VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engineering

CSE3046-Programming for Data Science

Process: Obtain Data from Various Resources

Lab Assignment-4

Statisical Methods and Hypothesis

NAME :D VASANTH KUMAR

REG NO: 19BDS0083

Submitted Date: 15/10/2021

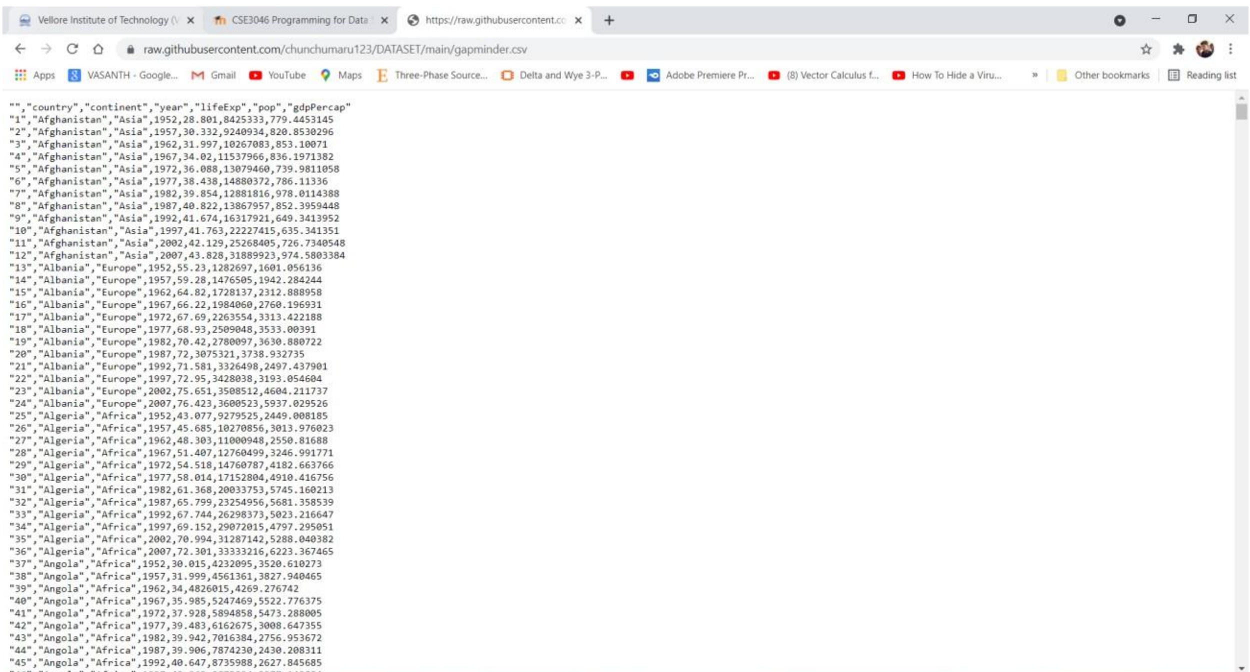
Course Instructor: Dr. Anthoniraj A

Retrieve the data set from the following URL,

https://raw.githubusercontent.com/chunchumaru123/DATASET/main/gapminder.csv

STEP 1:

THE DATA SET:



```
"","country","continent","year","lifeExp","pop","gdpPercap"
"1","Afghanistan","Asia",1952,28.801,8425333,779.4453145
"2","Afghanistan","Asia",1957,30.332,9240934,820.8530296
"3","Afghanistan","Asia",1962,31.997,10267083,853.10071
"4","Afghanistan","Asia",1967,34.02,11537966,836.1971382
"5","Afghanistan","Asia",1972,36.088,13079460,739.9811058
"6","Afghanistan","Asia",1977,38.438,14880372,786.11336
"7","Afghanistan","Asia",1982,39.854,12881816,978.0114388
"8","Afghanistan","Asia",1987,40.822,13867957,852.3959448
"9","Afghanistan","Asia",1992,41.674,16317921,649.3413952
"10","Afghanistan","Asia",1997,41.763,22227415,635.341351
"11","Afghanistan","Asia",2002,42.129,25268405,726.7340548
"12","Afghanistan","Asia",2007,43.828,31889923,974.5803384
"13","Albania","Europe",1952,55.23,1282697,1601.056136
"14","Albania","Europe",1957,59.28,1476505,1942.284244
"15","Albania","Europe",1962,64.82,1728137,2312.888958
"16","Albania","Europe",1967,66.22,1984060,2760.196931
"17","Albania","Europe",1972,67.69,2263554,3313.422188
"18","Albania","Europe",1977,68.93,2509048,3533.00391
"19","Albania","Europe",1982,70.42,2780097,3630.880722
"20","Albania","Europe",1987,72,3075321,3738.932735
"21","Albania","Europe",1992,71.581,3326498,2497.437901
"22","Albania","Europe",1997,72.95,3428038,3193.054604
"23","Albania","Europe",2002,75.651,3508512,4604.211737
"24","Albania","Europe",2007,76.423,3600523,5937.029526
"25","Algeria","Africa",1952,43.077,9279525,2449.008185
"26","Algeria","Africa",1957,45.685,10270856,3013.976023
"27","Algeria","Africa",1962,48.303,11000948,2550.81688
"28","Algeria","Africa",1967,51.407,12760499,3246.991771
"29","Algeria","Africa",1972,54.518,14760787,4182.663766
"30","Algeria","Africa",1977,58.014,17152804,4910.416756
"31","Algeria","Africa",1982,61.368,20033753,5745.160213
"32","Algeria","Africa",1987,65.799,23254956,5681.358539
"33","Algeria","Africa",1992,67.744,26298373,5023.216647
"34","Algeria","Africa",1997,69.152,29072015,4797.295051
"35","Algeria","Africa",2002,70.994,31287142,5288.040382
"36","Algeria","Africa",2007,72.301,33333216,6223.367465
"37","Angola","Africa",1952,30.015,4232095,3520.610273
"38","Angola","Africa",1957,31.999,4561361,3827.940465
"39","Angola","Africa",1962,34,4826015,4269.276742
"40","Angola","Africa",1967,35.985,5247469,5522.776375
"41","Angola","Africa",1972,37.928,5894858,5473.288005
"42","Angola","Africa",1977,39.483,6162675,3008.647355
"43","Angola","Africa",1982,39.942,7016384,2756.953672
"44","Angola","Africa",1987,39.906,7874230,2430.208311
"45","Angola","Africa",1992,40.647,8735988,2627.845685
```

STEP 2:

The aim of this activity is to understand the various statistical methods and empirical rule. You can use the same data set which was used for Missing Data Imputation. You are expected to analyze the data without using any predefined R functions (You have to use user-defined functions for ALL methods expect plots and graphs - E.g. Do not use predefined function mean() instead of writing your own function with simple loop class_mean())

1. Find Mean (0.5)
2. Find Median (0.5)
3. Find Mode (1)
4. Find IQR (1)
5. Find Standard Deviation (1)
6. Find Probability values on Empirical Rule (1)
7. Plot the Graph/Histogram/Normal Distribution and Compare your functions return value with predefined functions in R for mean, median, IQR, and sd. (1)
8. Formulate the Null Hypothesis and Alternative Hypothesis for your data set and prove it based on the p-value. (3 Marks)

SAMPLE CODE:

19BDS0083.R

```r

#------------------------------------------------------------------------------------------------------------------
#WRITING META DATA

#USER INFROMATION : 19BDS0083 D VASANTH KUMAR

#DATA SOURCE : chunchumaru123/DATASET/gapminder.csv

#Description: Per-capita GDP (Gross domestic product) is given in units of international dollars, "a hypot

#Data shape:    1704 rows and 7 columns
#TAGS FOR THE DATA SET : MULTIVARIATE,TIME-SERIES,STATISTICAL DATA

#------------------------------------------------------------------------------------------------------------------
#LIBRARIES USED
library(rvest)
library(dplyr)
library(tidyr)
library(utils)
library(ggplot2)

#READING THE DATA SET
gap <- read.csv("https://raw.githubusercontent.com/chunchumaru123/DATASET/main/gapminder.csv")
gap <-gap[2:7]

#STORING THE DATA SET IN NEW VARIABLE KEEPS THE ORIGINALITY OF THE DATA SET BEFORE CLEANING
df1 <-data.frame(gap)

View(df1)

str(df1)
summary(df1)
attach(df1)
#---------------------------------------------------------------------------
#CALCULATING MEAN OF A COLUMN
#FUNCITON TO CALCULATE MEAN
get_mean <- function(df1)
{
  b=0
  n=0
  for(i in df1$pop)
  {
    b <- b+i
    n=n+1
  }
  mean_1=b/n
  return(mean_1)
}
#Main Function Calling- User defined
get_mean1 <-get_mean(df1)
print("Mean:")
print(get_mean1)
#checking mean with inbuilt Function
```

```r
#
#CALCULATING MEDIAN OF A COLUMN

n=dim(df1)[1]
#THE COLUMN IS SORTED USING INBUILT FUNCTION
s_df1=sort(df1$pop)
#FUNCITON TO CALCULATE MEDIAN
get_median <- function(s_df1)
{
  if(n%%2==0)
  {
    median1=s_df1[n/2]
    median2=s_df1[(n-1)/2]
    medians=(median1+median2)/2
  }else{
    medians=s_df1[n/2]
  }
  return(medians)
}
#Main Function Calling- User defined
get_median <- get_median(s_df1)
print(paste("Median:",get_median))
#checking median inbuilt Function
median(s_df1)
#----------------------------------------------------------------------------
#CALCULATING MODE OF A COLUMN
#FUNCITON TO CALCULATE MODE
Mode <- function(x) {
  ux <- unique(x)
  mode1 <- ux[which.max(tabulate(match(x, ux)))]
  return(mode1)
}

x=df1$pop
#Main Function Calling- User defined
get_mode <- Mode(x)
print(paste("Mode",get_mode))

#Testing with inbuilt Function
#NO BUILT-IN FUNCTION FOR MODE
#----------------------------------------------------------------------------
#CALCULATING INTER QUARTILE RANGE OF A COLUMN
#INITIALIZING THE GLOBAL VARIABLES
x=df1$pop
n=dim(df1)[1]
z=0
r=0
#THE COLUMN POP AFTER SORTING
x1=sort(x)
```

SAMPLE DATA SET:

df6 × Untitled1* × STATISTICAL.R × MANAGING_N_VALUES.R × Untitled2* × chi × ll_asia × ll_asia_europe × df_dup × df1 ×

Filter

	country	continent	year	lifeExp	pop	gdpPercap
1	Afghanistan	Asia	1952	28.801	8425333	779.4453
2	Afghanistan	Asia	1957	30.332	9240934	820.8530
3	Afghanistan	Asia	1962	31.997	10267083	853.1007
4	Afghanistan	Asia	1967	34.020	11537966	836.1971
5	Afghanistan	Asia	1972	36.088	13079460	739.9811
6	Afghanistan	Asia	1977	38.438	14880372	786.1134
7	Afghanistan	Asia	1982	39.854	12881816	978.0114
8	Afghanistan	Asia	1987	40.822	13867957	852.3959
9	Afghanistan	Asia	1992	41.674	16317921	649.3414
10	Afghanistan	Asia	1997	41.763	22227415	635.3414
11	Afghanistan	Asia	2002	42.129	25268405	726.7341
12	Afghanistan	Asia	2007	43.828	31889923	974.5803
13	Albania	Europe	1952	55.230	1282697	1601.0561
14	Albania	Europe	1957	59.280	1476505	1942.2842
15	Albania	Europe	1962	64.820	1728137	2312.8890
16	Albania	Europe	1967	66.220	1984060	2760.1969
17	Albania	Europe	1972	67.690	2263554	3313.4222
18	Albania	Europe	1977	68.930	2509048	3533.0039
19	Albania	Europe	1982	70.420	2780097	3630.8807
20	Albania	Europe	1987	72.000	3075321	3738.9327
21	Albania	Europe	1992	71.581	3326498	2497.4379
22	Albania	Europe	1997	72.950	3428038	3193.0546
23	Albania	Europe	2002	75.651	3508512	4604.2117
24	Albania	Europe	2007	76.423	3600523	5937.0295
25	Algeria	Africa	1952	43.077	9279525	2449.0082
26	Algeria	Africa	1957	45.685	10270856	3013.9760
27	Algeria	Africa	1962	48.303	11000948	2550.8169
28	Algeria	Africa	1967	51.407	12760499	3246.9918
29	Algeria	Africa	1972	54.518	14760787	4182.6638
30	Algeria	Africa	1977	58.014	17152804	4910.4168
31	Algeria	Africa	1982	61.368	20033753	5745.1602

Showing 1 to 31 of 1,704 entries, 6 total columns

CODE IN R STUDIO:

#____

#WRITING META DATA

#USER INFROMATION : 19BDS0083 D VASANTH KUMAR

#DATA SOURCE : chunchumaru123/DATASET/gapminder.csv

#Description: Per-capita GDP (Gross domestic product) is given in units of international dollars, “a hypothetical unit of currency that has the same purchasing power parity that the U.S. dollar had in the United States at a given point in time” – 2005, in this case.

#Data shape: 1704 rows and 7 columns

#TAGS FOR THE DATA SET : MULTIVARIATE,TIME-SERIES,STATISTICAL DATA

#____

```
#LIBRARIES USED

library(rvest)

library(dplyr)

library(tidyr)

library(utils)

library(ggplot2)

#READING THE DATA SET

gap <-
read.csv("https://raw.githubusercontent.com/chunchumaru123/DATAS
ET/main/gapminder.csv")

gap <-gap[2:7]

#STORING THE DATA SET IN NEW VARIABLE KEEPS THE ORIGINALITY OF
THE DATA SET BEFORE CLEANING

df1 <-data.frame(gap)

View(df1)

str(df1)

summary(df1)

attach(df1)

#____________________________________________________

#CALCULATING MEAN OF A COLUMN

#FUNCITON TO CALCULATE MEAN
```

```
get_mean <- function(df1)
{
  b=0
  n=0
  for(i in df1$pop)
  {
    b <- b+i
    n=n+1
  }
  mean_1=b/n
  return(mean_1)
}
#Main Function Calling- User defined
get_mean1 <-get_mean(df1)
print("Mean:")
print(get_mean1)
#checking mean with inbuilt Function
print(paste("Mean(inbuilt-function):",round(mean(df1$pop),0)))
#________________________________________________________
```

#CALCULATING MEDIAN OF A COLUMN

```
n=dim(df1)[1]
#THE COLUMN IS SORTED USING INBUILT FUNCTION
s_df1=sort(df1$pop)
#FUNCITON TO CALCULATE MEDIAN
get_median <- function(s_df1)
{
  if(n%%2==0)
  {
    median1=s_df1[n/2]
    median2=s_df1[(n-1)/2]
    medians=(median1+median2)/2
  }else{
    medians=s_df1[n/2]
  }
  return(medians)
}
#Main Function Calling- User defined
get_median <- get_median(s_df1)
```

```
print(paste("Median:",get_median))

#checking median inbuilt Function

median(s_df1)

#____________________________________________________________

#CALCULATING MODE OF A COLUMN

#FUNCITON TO CALCULATE MODE

Mode <- function(x) {

  ux <- unique(x)

  mode1 <- ux[which.max(tabulate(match(x, ux)))]

  return(mode1)

}


x=df1$pop

#Main Function Calling- User defined

get_mode <- Mode(x)

print(paste("Mode",get_mode))


#Testing with inbuilt Function

#NO BUILT-IN FUNCTION FOR MODE

#____________________________________________________________
```

```
#CALCULATING INTER QUARTILE RANGE OF A COLUMN

#INITIALIZING THE GLOBAL VARIABLES

x=df1$pop

n=dim(df1)[1]

z=0

r=0

#THE COLUMN POP AFTER SORTING

x1=sort(x)

#Function to give index of the median

median1 <- function(x1,a,n){

  z = n - a + 1

  z = (z + 1) / 2 - 1

  return(z + a)

}

#Function to give iqr

iqr <-function(x,n){


  #Index of median of entire data

  mid_index=median1(x1,0,n)

  #Median of first half
```

```
  Q1 <- x1[median1(x1,0,mid_index)]
  #Median of second half
  Q3 <-x1[median1(x1, mid_index, n)]
  #IQR calculation
  return(Q3-Q1)
}
#Main Function Calling- User defined
print(paste("IQR(user-defined func):",iqr(x,n)))
#Testing with inbuilt Function
IQR(x1)
#______________________________________________

#CALCULATING STANDARD DEVIATION OF A COLUMN
n=dim(df1)[1]
#FUNCITON TO CALCULATE SD
calculatesd <- function(x)
{
  sum=0.0
  SD=0.0
  mean_new=get_mean1
```

```
  for(i in x)
  {
    SD=SD+(i-mean_new)^2
  }
  return(sqrt(SD/n))
}
#Main Function Calling- User defined
get_sd <-calculatesd(df1$pop)
print(paste("Standard Deviation:",round(calculatesd(df1$pop),0)))
#Testing with inbuilt Function
sd(df1$pop)
#________________________________________________________________

#Empirical Rule Check
#68%, 95%, and 99.7% Sigma rule

the.mean=mean(df1$pop)
the.sd=get_sd
#calculate the lower and upper bounds:
lower.bounds = the.mean - 1:3*the.sd
```

```
upper.bounds = the.mean + 1:3*the.sd


#calculate the proportion of observations between each pair of the upper and lower bounds

one.sd = mean(df1$pop > lower.bounds[1] & df1$pop < upper.bounds[1]) #68%

two.sd = mean(df1$pop > lower.bounds[2] & df1$pop < upper.bounds[2]) #95%

three.sd = mean(df1$pop > lower.bounds[3] & df1$pop < upper.bounds[3]) #99.7%


#______________________________________________________

#Histogram

hist(df1$pop)

# after seeing the histogram it is obvious that the data is positively skewed.Hence to reduce it log is used.

#A log transformation is a process of applying a logarithm to data to reduce its skew.usually done when the numbers are highly skewed to reduce the skew so the data can be understood easier


hist(log(df1$pop),main = "population plot ",col = "darkmagenta")
```

```
#Normal Probability Plot

qqnorm(log(df1$pop),main = "Normal Probability Plot population")

qqline(log(df1$pop))

#___________________________________________________________

#LINE GRAPH


plot(log(df1$pop),type = "o",main="population chart",col="RED",xlim =
c(0,200))

#___________________________________________________________

#NORMAL DISTRIBUTION CURVE


x <- log(df1$pop)

y <-dnorm(x,mean=get_mean1,sd=get_sd)


plot(x,y,col="darkmagenta")

#___________________________________________________________


#HYPOTHESIS TESTING


#performing two-tailed-t-test
```

#Step 1: State the null and alternate hypothesis

(null hypothesis) H0: the difference in the lifeExp of countries Ireland and south Africa is zero

(alternate hypothesis) H1: there exists a difference between lifeExp of countries Ireland and south Africa

#Step 2: Collect data-creating new data frame

```
##Filtering the data by country of interest South africa and Ireland

df_dup <- df1 %>% select(country,lifeExp)%>%

filter(country=="South Africa" | country=="Ireland")%>%

group_by(country)
```

#Step 3: Perform a statistical test

```
##Using t.test

t.test(data=df_dup,lifeExp ~country)
```

#Step 4: Decide whether the null hypothesis is supported or refuted

#the Result tells us the average lifeExp in Ireland and South Africa is 73 years and 53 years respectively

with a difference of 20 years

Since the P- value is close to zero,it is unlikely that null hypothesis will happen

#So there is exists a difference between lifeExp for the countries according to alternate hypothesis

#we can see that the difference in means for our sample data is 73.01725 and 53.99317 , and the confidence interval shows that the true difference

#in means is between 15.07022 and 22.97794. So, 95% of the time, the true difference in means will be different from 0. Our p-value of 4.466e-09 is

#much smaller than 0.05, so we can reject the null hypothesis of no difference and say with a high degree of confidence that the true difference in means is not equal to zero.

#So null hypothesis is rejected

SAMPLE OUTPUT:

df6 × | Untitled1* × | STATISTICAL.R × | MANAGING_N_VALUES.R ×

Filter

	country	lifeExp
1	Ireland	66.910
2	Ireland	68.900
3	Ireland	70.290
4	Ireland	71.080
5	Ireland	71.280
6	Ireland	72.030
7	Ireland	73.100
8	Ireland	74.360
9	Ireland	75.467
10	Ireland	76.122
11	Ireland	77.783
12	Ireland	78.885
13	South Africa	45.009
14	South Africa	47.985
15	South Africa	49.951
16	South Africa	51.927
17	South Africa	53.696
18	South Africa	55.527
19	South Africa	58.161
20	South Africa	60.834
21	South Africa	61.888
22	South Africa	60.236
23	South Africa	53.365
24	South Africa	49.339

Showing 1 to 24 of 24 entries, 2 total columns

```
>
> #HYPOTHESIS TESTING
>
> #performing two-tailed-t-test
> #Step 1: State the null and alternate hypothesis
>
> # (null hypothesis) H0: the difference in the lifeExp of countries Ireland and south Africa is zero
> # (alternate hypothesis) H1: there exists a difference between lifeExp of countries Ireland and south Africa
>
> #Step 2: Collect data-creating new data frame
>
> ##Filtering the data by country of interest South africa and Ireland
> df_dup <- df1 %>% select(country,lifeExp)%>%
+   filter(country=="South Africa" | country=="Ireland")%>%
+   group_by(country)
>
> #Step 3: Perform a statistical test
>
> ##Using t.test
> t.test(data=df_dup,lifeExp ~country)

        Welch Two Sample t-test

data:  lifeExp by country
t = 10.067, df = 19.109, p-value = 4.466e-09
alternative hypothesis: true difference in means between group Ireland and group South Africa is not equal to 0
95 percent confidence interval:
 15.07022 22.97794
sample estimates:
     mean in group Ireland mean in group South Africa
                  73.01725                   53.99317


>
> #Step 4: Decide whether the null hypothesis is supported or refuted
>
> #the Result tells us the average lifeExp in Ireland and South Africa is 73 years and 53 years respectively
> # with a difference of 20 years
>
> # Since the P- value is close to zero,it is unlikely that null hypothesis will happen
> #So there is exists a difference between lifeExp for the countries according to alternate hypothesis
>
> #we can see that the difference in means for our sample data is 73.01725 and 53.99317  , and the confidence interval shows that the true difference
> #in means is between 15.07022 and 22.97794. So, 95% of the time, the true difference in means will be different from 0. Our p-value of  4.466e-09 is
> #much smaller than 0.05, so we can reject the null hypothesis of no difference and say with a high degree of confidence that the true difference in means is not equal to zero.
>
> #So null hypothesis is rejected
>
```