing = TRUE),]
y_f = data.frame(nba.ordered$Name, nba.ordered$TotalPoints)
colnames(y_f) <-  c("Names", "TotalPoints")
head(y_f, 10)

# 2-g
y_g_line <- gvisLineChart(y_f[1:10,], xvar = "Names", yvar = "TotalPoints", options = list(title = "Players vs TotalPoints"))

y_g_bubble <- gvisBubbleChart(nba.ordered[1:10,], idvar = "Name", xvar = "FreeThrowsAttempted",  yvar =
                  "FreeThrowsMade", colorvar = “Team", options = list(title = "FreeThrowsAttempted vs FreeThrowsMade"))

y_g_bar <- gvisColumnChart(data.frame(Players=nba.ordered$Name[1:10], GoalsAttempted=nba.ordered
                  $FieldGoalsAttempted[1:10], GoalsMade = nba.ordered$FieldGoalsMade[1:10]), xvar = "Players", yvar =
                  c("GoalsAttempted", “GoalsMade"), options = list(title = "Players vs FieldGoals", isStacked = TRUE))

y_g_scatter <- gvisScatterChart(data.frame(TotalMinutes = nba.ordered$TotalMinutesPlayed, TotalPoints = nba.ordered
                  $TotalPoints), options = list(title = "TotalMinutesPlayed vs TotalPoints"))

y_g_histogram <- gvisHistogram(data.frame(TotalMinutes = nba.ordered$TotalMinutesPlayed), options = list(title = "Histogram
                  of the Total Minutes played"))

y_g_charts <- gvisMerge(gvisMerge(gvisMerge(y_g_line, y_g_bubble, horizontal = TRUE), gvisMerge(y_g_bar, y_g_scatter,
                  horizontal = TRUE)),  y_g_histogram, horizontal = FALSE)
plot(y_g_charts)

# ===== Problem-3 ======
library(XML)
library(stringr)
library(stringi)

# Fetching the data
webpage <- paste0("http://www.landofbasketball.com/","championships/year_by_year.htm")
data <- readHTMLTable(webpage, which = 1, stringsAsFactors = FALSE)
dim(data)

# Creating an empty data frame
nba.data <- data.frame(Year = numeric(0), Win = numeric(0), Score = numeric(0), Lose = numeric(0),
          Finals.MVP = numeric(0), Seasons.MVP = numeric(0))
nba.data

# Processing the data
for (ii in 1:dim(data)[1]){
  z <- strsplit(data[ii,], "\\s+")
  numeric.indexes <- grep("-", z[[1]])
  finals.index <- match("Finals", z[[1]])
  season.index <- match("Season", z[[1]])
  a <- z[[1]][1]
  b <- paste(z[[1]][2:(numeric.indexes[2]-1)], collapse = " ")
  c <- z[[1]][numeric.indexes[2]]
  d <- paste(z[[1]][(numeric.indexes[2]+1):(finals.index-1)], collapse = " ")
  e <- paste(z[[1]][(finals.index+2):(season.index-1)], collapse = " ")
  f <- paste(z[[1]][(season.index+2):length(z[[1]])], collapse = " ")
  nba.data[ii,] <- cbind(a,b,c,d,e,f)
}
dim(nba.data)
head(nba.data)

# 3-a
z_a <- nba.data[nba.data$Score == "4-0",]
sprintf("The series swept for %d times",nrow(z_a))

# 3-b
z_b <- table(nba.data$Score)
sprintf("Number of times the series's been decided by 7 times was %d.",z_b["4-3"])

# 3-c
z_c <- as.data.frame(table(nba.data$Win))
z_c.ordered <- z_c[order(z_c$Freq, decreasing = TRUE),]
head(z_c.ordered,5)

# 3-d
z_d_tabled<- table(nba.data$Finals.MVP)

d = z_d_tabled > 1 & names(z_d_tabled) != "-"
z = names(d[d == TRUE])
z_d <- data.frame(z_d_tabled[z])
z_d
```