

EXERCISE 2

ANIL AKSU

PROBLEM 1

The use of basic statistic tools are incorporated in this exercise. For the given data set, the mean and the standard deviation is calculated. The mean is given as:

$$(1) \quad \mu = \frac{1}{N} \sum_1^N x_i.$$

and also

$$(2) \quad \sigma^2 = \frac{1}{N-1} \sum_1^N (x_i - \mu)^2.$$

Write an algorithm to calculate these values for the given data set.

Solution:

LISTING 1. Main Script

```
1 #####
2 #
3 #   Mean and Standard Deviation Calculation
4 #           by Anil Aksu
5 #   It is developed to show some basics of R
6 #
7 #####
8
9 ## library required to read excel files
10 require(gdata)
11 ## functions
12 source('getMean.R')
13 source('getSigma.R')
14 ## the random data read from excel file
15 RandomNumbers <- read.xls("RandomNumbers.xlsx", perl = ...
    "C:\\Perl\\bin\\perl.exe")
16 ## let's calculate the mean
17 Mean <- getMean(RandomNumbers)
18 ## the standard deviation
19 Sigma <- getSigma(RandomNumbers)
20 ## this function gets numbers from console
```

Date: May 11, 2017.

```

21
22 ## let's output them
23 print("Random Numbers")
24 print(RandomNumbers[,1])
25 print("The Mean")
26 print(Mean)
27 print("The Standard Deviation")
28 print(Sigma)

```

LISTING 2. Mean Calculating Function

```

29 getMean<- function (Numbers) {
30
31 #####
32 #
33 # This function calculates the mean of given
34 # random numbers
35 #
36 # Input: Random Numbers
37 #
38 # Output: Mean of random numbers
39 #
40 #####
41
42 ## the length of Number Array
43 numSize <- length(Numbers[,1]);
44 ## Mean of these numbers
45 Mean <- sum(Numbers)/numSize;
46 return (Mean)
47 }

```

LISTING 3. Standard Deviation Calculating Function

```

48 getSigma<- function (Numbers) {
49
50 #####
51 #
52 # This function calculates the mean of given
53 # random numbers
54 #
55 # Input: Random Numbers
56 #
57 # Output: standard deviation of random numbers
58 #
59 #####
60
61 ## the length of Number Array
62 numSize <- length(Numbers[,1]);
63 ## Mean of these numbers
64 Mean <- getMean(Numbers);
65

```

```
66 ## standard deviation
67 Sigma=0.
68 for (i in 1:numSize){
69     # this converts data into integer as.integer()
70     Sigma <- Sigma+Numbers[i,1]^2.
71 }
72 Sigma <- sqrt((Sigma/(numSize-1)));
73 return(Sigma)
```