

## VIT-AP UNIVERSITY, ANDHRA PRADESH

### CSE2047 – Data Analytics - Lab Sheet : 6

**Academic year:** 2020-2021

**Branch/ Class:** B.Tech/M.Tech

**Semester:** Fall

**Date:**

**Faculty Name:** Prof. S.Gopikrishnan

**School:** SCOPE

**Student name:** Valiveti Manikanta bhuvanesh

**Reg. no.:** 19BCD7088

---

#### LAB 6

#### Questions:

**1. Create a student result dataset with numeric values.**

**a. write a function for calculating the mean.**

**b. Write a function to compute std.deviation.**

```
df<-read.csv('Student_Data_cleaned.csv')
```

```
df
```

```
v<-colnames(df)
```

```
v<-v[c(6:12)]
```

```
for (i in v) {
```

```
  print(paste("mean of ",i,mean(df[,i])))
```

```
  print(paste("sd of ",i,sd(df[,i])))
```

```
}
```

```
## =====>=====
```

```
> v<-colnames(df)
```

```
> v<-v[c(6:12)]
```

```
>
```

```
> for (i in v) {
```

```
+   print(paste("mean of ",i,mean(df[,i])))
```

```
+   print(paste("sd of ",i,sd(df[,i])))
```

```
+ }
```

```
[1] "mean of  cat1 27.3859649122807"
```

```
[1] "sd of  cat1 9.81753967208855"
```

```
[1] "mean of  cat2 31.4649122807018"
```

```
[1] "sd of  cat2 9.47735065504049"
```

```
[1] "mean of  da01 18.0925925925926"
```

```
[1] "sd of  da01 2.94150422232352"
```

```
[1] "mean of  fat 21.5964912280702"
```

```
[1] "sd of  fat 10.3783078188574"
```

```
[1] "mean of  lab 62.0925925925926"
```

```
[1] "sd of  lab 17.6050606751485"
```

```
[1] "mean of  quiz1 16.0566037735849"
```

```
[1] "sd of  quiz1 4.36469232404278"
```

```
[1] "mean of  gt 65.6923066313583"
```

```
[1] "sd of  gt 14.6833000568838"
```

```
>
```

---

**2. Use Covid.csv and weather.csv. Do all observations (min, max, mean, variance, SD, range) in both dataframe.**

```
df<-read.csv('COVID_country_wise_latest.csv')

head(df)

df1<-read.csv('weatherHistory.csv')

head(df1)


v1<-colnames(df)

v2<-colnames(df1)

for(i in v1){

  if(typeof(df[,i])=="integer"){

    print(paste("mean of ",i,mean(df[,i],na.rm = TRUE)))

    print(paste("sd of ",i,sd(df[,i],na.rm = TRUE)))

    print(paste("min of ",i,min(df[,i],na.rm = TRUE)))

    print(paste("max of ",i,max(df[,i],na.rm = TRUE)))

    print(paste("range of ",i,range(df[,i],na.rm = TRUE)[1],range(df[,i],na.rm = TRUE)[2]))

    print(paste("var of ",i,var(df[,i],na.rm = TRUE)))

  }

}


for(i in v2){

  if(typeof(df1[,i])=="double"|typeof(df1[,i])=="integer"){

    print(paste("mean of ",i,mean(df1[,i],na.rm = TRUE)))

  }

}
```

```

print(paste("sd of ",i,sd(df1[,i],na.rm = TRUE)))

print(paste("min of ",i,min(df1[,i],na.rm = TRUE)))

print(paste("max of ",i,max(df1[,i],na.rm = TRUE)))

print(paste("range of ",i,range(df1[,i],na.rm = TRUE)[1],range(df1[,i],na.rm =
TRUE)[2]))

print(paste("var of ",i,var(df1[,i],na.rm = TRUE)))

}

}

```

```

> df<-read.csv('COVID_country_wise_latest.csv')
> head(df)
  Country.Region Confirmed Deaths Recovered Active New.Cases New.deaths New.recovered
1 Afghanistan    36263    1269    25198    9796    106        10        18
2 Albania         4880     144     2745    1991    117         6        63
3 Algeria        27973    1163    18837    7973    616         8       749
4 Andorra         907      52     803     52      10         0         0
5 Angola         950      41     242     667     18         1         0
6 Antigua and Barbuda 86       3      65     18         4         0         5
  Deaths...100.Cases Recovered...100.Cases Deaths...100.Recovered Confirmed.last.week X1.week.change
1 3.50 69.49 5.04 35526 737
2 2.95 56.25 5.25 4171 709
3 4.16 67.34 6.17 23691 4282
4 5.73 88.53 6.48 884 23
5 4.32 25.47 16.94 749 201
6 3.49 75.58 4.62 76 10
  X1.week...increase WHO.Region
1 2.07 Eastern Mediterranean
2 17.00 Europe
3 18.07 Africa
4 2.60 Europe
5 26.84 Africa
6 13.16 Americas
> df1<-read.csv('weatherHistory.csv')
> head(df1)
  Formatted.Date Summary Precip.Type Temperature..C. Apparent.Temperature..C.
1 2006-04-01 00:00:00.000 +0200 Partly cloudy rain 9.472222 7.388889
2 2006-04-01 01:00:00.000 +0200 Partly cloudy rain 9.355556 7.227778
3 2006-04-01 02:00:00.000 +0200 Mostly cloudy rain 9.377778 9.377778
4 2006-04-01 03:00:00.000 +0200 Partly cloudy rain 8.288889 5.944444
5 2006-04-01 04:00:00.000 +0200 Mostly cloudy rain 8.755556 6.977778
6 2006-04-01 05:00:00.000 +0200 Partly cloudy rain 9.222222 7.111111
  Humidity wind.Speed..km.h. wind.Bearing..degrees. visibility..km. Loud.Cover Pressure..millibars.
1 0.89 14.1197 251 15.8263 0 1015.13
2 0.86 14.2646 259 15.8263 0 1015.63
3 0.89 3.9284 204 14.9569 0 1015.94
4 0.83 14.1036 269 15.8263 0 1016.41
5 0.83 11.0446 259 15.8263 0 1016.51
6 0.85 13.9587 258 14.9569 0 1016.66
  Daily.Summary
1 Partly cloudy throughout the day.
2 Partly cloudy throughout the day.
3 Partly cloudy throughout the day.
4 Partly cloudy throughout the day.
5 Partly cloudy throughout the day.
6 Partly cloudy throughout the day.

```

```

> v1<-colnames(df)
> v2<-colnames(df1)
> for(i in v1){
+   if(typeof(df[,i])=="integer"){
+     print(paste("mean of ",i,mean(df[,i],na.rm = TRUE)))
+     print(paste("sd of ",i,sd(df[,i],na.rm = TRUE)))
+     print(paste("min of ",i,min(df[,i],na.rm = TRUE)))
+     print(paste("max of ",i,max(df[,i],na.rm = TRUE)))
+     print(paste("range of ",i,range(df[,i],na.rm = TRUE)[1],range(df[,i],na.rm = TRUE)[2]))
+     print(paste("var of ",i,var(df[,i],na.rm = TRUE)))
+   }
+ }
[1] "mean of Confirmed 88130.935828877"
[1] "sd of Confirmed 383318.663830615"
[1] "min of Confirmed 10"
[1] "max of Confirmed 4290259"
[1] "range of Confirmed 10 4290259"
[1] "var of Confirmed 146933198040.888"
[1] "mean of Deaths 3497.51871657754"
[1] "sd of Deaths 14100.0024820185"
[1] "min of Deaths 0"
[1] "max of Deaths 148011"
[1] "range of Deaths 0 148011"
[1] "var of Deaths 198810069.992927"
[1] "mean of Recovered 50631.4812834225"
[1] "sd of Recovered 190188.18964314"
[1] "min of Recovered 0"
[1] "max of Recovered 1846641"
[1] "range of Recovered 0 1846641"
[1] "var of Recovered 36171547479.7349"
[1] "mean of Active 34001.935828877"
[1] "sd of Active 213326.173371429"
[1] "min of Active 0"
[1] "max of Active 2816444"
[1] "range of Active 0 2816444"
[1] "var of Active 45508056245.2969"
[1] "mean of New.cases 1222.95721925134"
[1] "sd of New.cases 5710.37479028056"
[1] "min of New.cases 0"
[1] "max of New.cases 56336"
[1] "range of New.cases 0 56336"
[1] "var of New.cases 32608380.2454718"
[1] "mean of New.deaths 28.9572192513369"
[1] "sd of New.deaths 120.037172955508"
[1] "min of New.deaths 0"
[1] "max of New.deaths 1076"
[1] "range of New.deaths 0 1076"
[1] "var of New.deaths 14408.9228911506"
[1] "mean of New.recovered 933.812834224599"
[1] "sd of New.recovered 4197.71963468826"
[1] "min of New.recovered 0"
[1] "max of New.recovered 33728"
[1] "range of New.recovered 0 33728"
[1] "var of New.recovered 17620850.1314473"
[1] "mean of Confirmed.last.week 78682.4759358289"
[1] "sd of Confirmed.last.week 338273.676566537"
[1] "min of Confirmed.last.week 10"
[1] "max of Confirmed.last.week 3834677"
[1] "range of Confirmed.last.week 10 3834677"
[1] "var of Confirmed.last.week 114429080257.842"
[1] "mean of X1.week.change 9448.45989304813"
[1] "sd of X1.week.change 47491.1276840353"
[1] "min of X1.week.change -47"
[1] "max of X1.week.change 455582"
[1] "range of X1.week.change -47 455582"
[1] "var of X1.week.change 2255407208.70134"

```

```

> for(i in v2){
+   if(typeof(df1[,i])=="double"|typeof(df1[,i])=="integer"){
+     print(paste("mean of ",i,mean(df1[,i],na.rm = TRUE)))
+     print(paste("sd of ",i,sd(df1[,i],na.rm = TRUE)))
+     print(paste("min of ",i,min(df1[,i],na.rm = TRUE)))
+     print(paste("max of ",i,max(df1[,i],na.rm = TRUE)))
+     print(paste("range of ",i,range(df1[,i],na.rm = TRUE)[1],range(df1[,i],na.rm = TRUE)[2]))
+     print(paste("var of ",i,var(df1[,i],na.rm = TRUE)))
+   }
+ }
[1] "mean of Temperature..C. 11.8939959297555"
[1] "sd of Temperature..C. 9.67576675986136"
[1] "min of Temperature..C. -21.8222222222222"
[1] "max of Temperature..C. 39.9055555555556"
[1] "range of Temperature..C. -21.8222222222222 39.9055555555556"
[1] "var of Temperature..C. 93.620462391238"
[1] "mean of Apparent.Temperature..C. 10.8031166912851"
[1] "sd of Apparent.Temperature..C. 10.8291095754245"
[1] "min of Apparent.Temperature..C. -27.7166666666667"
[1] "max of Apparent.Temperature..C. 39.3444444444444"
[1] "range of Apparent.Temperature..C. -27.7166666666667 39.3444444444444"
[1] "var of Apparent.Temperature..C. 117.269614196551"
[1] "mean of Humidity 0.731187710487445"
[1] "sd of Humidity 0.196949231653658"
[1] "min of Humidity 0"
[1] "max of Humidity 1"
[1] "range of Humidity 0 1"
[1] "var of Humidity 0.0387889998489661"
[1] "mean of Wind.Speed..km.h. 10.8322161833522"
[1] "sd of Wind.Speed..km.h. 6.98879944722188"
[1] "min of Wind.Speed..km.h. 0"
[1] "max of Wind.Speed..km.h. 63.8526"
[1] "range of Wind.Speed..km.h. 0 63.8526"
[1] "var of Wind.Speed..km.h. 48.8433177134889"
[1] "mean of Wind.Bearing..degrees. 188.038830591794"
[1] "sd of Wind.Bearing..degrees. 107.776108745537"
[1] "min of Wind.Bearing..degrees. 0"
[1] "max of Wind.Bearing..degrees. 359"
[1] "range of Wind.Bearing..degrees. 0 359"
[1] "var of Wind.Bearing..degrees. 11615.6896163298"
[1] "mean of Visibility..km. 10.245467032993"
[1] "sd of Visibility..km. 4.11058209629388"
[1] "min of Visibility..km. 0"
[1] "max of Visibility..km. 16.1"
[1] "range of Visibility..km. 0 16.1"
[1] "var of Visibility..km. 16.8968851703718"
[1] "mean of Loud.Cover 0"
[1] "sd of Loud.Cover 0"
[1] "min of Loud.Cover 0"
[1] "max of Loud.Cover 0"
[1] "range of Loud.Cover 0 0"
[1] "var of Loud.Cover 0"
[1] "mean of Pressure..millibars. 1001.97239270183"
[1] "sd of Pressure..millibars. 122.188846500236"
[1] "min of Pressure..millibars. 0"
[1] "max of Pressure..millibars. 1046.38"
[1] "range of Pressure..millibars. 0 1046.38"
[1] "var of Pressure..millibars. 14930.1142090581"
>

```

### 3. Write a function that has three vector arguments for merging the into an existing dataframe.

```

merger<-function(a,b,c){
  df$dummy1=a
  df$dummy2=b
  df$dummy3=c
  head(df)
}
a=runif(dim(df)[1], min=1, max=50)
b=runif(dim(df)[1], min=6, max=40)
c=runif(dim(df)[1], min=9, max=30)
merger(a,b,c)

```

```

> merger<-function(a,b,c){
+   df$dummy1=a
+   df$dummy2=b
+   df$dummy3=c
+   head(df)
+ }
> a=runif(dim(df)[1], min=1, max=50)
> b=runif(dim(df)[1], min=6, max=40)
> c=runif(dim(df)[1], min=9, max=30)
> merger(a,b,c)

```

	Country.Region	Confirmed	Deaths	Recovered	Active	New.cases	New.deaths	New.recovered
1	Afghanistan	36263	1269	25198	9796	106	10	18
2	Albania	4880	144	2745	1991	117	6	63
3	Algeria	27973	1163	18837	7973	616	8	749
4	Andorra	907	52	803	52	10	0	0
5	Angola	950	41	242	667	18	1	0
6	Antigua and Barbuda	86	3	65	18	4	0	5

  

	Deaths...100.Cases	Recovered...100.Cases	Deaths...100.Recovered	Confirmed.last.week	X1.week.change
1	3.50	69.49	5.04	35526	737
2	2.95	56.25	5.25	4171	709
3	4.16	67.34	6.17	23691	4282
4	5.73	88.53	6.48	884	23
5	4.32	25.47	16.94	749	201
6	3.49	75.58	4.62	76	10

  

	X1.week...increase	WHO.Region	dummy1	dummy2	dummy3
1	2.07	Eastern Mediterranean	16.755009	24.996440	16.646871
2	17.00	Europe	47.429244	8.386632	17.816690
3	18.07	Africa	6.550106	17.928001	10.595239
4	2.60	Europe	34.230913	19.397457	23.176103
5	26.84	Africa	40.770837	38.533374	9.377674
6	13.16	Americas	38.597345	35.063173	16.558737

4. After merging create a function compute to find out min, max and avg of all numeric columns.

```

v3<-colnames(df)
for(i in v3){
  if(typeof(df[,i])=="integer"){
    print(paste("avg of ",i,mean(df[,i],na.rm = TRUE)))
    print(paste("min of ",i,min(df[,i],na.rm = TRUE)))
    print(paste("max of ",i,max(df[,i],na.rm = TRUE)))
  }
}

```

```

> v3<-colnames(df)
> for(i in v3){
+   if(typeof(df[,i])=="integer"){
+     print(paste("avg of ",i,mean(df[,i],na.rm = TRUE)))
+     print(paste("min of ",i,min(df[,i],na.rm = TRUE)))
+     print(paste("max of ",i,max(df[,i],na.rm = TRUE)))
+   }
+ }

```

```

[1] "avg of Confirmed 88130.935828877"
[1] "min of Confirmed 10"
[1] "max of Confirmed 4290259"
[1] "avg of Deaths 3497.51871657754"
[1] "min of Deaths 0"
[1] "max of Deaths 148011"
[1] "avg of Recovered 50631.4812834225"
[1] "min of Recovered 0"
[1] "max of Recovered 1846641"
[1] "avg of Active 34001.935828877"
[1] "min of Active 0"
[1] "max of Active 2816444"
[1] "avg of New.cases 1222.95721925134"
[1] "min of New.cases 0"
[1] "max of New.cases 56336"
[1] "avg of New.deaths 28.9572192513369"
[1] "min of New.deaths 0"
[1] "max of New.deaths 1076"
[1] "avg of New.recovered 933.812834224599"
[1] "min of New.recovered 0"
[1] "max of New.recovered 33728"
[1] "avg of Confirmed.last.week 78682.4759358289"
[1] "min of Confirmed.last.week 10"
[1] "max of Confirmed.last.week 3834677"
[1] "avg of X1.week.change 9448.45989304813"
[1] "min of X1.week.change -47"
[1] "max of X1.week.change 455582"

```

5. The summary values should be in a single data frame with the following columns: variable name, mean, sd, minimum, and maximum.

```
j=0
summary<-data.frame()
for(i in v3){
  if(typeof(df[,i])=="integer"){
    j=j+1
    summary[j,1]=i
    summary[j,2]=mean(df[,i],na.rm = TRUE)
    summary[j,3]=sd(df[,i],na.rm = TRUE)
    summary[j,4]=min(df[,i],na.rm = TRUE)
    summary[j,5]=max(df[,i],na.rm = TRUE)
  }
}
colnames(summary)<-c("name","mean","standard deviation","min","max")
summary
```

```
> j=0
> summary<-data.frame()
> for(i in v3){
+   if(typeof(df[,i])=="integer"){
+     j=j+1
+     summary[j,1]=i
+     summary[j,2]=mean(df[,i],na.rm = TRUE)
+     summary[j,3]=sd(df[,i],na.rm = TRUE)
+     summary[j,4]=min(df[,i],na.rm = TRUE)
+     summary[j,5]=max(df[,i],na.rm = TRUE)
+   }
+ }
> colnames(summary)<-c("name","mean","standard deviation","min","max")
> summary
```

	name	mean	standard deviation	min	max
1	Confirmed	88130.93583	383318.6638	10	4290259
2	Deaths	3497.51872	14100.0025	0	148011
3	Recovered	50631.48128	190188.1896	0	1846641
4	Active	34001.93583	213326.1734	0	2816444
5	New.cases	1222.95722	5710.3748	0	56336
6	New.deaths	28.95722	120.0372	0	1076
7	New.recovered	933.81283	4197.7196	0	33728
8	Confirmed.last.week	78682.47594	338273.6766	10	3834677
9	X1.week.change	9448.45989	47491.1277	-47	455582

6. Write a function so that the summary of the dataframe should be written to a csv file and to R.

```
writer<-function(df){
  write.csv(df,'summary.csv')
}
writer(summary)
df<-read.csv("summary.csv")
df
```

```
> writer<-function(df){
+   write.csv(df,'summary.csv')
+ }
> writer(summary)
> df<-read.csv("summary.csv")
> df
```

	name	mean	standard deviation	min	max
1	Confirmed	88130.93583	383318.6638	10	4290259
2	Deaths	3497.51872	14100.0025	0	148011
3	Recovered	50631.48128	190188.1896	0	1846641
4	Active	34001.93583	213326.1734	0	2816444
5	New.cases	1222.95722	5710.3748	0	56336
6	New.deaths	28.95722	120.0372	0	1076
7	New.recovered	933.81283	4197.7196	0	33728
8	Confirmed.last.week	78682.47594	338273.6766	10	3834677
9	X1.week.change	9448.45989	47491.1277	-47	455582