HW1_IS457_85

Mon Sep 17, 2018

Part 1. LifeCycleSavings Data

In this part, we will work with a built-in dataset -- LifeCycleSavings.

(1) R has a built-in help function, write your call to the help function below, as well as something that you learned about this dataset from the help function. (1 pt)

Ans :

help(LifeCycleSavings)

This dataset gives us information of life-cycle savings from the decade 1960-1970 from 50 countries according to the hypothesis developed by Franco Modigliani. It contains 50 observations from 50 countries on 5 variables viz. 1) savings ratio 2) population under 15 3) population over 75 4) real per-capita disposable income (di) 5) percent growth rate of disposable income.

(2) Describe this dataset (structure, variables, value types, size, etc.) (2 pts)

Ans :

names(LifeCycleSavings)

```
[1] "sr"    "pop15" "pop75" "dpi"   "ddpi"
```

summary(LifeCycleSavings)

```
       sr             pop15           pop75            dpi               ddpi       
 Min.   : 0.600   Min.   :21.44   Min.   :0.560   Min.   :  88.94   Min.   : 0.220  
 1st Qu.: 6.970   1st Qu.:26.21   1st Qu.:1.125   1st Qu.: 288.21   1st Qu.: 2.002  
 Median :10.510   Median :32.58   Median :2.175   Median : 695.66   Median : 3.000  
 Mean   : 9.671   Mean   :35.09   Mean   :2.293   Mean   :1106.76   Mean   : 3.758  
 3rd Qu.:12.617   3rd Qu.:44.06   3rd Qu.:3.325   3rd Qu.:1795.62   3rd Qu.: 4.478  
 Max.   :21.100   Max.   :47.64   Max.   :4.700   Max.   :4001.89   Max.   :16.710  
```

str(LifeCycleSavings)

```
'data.frame':  50 obs. of  5 variables:
 $ sr   : num  11.43 12.07 13.17 5.75 12.88 ...
 $ pop15: num  29.4 23.3 23.8 41.9 42.2 ...
 $ pop75: num  2.87 4.41 4.43 1.67 0.83 2.85 1.34 0.67 1.06 1.14 ...
 $ dpi  : num  2330 1508 2108 189 728 ...
 $ ddpi : num  2.87 3.93 3.82 0.22 4.56 2.43 2.67 6.51 3.08 2.8 ...
```

This dataset consists of 5 variables (columns) and 50 observations (rows) for the variables from 50 countries. The name for each column is “sr”, “pop15”, “pop75”, “dpi”, “ddpi” respectively. The datatype for each column is of numeric. The dataset has the dimensions of [1] 50 5 which stands form 50 rows and 5 columns. The summary() function summarizes the values of each variable according to its data type. In this case its gives us the mean, median, min, max value for each columns.

(3) What is "aggregate personal savings" in this dataset? Calculate the average aggregate personal savings of these 50 countries. (1 pt)

Ans :

```
mean(LifeCycleSavings[["sr"]])
```

```
[1] 9.671
```

The “aggregate personal savings” in this dataset is the ratio between the personal savings and the disposable income of individual. It signifies the average savings rate of the individuals from a particular country from the decade 1960-1970. Running the above command gives us the average aggregate personal savings to be 9.671.

(4) What is "dpi" in this dataset? Find the highest and lowest dpi. (2 pts)

Ans :

```
summary(LifeCycleSavings[4])
```

```
      dpi
 Min.   :  88.94
 1st Qu.: 288.21
 Median : 695.66
 Mean   :1106.76
 3rd Qu.:1795.62
 Max.   :4001.89
```

```
min(LifeCycleSavings[4])
```

```
[1] 88.94
```

```
max(LifeCycleSavings[4])
```

```
[1] 4001.89
```

The dpi in the dataset is the real per-capita disposable income for each country. Disposable income means the income the remains in a person's hand after deducting taxes and other mandatory charges. The person can either spend it or save it. Per-capita means that the income is for each individual rather than household income. The highest and the lowest dpi in the dataset is 4001.89 and 88.94 respectively.

(5) How many countries have a dpi above median? (2 pts) hint: you might need to find a function to count rows.

Ans :

```
m = median(LifeCycleSavings[["dpi"]])
nrow(LifeCycleSavings[LifeCycleSavings$dpi > m,])
```

```
[1] 25
```

The above code shows us that there are 25 countries having a dpi greater than median.

(6) What is the highest aggregate personal savings of the countries whose pop15s are more than 10 times their pop75s? (2 pts)

Ans :

```
d = LifeCycleSavings[LifeCycleSavings$pop15 > (10*LifeCycleSavings$pop75),]
max(d["sr"])
```

```
[1] 21.1
```

The highest aggregate personal savings of the countries is 21.1.

(7) For the countries with dpi above the 75th percentile, what is their average aggregate personal savings? For the countries with dpi above the 75th percentile, what is their median aggregate personal savings? Why are these two statistics different?

Ans :

```
mean(LifeCycleSavings$sr[LifeCycleSavings$dpi > 1795.62])
```

```
[1] 10.28154
```

```
median(LifeCycleSavings$sr[LifeCycleSavings$dpi > 1795.62])
```

```
[1] 10.35
```

The values are different because the mean averages all the values evenly and even considers any outlier values whereas the median sorts the sample values and then picks out the middlemost value in case of odd number of samples and takes the average of two middle values in case of even number of samples.

(8) Let's look at countries with dpi below the 25th percentile. What is their average and their median aggregate personal savings? Why are these two statistics different? Is the pattern of difference different than what you saw in Q7? Why or Why not?

Ans :

```
mean(LifeCycleSavings$sr[LifeCycleSavings$dpi < 288.21])
```

```
[1] 7.543846
```

```
median(LifeCycleSavings$sr[LifeCycleSavings$dpi < 288.21])
```

```
[1] 5.75
```

The values are different for the same reasons they were different in Q7. There is a strong positive correlation between the values of mean and median. The pattern of difference is same as that of the Q7 as the relationship between the mean and median remains the same across different quartile ranges.

(9). (3 pts)
Try running each expression in R.
Record the error message in a comment
Explain what it means.
Be sure to directly relate the wording of the error message with the problem you find in the expression.

```
LifeCycleSavings[LifeCycleSavings$pop15 > 10]
```

Error message here

```
Error in `[.data.frame`(LifeCycleSavings, LifeCycleSavings$pop15 > 10) :
  undefined columns selected
```

Explanation here

Explanation: The above error is shown because R expects the expression between the square brackets to subset rows and columns in case of a dataframe. The above expression **LifeCycleSavings$pop15 > 10** subsets only the rows of the dataframe and does not provide any condition for the columns. If we want the above code to work for all columns then the correct statement is **LifeCycleSavings[LifeCycleSavings$pop15 > 10,]**. The comma subsets the column by **all**.

```
mean(pop15,pop75)
```

Error message here

```
Error in mean(pop15, pop75) : object 'pop15' not found
```

Explanation here

Explanation: **pop15** is a variable of the dataset **LifeCycleSavings**. So, to use it we must access them through the dataset name. Hence R throws a no object found error.

```
mean(LifeCycleSavings$pop15, LifeCycleSavings$pop75)
```

Error message here

```
Error in mean.default(LifeCycleSavings$pop15, LifeCycleSavings$pop75) :
'trim' must be numeric of length one
```

Explanation here

Explanation: The syntax for the mean function is mean(object, trim =). Here **LifeCycleSavings$pop75** does not have length one. So, it cannot be used as a trim value.

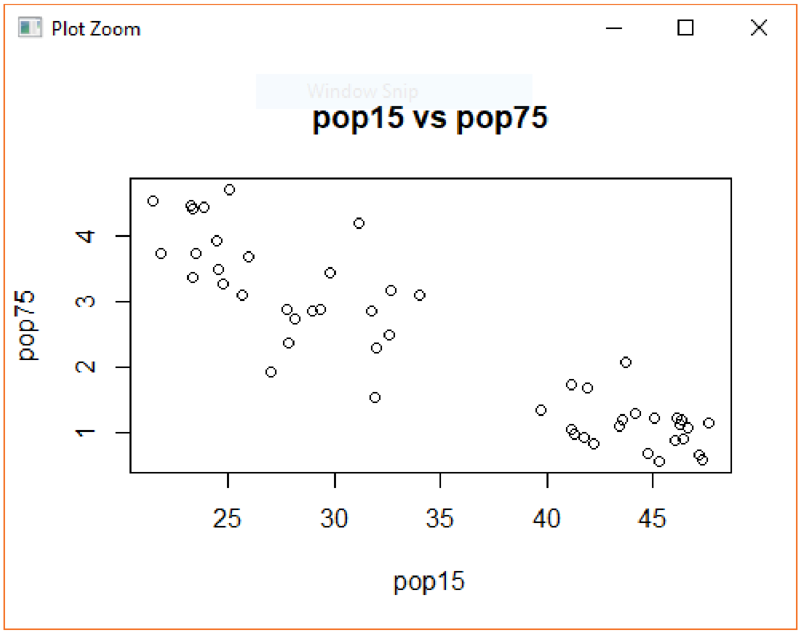
Part 2. Plot analysis

Run the following code to make a plot.
(don't worry right now about what this code is doing)

```
plot(LifeCycleSavings$pop15, LifeCycleSavings$pop75, xlab = 'pop15', ylab = 'pop75', main = 'pop15 vs pop75')
```

(1) Use the Zoom button in the Plots window to enlarge the plot. Resize the plot so that it is long and short, making it easier to read. Include an image of this plot in the homework you turn in. (1 pt)

Ans :



(2) Make an interesting observation about the relationship between pop15 and pop75 based on your plot. (something that you couldn't see with the calculations so far.) (1 pt)

Ans :

The relation between the two variables show negative correlation from the above graph which means that as the percentage of population under 15 increases the percentage of population over 75 decreases. This makes sense as the population distribution is over the entire age range and is hardly concentrated over and under both the age range.

(3) Based on our analysis so far, what interesting question about the LifeCycleSavings data would you like to answer, but don't yet know how to do it? (1 pt)

Ans :

The relationship between the pop75 and pop15 variables divides the data into two clusters. I would like to know what factors might have influenced this clustering of data.

Part 3. Random number generators

For the remainder of this assignment we will work with one of the random number generators in R.

(1) Use you UIN number to set the seed in the set.seed() function. (1 pt)

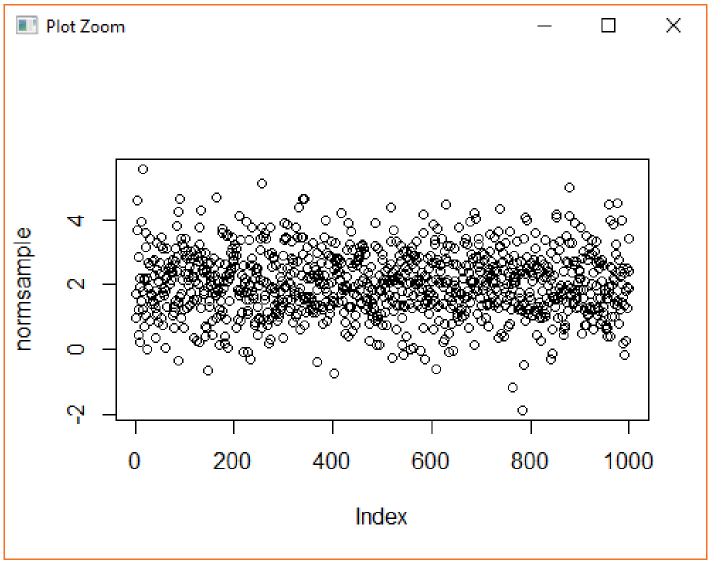
Ans :

set.seed(668936908)

(2) Generate a vector called "normsample" containing 1000 random samples from a normal distribution with mean 2 and standard deviation 1. (1 pt)

Ans :

```
normsample = rnorm(1000,2,1)
plot(normsample)
```



(3) Calculate the mean and standard deviation of the normsample. (2 pts)

Ans :

```
normsample_mean = mean(normsample)
normsample_mean
```

```
[1] 2.023946
```

```
normsample_sd = sd(normsample)
normsample_sd
```

```
[1] 0.9875672
```

The mean of the above normsample is 2.02 and the standard deviation is 0.99.

(4) Use logical operations (>,<,==,....) to calculate the fraction of the values in "normsample" that are more than 3. (1 pt)

Ans :

```
no_greaterthan3 = length(normsample[normsample>3])
fraction = (no_greaterthan3/length(normsample))
fraction
```

```
[1] 0.156
```

Running the above code gives me the fraction value as 0.156

(8). Find the area under the normal(2, 1) curve to the right of 3. This should be the probability of getting a random value more than 3. (Hint: Look up the help for rnorm. You will see a few other functions listed. Use one of them to figure out about what answer you should expect.)
What value do you expect?
What value did you get?
Why might they be different? (3 pts)

Ans :

```
right_area = pnorm(3, mean = 2, sd = 1, lower.tail = FALSE)
right_area
```

The value I expected was 0.158
The value I got is 0.156
The reason they are different is because of the chance factor. The expected value is a theoretical value for the given outcome. The real value corresponds to this value as it is closer to this value but may never the same as it because of the chance factor influencing the real value.