STT 301 Homework Assignment 1 Solutions

Shawn Santo

September 10, 2018

Question 1 (3 points)

This question focuses on some basic manipulations of vectors in R.

Part a (1 point)

Create four vectors in R: one called `nums` which contains the integers from 5 through 14; one called `charnums` which contains character representations of the numbers 1 through 4, namely, “1”, “2”, “3”, “4”; one called `mixed` which contains the same values as in `charnums` , but which also contains the letters “a” and “b”; and one called `bool` which contains the logical values TRUE, TRUE, TRUE, FALSE, FALSE, TRUE.

```
nums <- c(5:14)
charnums <- c("1", "2", "3", "4")
mixed <- c(charnums, "a", "b")
bool <- c(T, T, T, F, F, T)
```

Part b (1 point)

i. Convert `nums` to character.
ii. Convert `charnums` to numeric.
iii. Investigate what happens when you convert `mixed` to numeric.
iv. Investigate what happens when you covert `bool` to character and then when you convert the character result of `bool` to numeric.

Comment on each of these conversions below the `R` chunk.

```
as.character(nums)
```

```
 [1] "5"  "6"  "7"  "8"  "9"  "10" "11" "12" "13" "14"
```

```
as.numeric(charnums)
```

```
[1] 1 2 3 4
```

```
as.numeric(mixed)
```

```
Warning: NAs introduced by coercion
```

```
[1]  1  2  3  4 NA NA
```

```
bool_char <- as.character(bool)
bool_char
```

```
[1] "TRUE"  "TRUE"  "TRUE"  "FALSE" "FALSE" "TRUE"
```

```
as.numeric(bool_char)
```

```
Warning: NAs introduced by coercion
```

```
[1] NA NA NA NA NA NA
```

When `nums` is converted to type character, character representations of the numbers are returned. When `charnums` is converted to numeric, the elements are returned as type numeric. However,when `mixed` is converted to numeric, numeric values are returned for “1”, “2”, “3”, and “4”, but “a” and “b” are represented as missing values since there is no numeric equivalent. The vector `bool_char` is of type character with each element being “TRUE” or “FALSE”. Trying to convert `bool_char` to numeric results in `NA` values since no equivalent numeric values exists for character representations of “TRUE” and “FALSE”.

Part c (1 point)

i. Extract the first element of `bool` .

ii. Extract the last element of `nums.` You should give code which would work regardless of the number of elements in `nums` .

iii. Extract all but the first element of `nums.`

iv. Extract all but the first two and last two elements of `nums.` Again, give code that would work regardless of how many elements `nums` contains.

```
bool[1]
```

```
[1] TRUE
```

```
nums[length(nums)]
```

```
[1] 14
```

```
nums[-1]
```

```
[1]  6  7  8  9 10 11 12 13 14
```

```
nums[3:(length(nums) - 2)]
```

```
[1]  7  8  9 10 11 12
```

Question 2 (4 points)

For this question you will work with data from the 2016-2017 NBA season. The `R` chunk below gives the code to read in the data. Use the `nba` data frame for the questions that follow. If you are unfamiliar with basketball terminology, most of the definitions for the variables can be found at Basketball Reference (https://www.basketball-reference.com/about/glossary.html).

The first time you knit the .Rmd file it will be slow. The chunk option `cache=TRUE` will speed up the knit time after the initial knit.

```
# read in the data
nba_raw <- read.csv("http://users.stat.ufl.edu/~winner/data/nba_player_201617.csv",
                    stringsAsFactors = FALSE)
# remove the last column
nba <- nba_raw[, -31]
```

Part a (1 point)

Create a data frame containing all the variables for a player of your choice. Save the data frame using the player’s last name. You may want to use the `subset` function.

```
embiid <- subset(nba, Player == "Joel Embiid")
# check size of data frame
dim(embiid)
```

```
[1] 31 30
```

Part b (2 points)

Compute the mean, standard deviation, and median for any three variables corresponding to the player you selected in the previous part. To extract a single variable you can use `[ , ]` or `$`. For example, if my data frame was `westbrook`, `westbrook$PTS` would provide me the points vector from the `westbrook` data frame.

```
# minutes per game
sum_names <- c("mean", "standard deviation", "median")
mpg_sum <- c(mean(embiid$Minutes), sd(embiid$Minutes), median(embiid$Minutes))
names(mpg_sum) <- sum_names

# points per game
pts_sum <- c(mean(embiid$PTS), sd(embiid$PTS), median(embiid$PTS))
names(pts_sum) <- sum_names

# blocks per game
blk_sum <- c(mean(embiid$BLK), sd(embiid$BLK), median(embiid$BLK))
names(blk_sum) <- sum_names

mpg_sum
```

```
            mean standard deviation              median 
       25.357097           3.186446           25.880000 
```

```
pts_sum
```

```
            mean standard deviation              median 
       20.225806           6.205963           22.000000 
```

```
blk_sum
```

```
            mean standard deviation              median 
        2.451613           1.178663            2.000000 
```

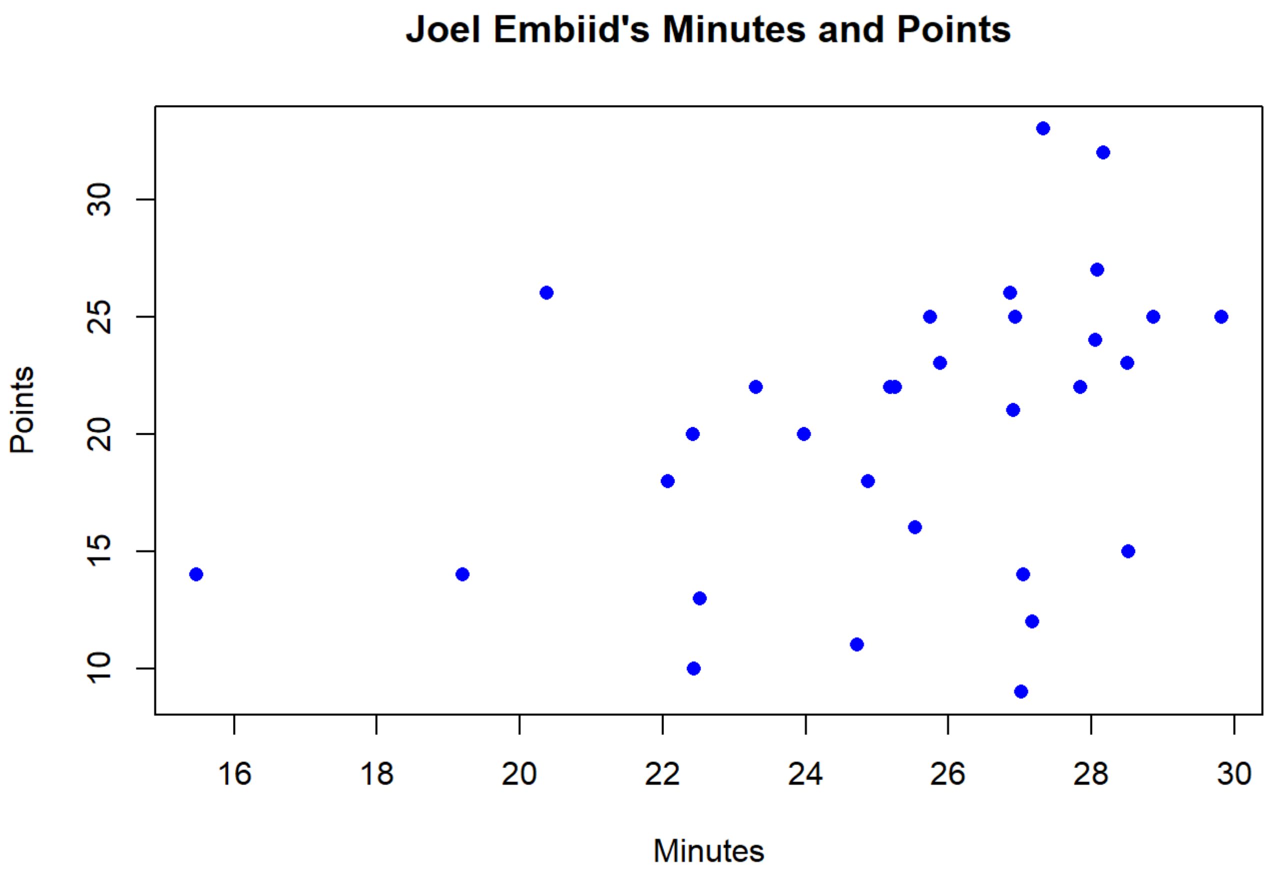
Part c (1 point)

Compute the correlation between `Minutes` and `PTS` for the player you selected, and plot the two variables. `Minutes` should be on the x-axis, and `PTS` should be on the y-axis. If you want to tidy your plot a bit include `xlab = "Minutes", ylab = "Points"` inside the plot function. Look at the help if you want to see how to change colors or points.

```
cor(embiid$Minutes, embiid$PTS)
```

```
[1] 0.3878477
```

```
plot(x = embiid$Minutes, y = embiid$PTS,
     xlab = "Minutes", ylab = "Points", main = "Joel Embiid's Minutes and Points",
     col = "blue", pch = 16)
```



Question 3 (3 points)

For this question you will work with the `nba` data frame created in question 2.

Select two players. For each player, compute their field goal percentage and effective field goal percentage for all their games played. Use at least two descriptive statistics to determine which player had the better field goal percentage, and which player had the better effective field goal percentage for the 2016-2017 NBA season. Your answer should include an explanation.

Field Goal Percentage; the formula is `FG / FGA`.

Effective Field Goal Percentage; the formula is `(FG + 0.5 * X3P) / FGA`. This statistic adjusts for the fact that a 3-point field goal is worth one more point than a 2-point field goal. For example, suppose Player A goes 4 for 10 with 2 threes, while Player B goes 5 for 10 with 0 threes. Each player would have 10 points from field goals, and each would have the same effective field goal percentage (50%).

```
westbrook <- subset(nba, Player == "Russell Westbrook")
durant <- subset(nba, Player == "Kevin Durant")

dim(westbrook)
```

```
[1] 81 30
```

```
dim(durant)
```

```
[1] 62 30
```

```
# fgp for each player
westbrook_fgp <- westbrook$FG / westbrook$FGA
durant_fgp <- durant$FG / durant$FGA

summary(westbrook_fgp, na.rm = TRUE)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 0.2000  0.3333  0.4333  0.4251  0.5000  1.0000 
```

```
summary(durant_fgp, na.rm = TRUE)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 0.0000  0.4706  0.5333  0.5340  0.6250  0.8462 
```

Durant appears to be a more efficient shooter than Westbrook. Durant has a higher mean per game field goal percentage and his first quartile is above Westbrook's median per game field goal percentage. I used the option `na.rm = TRUE` to remove any NaN values that would appear if a player had 0 field goal attempts in a game. In our calculation of field goal percentage, dividing 0 by 0 would result in a value NaN. This was probably not necessary for Westbrook as it would be jaw-dropping if he took 0 field goal attempts in a game.

```
#efgp for each player
westbrook_efgp <- (westbrook$FG + 0.5 * westbrook$X3P) / westbrook$FGA
durant_efgp <- (durant$FG + 0.5 * durant$X3P) / durant$FGA

summary(westbrook_efgp, na.rm = TRUE)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 0.2500  0.3889  0.4783  0.4763  0.5645  1.0000 
```

```
summary(durant_efgp, na.rm = TRUE)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 0.0000  0.5000  0.5844  0.5912  0.6867  1.0000 
```

A similar conclusion can be drawn for effective field goal percentage as with the field goal percentage. Durant appears to be the more efficient shooter.