MET CS544 A1 Foundation of Analytics
Homework#1

Problem#1:

The following sample data shows the scores of 10 students in an exam:
45, 80, 83, 78, 75, 77, 79, 83, 83, 100

Show detailed steps for each of the following (without using R):
a) Compute the mean, median, and mode.
b) Compute the variance and the standard deviation.
c) Compute the five number summary, the interquartile range, and outliers, if any.
d) Compute the variation of the data in the four quarters. Interpret the results.
e) Compute the standardized version (z-scores) of the above data.

Solution

a)The mean,median and the mode are calculated as

mean=(45+80+83+78+75+77+79+83+83+100)/10
=78.3
median=79.5
mode=83

b) Variance =sum $(x_i$-mean(x))^2 / (n-1) for all i =1 to n
= 184.6778
Standard Variation=sqrt(variance)
=13.58962

c) Five number summary, interquartile range, outliers

sorted_data = 45 75 77 78 79 80 83 83 83 100
First_quartile = 77
Second_quartile = 79.5
Third_quartile = 83
Interquartile= Third_quartile- First_quartile
=83-77=6
Outliers= 45 (in lower range) and 100 (in upper range)
Five number summary is (45, 77, 79.5, 83, 100)

d) Variation of data in four quarters

First quarter= Q1-minimum = 77 - 45 =32
Second quarter= Q2-Q1= 79.5 -77 =2.5
Third quarter =Q3-Q2 = 83 - 79.5 =3.5
Fourth quarter = maximum- Q3 = 100 - 83 =17

e) Standardized version (z-scores)

New_value= (value-mean(x)) / sd(x)

Old Value	45	80	83	78	75	77	79	83	83	100
New Value	-2.45	0.125	0.345	-0.022	-0.242	-0.095	0.0515	0.3458	0.3458	1.5968

Problem#2:

a) Using indexing, show the expression for accessing the first and last items. The code should not hardcode the value 10 for the number of items.

```
> #Loading the scores
> scores<-c(45, 80, 83, 78, 75, 77, 79, 83, 83, 100)
> scores
 [1]  45  80  83  78  75  77  79  83  83 100
> #a)Displaying first and last item of scores
> scores[c(1,length(scores))]
[1]  45 100
>
```

b) Using comparison operators, write the expression for scores less than the mean computed in 1a)

```
> #b)Booelan for the scores less than mean
> scores<mean(scores)
 [1]  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE
>
```

c) Using logical indexing and the expression from b), return all the scores less than the mean computed in 1a)

```
> scores[scores<mean(scores)]
[1] 45 78 75 77
```

d) Using rep function, create a sequence, as the same length as scores, of alternating TRUE, FALSE values. Using this sequence, return every other element from the scores. The code should not hardcode the value 10 for the number of scores. You can assume that there are even number of values in scores

```
> #d)Repetitive sequence of true and false and printing the scores with respect
to TRUE in rep.sequence
> (rep.sequence<-rep(c(TRUE,FALSE),5))
 [1]  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE
> scores[rep.sequence]
[1] 45 83 75 79 83
```

e) Using the paste function with LETTERS, show the code for the following output. The code should not hardcode the value 10 for the number of scores.

```
> #e)Pasting letters with scores
> paste(LETTERS[1:length(scores)],scores,sep="=")
 [1] "A=45"  "B=80"  "C=83"  "D=78"  "E=75"  "F=77"  "G=79"  "H=83"  "I=83"  "J=100"
>
```

f) Create a matrix of size 2 x 5 using the scores data. The first five values belong to the first row of the matrix. Assign the result to the variable, scores.matrix, and display the result.

```
> score.matrix<-matrix(scores,nrow=2,ncol=5,byrow = TRUE)
> score.matrix
     [,1] [,2] [,3] [,4] [,5]
[1,]   45   80   83   78   75
[2,]   77   79   83   83  100
>
```

g) Without hardcoding the value 5, show the code for displaying the first and last columns of the matrix.

```
> #g)Displaying the first and the last column in the matrix
> score.matrix[,c(1,ncol(score.matrix))]
     [,1] [,2]
[1,]   45   75
[2,]   77  100
>
```

h) Assign row names for the scores.matrix as Student_1, Student_2,… and column names as Quiz_1, Quiz_2 …. The code should not hard code the values 2 and 5.

```
> #h)Assigning the names for the rows and columns of the matrix
> dimnames(score.matrix)<-list(c("Student_1","Student_2"),c("Quiz_1","Quiz_2","Quiz_3","Quiz_4",
"Quiz_5"))
> score.matrix
          Quiz_1 Quiz_2 Quiz_3 Quiz_4 Quiz_5
Student_1     45     80     83     78     75
Student_2     77     79     83     83    100
>
```

Problem#3:

Using the data from Forbes America's top colleges (http://www.forbes.com/top-colleges/list/), create a data frame, say colleges.info, using the first five colleges using the column names: Name, State, Cost, Population. Use numbers for Cost and Population. You can use shorter names for colleges and abbreviations for State.

a) Show the above code and display the resulting data frame.

```
> #a)Creating and displaying the data frame
> college.name<-c("Pomona College","Williams College","Stanford University","Princeton University",
"Yale University")
> college.state<-c("California","Massachusetts","California","New Jersey","Connecticut")
> college.cost<-c(62632,64020,62801,58965,63970)
> college.population<-c(1610,2150,18346,8014,12109)
>
> college.info<-data.frame(college.name,college.state,college.cost,college.population)
> names(college.info)<-c("Name","State","Cost","Population")
> college.info
                 Name         State  Cost Population
1       Pomona College    California 62632       1610
2     Williams College Massachusetts 64020       2150
3  Stanford University    California 62801      18346
4 Princeton University    New Jersey 58965       8014
5      Yale University   Connecticut 63970      12109
```

b) Show the summary for State, Cost, and Population.

```
> #b)Summary of the state,cost and population
> summary(college.info$State)
   California   Connecticut Massachusetts    New Jersey
            2             1             1             1
> summary(college.info$Cost)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  58960   62630   62800   62480   63970   64020
> summary(college.info$Population)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1610    2150    8014    8446   12110   18350
```

c) Show the data frame sliced using the columns Name and Cost.

```
> #c)Displaying the name and cost
> college.info[c("Name","Cost")]
                   Name  Cost
1       Pomona College 62632
2     Williams College 64020
3  Stanford University 62801
4 Princeton University 58965
5       Yale University 63970
>
```

d) Show the data frame sliced using the first and last row. Do not hard code 5 in the expression.

```
> #d)Displaying the first and last column in the matrix
> college.info[c(1,ncol(college.info))]
                   Name Population
1       Pomona College       1610
2     Williams College       2150
3  Stanford University      18346
4 Princeton University       8014
5       Yale University      12109
>
```

e) Show all rows of the sliced data frame whose Population is greater than 5000.

```
> #e)Displaying the datarows whose population is greater than 5000
> subset(college.info,Population>5000)
                   Name       State  Cost Population
3  Stanford University  California 62801      18346
4 Princeton University  New Jersey 58965       8014
5       Yale University Connecticut 63970      12109
>
```

f) Modify the data frame for next year where the Cost increases by 5%. Round off to the nearest dollar. Display the new resulting data frame.

```
> #f)Modifying the cost in the data with 5% increase and rounded to nearest dollar
> cost.rise<-college.info
> cost.rise$Cost<-round(cost.rise$Cost+0.05*cost.rise$Cost)
> cost.rise
                   Name         State  Cost Population
1       Pomona College    California 65764       1610
2     Williams College Massachusetts 67221       2150
3  Stanford University    California 65941      18346
4 Princeton University    New Jersey 61913       8014
5       Yale University   Connecticut 67168      12109
>
```