20BDS0146

VENNELA G

DATA VISUALIZATION & PRESENTATION LAB

LAB SLOT: L31+L32

LAB ASSIGNMENT 1

DATE: 30-01-2023

Ex.No.1

TITLE OF EXPERIMENT: Acquiring and plotting data

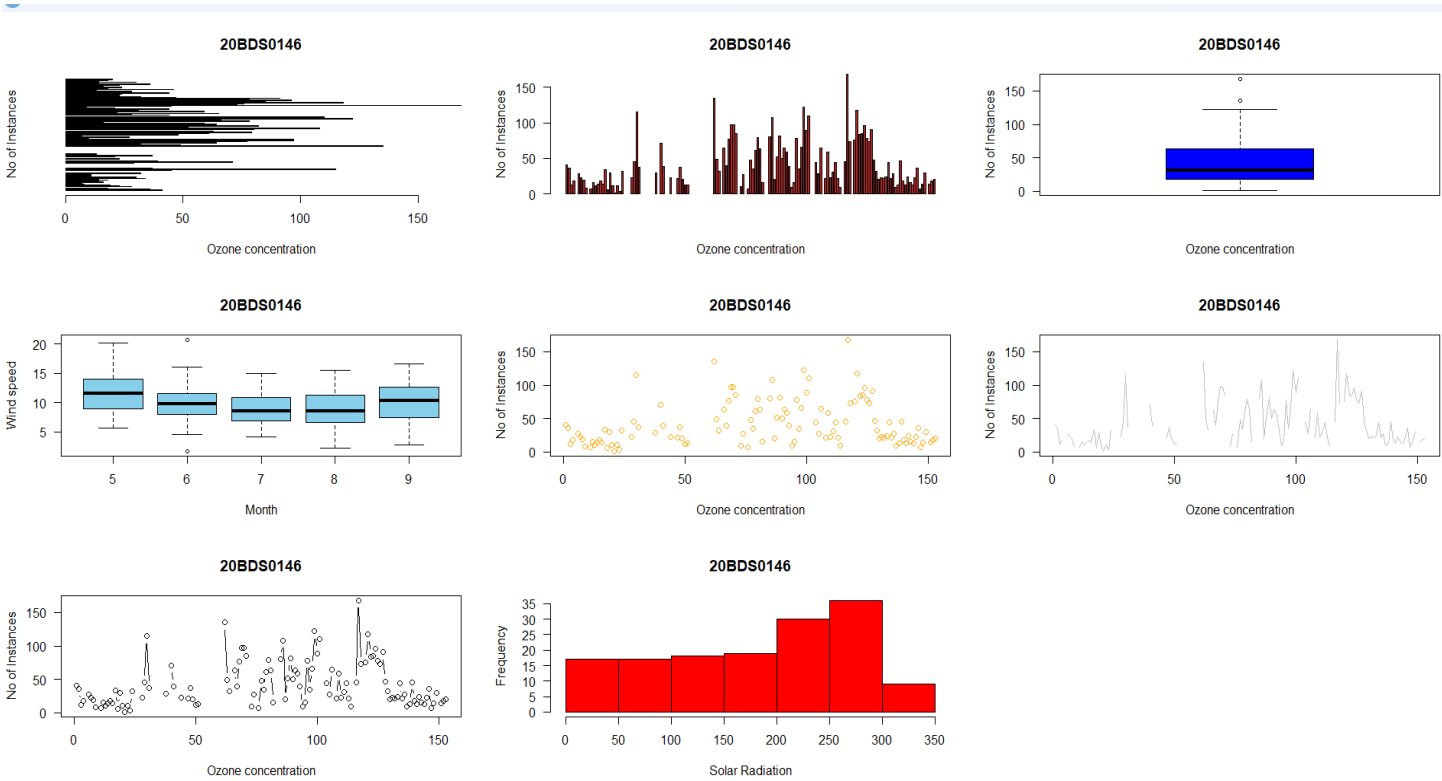
1. **AIM: Load the in-build dataset from R and draw various basic plot in R using grid (Horizontal bar plot, Vertical bar plot, box plot, multiple box plot, plot with point a line etc.,)**

CODE:

```
install.packages("lattice")

install.packages("ggplot2")

library(lattice)

library(ggplot2)

data("airquality")

View(airquality)

summary(airquality)

str(airquality)

par(mfrow=c(3,3),las=1,bg="white")

barplot(airquality$Ozone,xlab='Ozone concentration',ylab='No of Instances',main='20BDS0146',col='blue',horiz=T)

barplot(airquality$Ozone,xlab='Ozone concentration',ylab='No of Instances',main='20BDS0146',col='brown')
```

```
boxplot(airquality$Ozone,xlab='Ozone concentration',ylab='No of Instances',main='20BDS0146',col='blue')

boxplot(Wind~Month,data=airquality,xlab='Month',ylab='Wind speed',main='20BDS0146',col='skyblue')

plot(airquality$Ozone,type='p',xlab='Ozone concentration',ylab='No of Instances',main='20BDS0146',col='orange')

plot(airquality$Ozone,type='l',xlab='Ozone concentration',ylab='No of Instances',main='20BDS0146',col='grey')

plot(airquality$Ozone,type='b',xlab='Ozone concentration',ylab='No of Instances',main='20BDS0146')

hist(airquality$Solar.R,main="20BDS0146",xlab='Solar Radiation',col='red')

dev.off()
```

OUTPUT:



RESULT

We have loaded the in-build dataset airquality from R and have drawn various basic plots in R using grid (Horizontal bar plot, Vertical bar plot, box plot, multiple box plot, plot with point a line etc.,). We have estimated the Ozone concentration in the city and various parameters related to Ozone concentration.

2. AIM: Load in-build dataset mtcars and visualize data using visualization library ggplot

CODE:

mtcars

```
str(mtcars)
mtcars$cyl
#Visualization using libraries
install.packages("lattice")
install.packages("ggplot2")#grammar of graphics plot- 2nd version
library(lattice)
library(ggplot2)

#using factor
str(mtcars)#data entries .. means total 32 entries
View(mtcars)
factor(mtcars$gear,levels=c(3,4,5),labels=c('3Gear','4Gear','5Gear'))#to find unique numbers present in a column
#using ggplot and plot analog
plot(mtcars$wt,mtcars$mpg,main='20BDS0146')
ggplot(data=mtcars,aes(x=wt,y=mpg))+geom_point()+ggtitle("20BDS0146")
 #geometricpoint specifies scatter plot
ggplot(data=mtcars,aes(x=wt,y=mpg),main='20BDS0146')+geom_line()+xlab('Weight')+ylab('Mileage')+ggtitle("20BDS0146")

ggplot(data=mtcars,aes(x=wt,y=mpg,color=as.factor(gear)),main='20BDS0146')+geom_line()+xlab('Weight')+ylab('Mileage')+ggtitle("20BDS0146")
```

ggplot(data=mtcars,aes(x=wt,y=mpg,color=as.factor(cyl)),main='20BDS0146')+geom_line()+xlab('Weight')+ylab('Mileage')+ggtitle("20BDS0146")

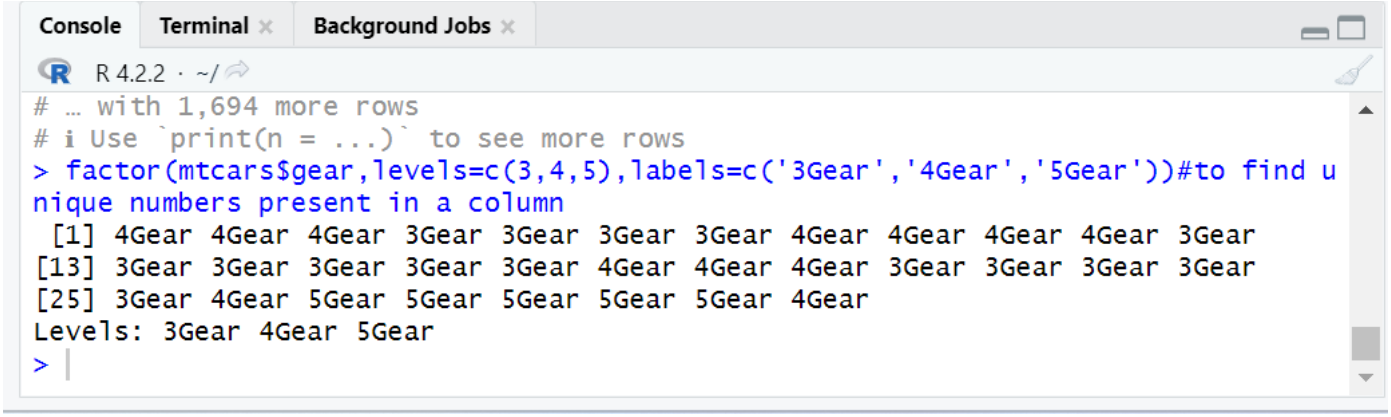
ggplot(data=mtcars,aes(x=wt,y=mpg,color=as.factor(cyl),size=qsec),main='20BDS0146')+geom_point()+xlab('Weight')+ylab('Mileage')+ggtitle("20BDS0146")

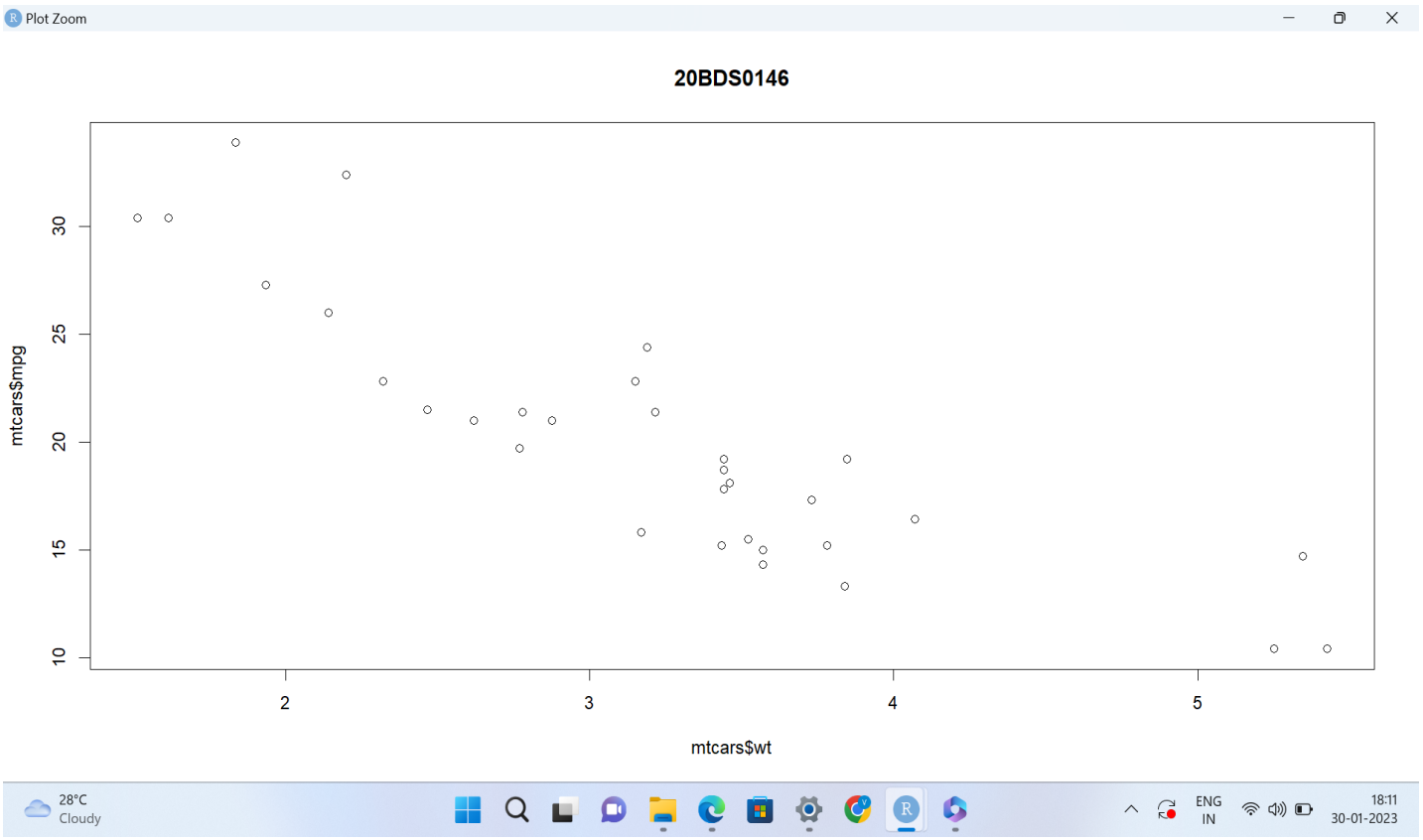
#colordisplays different colours for different unique cylinders

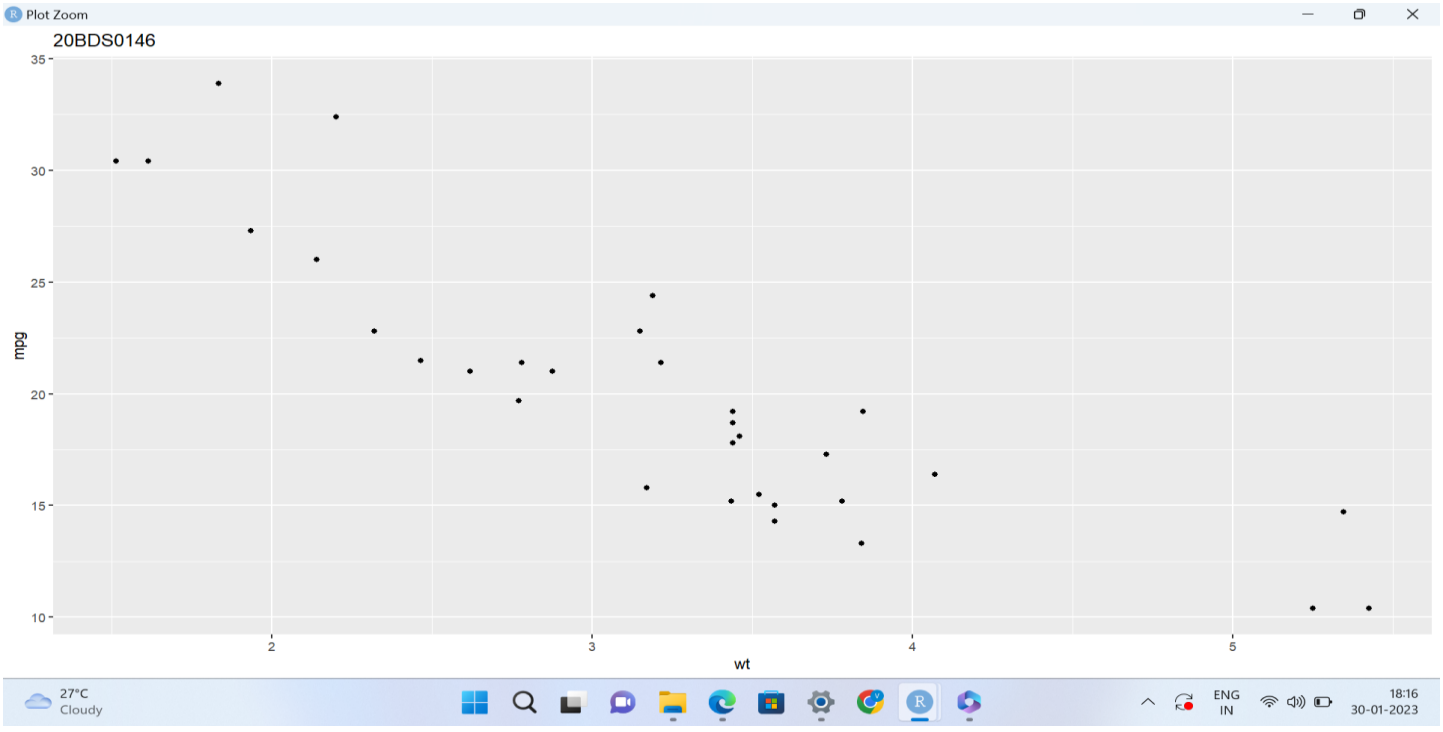
#size qsec means if size more qsec more and if size less then qsec less

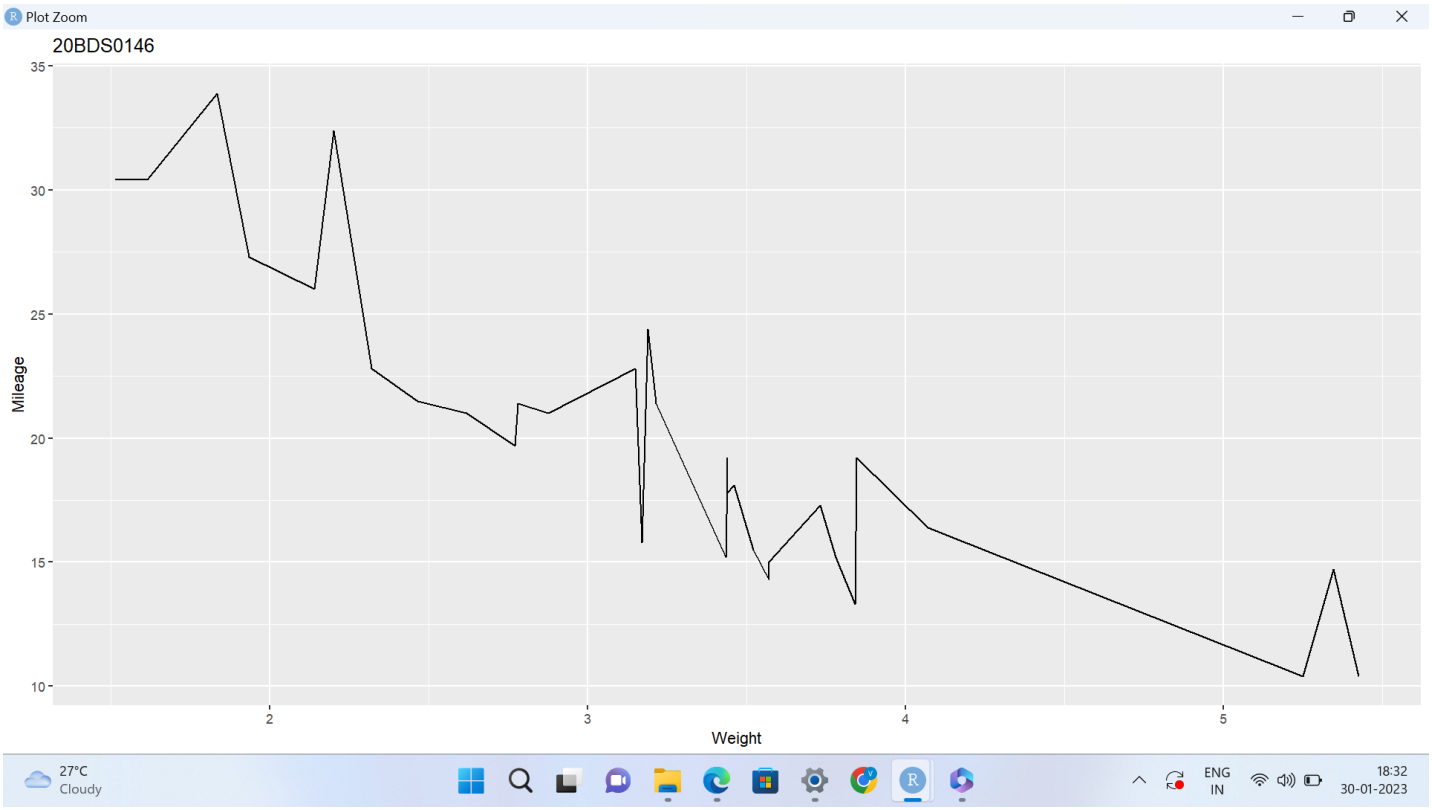
ggplot(data=mtcars,aes(x=wt,y=mpg,shape=as.factor(cyl),color=as.factor(cyl),size=qsec))+geom_point()+xlab('Weight')+ylab('Mileage')+ggtitle("20BDS0146")

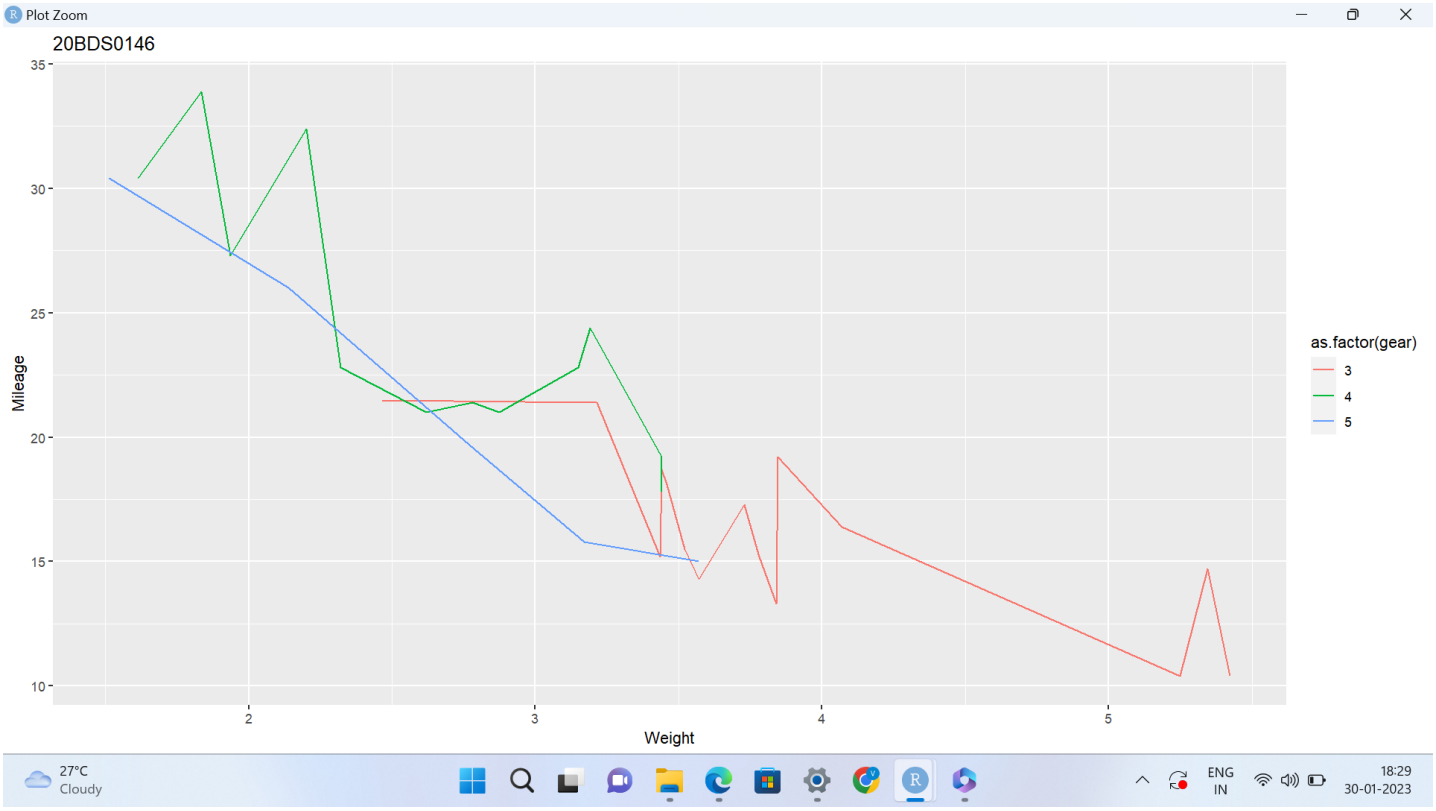
OUTPUT:

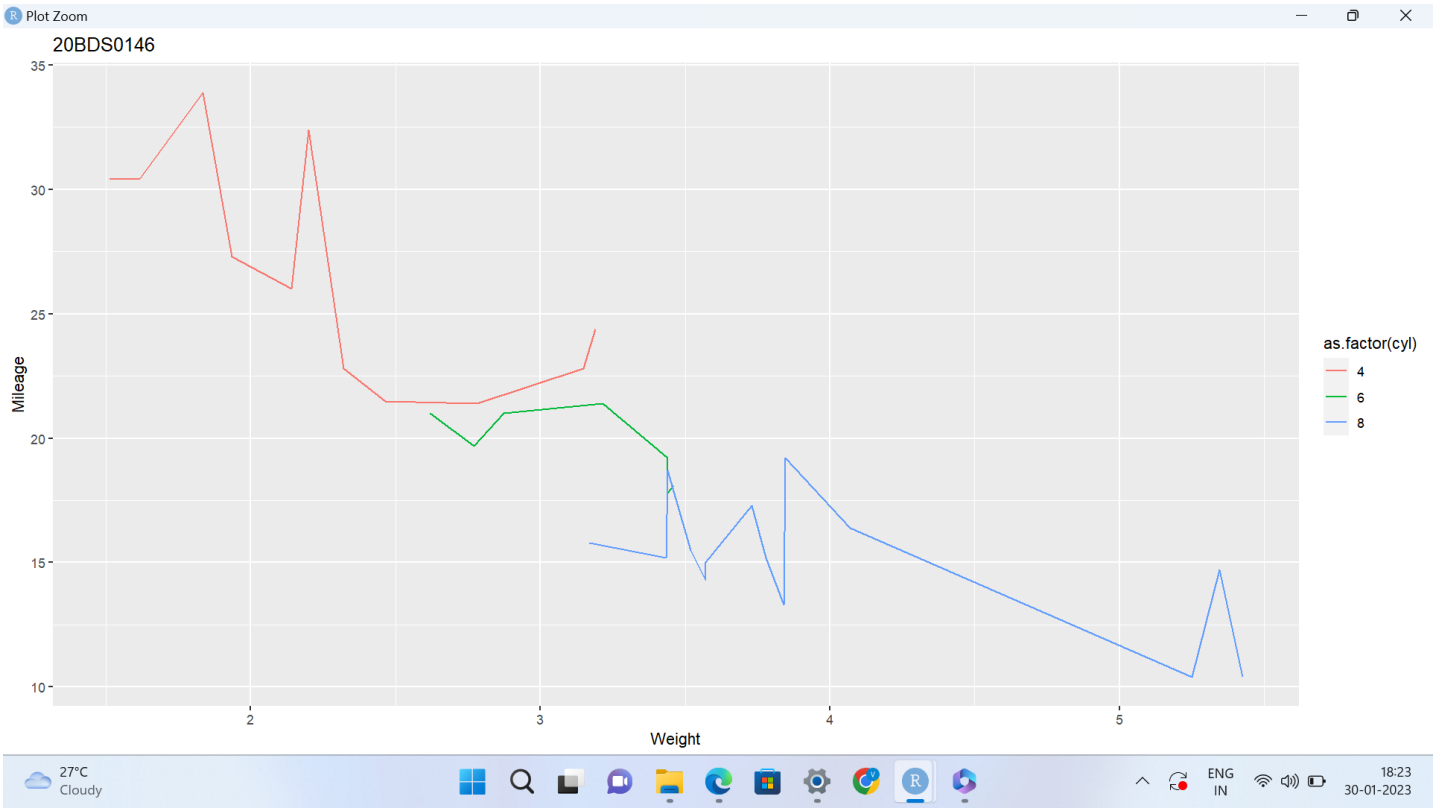


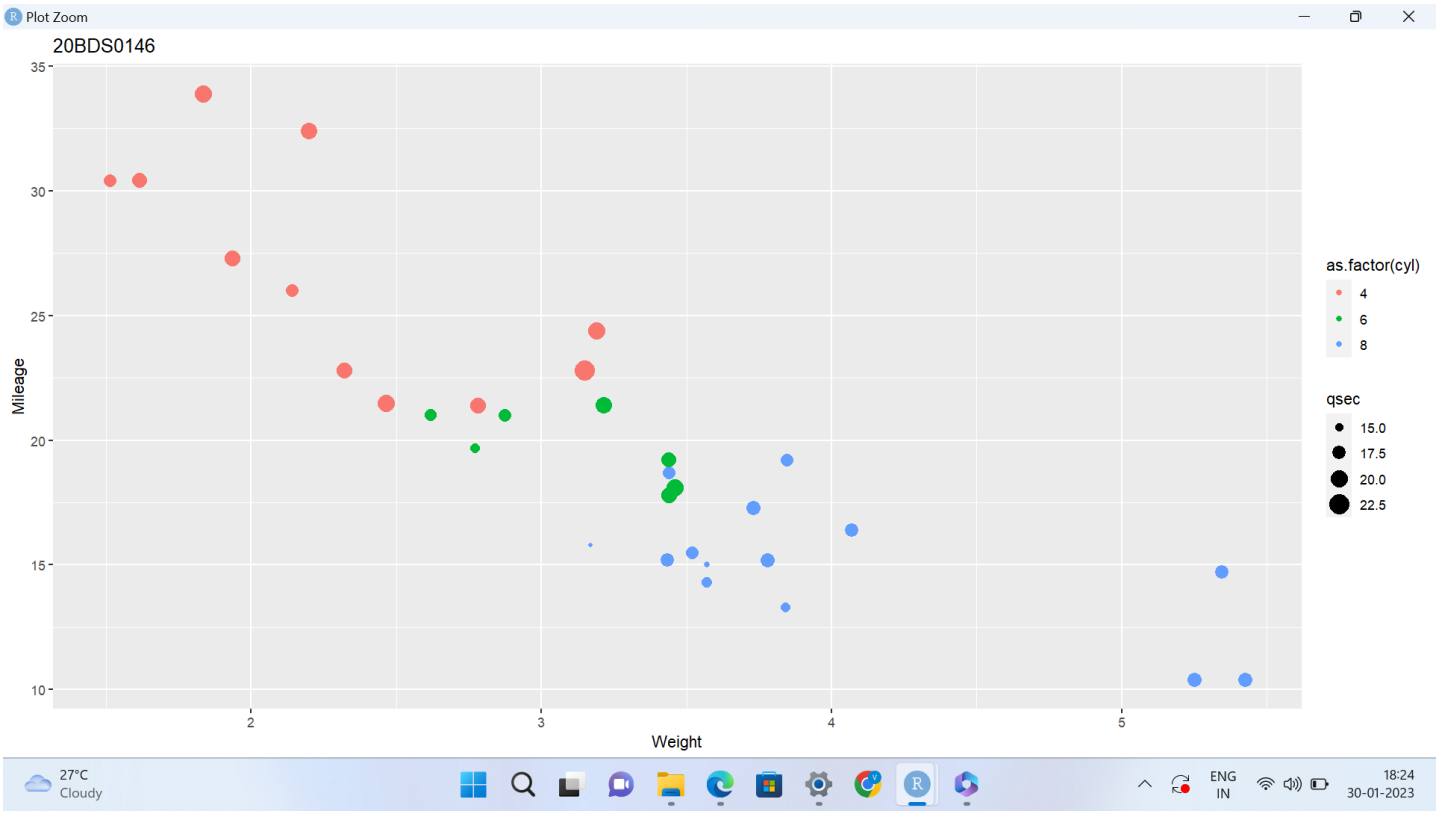

Plot Zoom
20BDS0146
mtcars$mpg
mtcars$wt
10
15
20
25
30
2
3
4
5
28°C
Cloudy
ENG
IN
18:11
30-01-2023

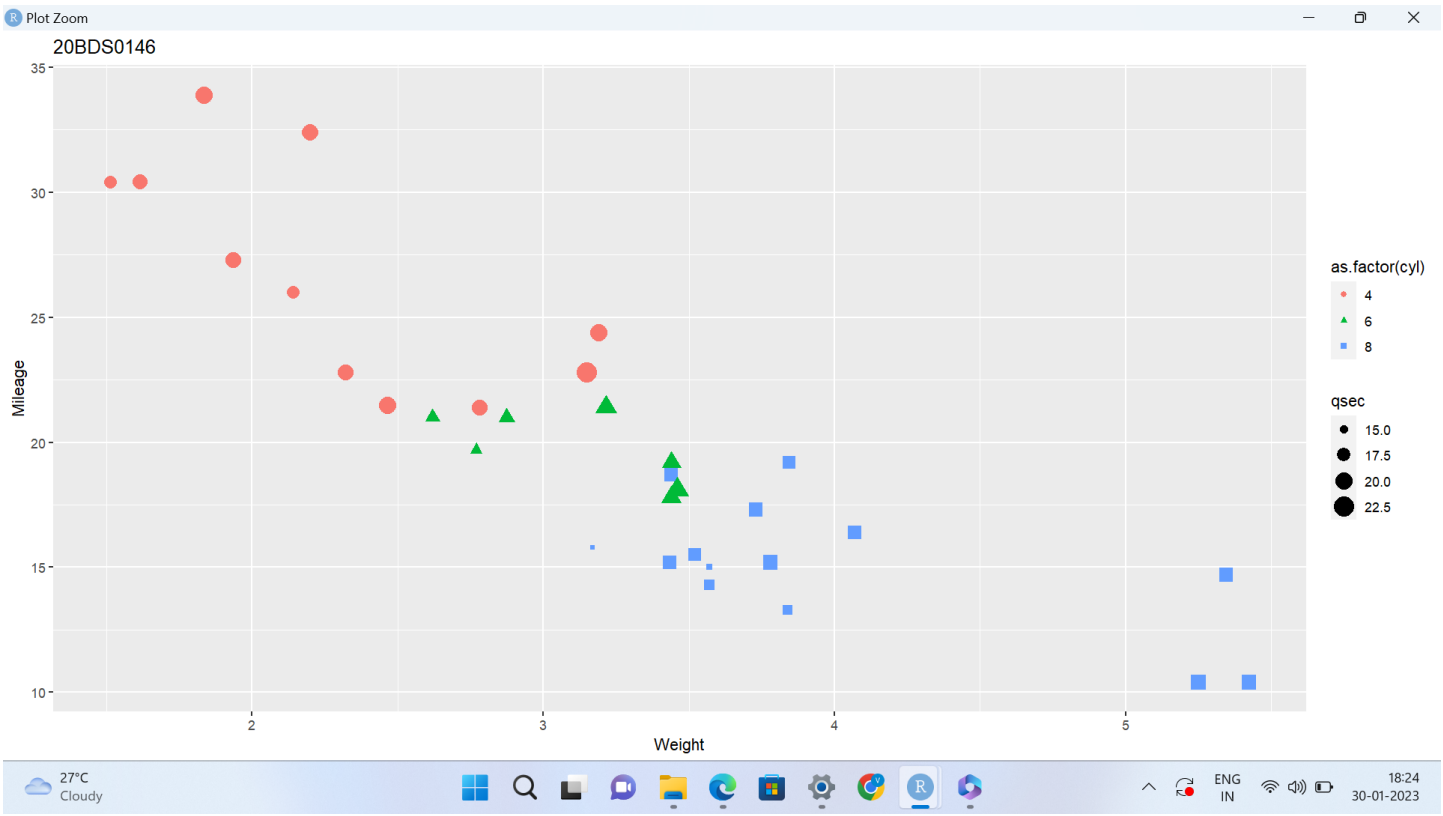

Plot Zoom
20BDS0146
mpg
wt
10
15
20
25
30
35
2
3
4
5
27°C
Cloudy
ENG
IN
18:16
30-01-2023


Plot Zoom
20BDS0146
Mileage
Weight
35
30
25
20
15
10
2
3
4
5
27°C
Cloudy
ENG
IN
18:32
30-01-2023


Plot Zoom
20BDS0146
Mileage
Weight
as.factor(gear)
3
4
5
35
30
25
20
15
10
2
3
4
5
27°C
Cloudy
ENG
IN
18:29
30-01-2023


Plot Zoom
20BDS0146
35
30
25
20
15
10
Mileage
2
3
4
5
Weight
as.factor(cyl)
4
6
8
27°C
Cloudy
ENG
IN
18:23
30-01-2023


Plot Zoom
20BDS0146
Mileage
Weight
as.factor(cyl)
4
6
8
qsec
15.0
17.5
20.0
22.5
27°C
Cloudy
ENG
IN
18:24
30-01-2023


Plot Zoom
20BDS0146
Mileage
Weight
as.factor(cyl)
4
6
8
qsec
15.0
17.5
20.0
22.5
27°C
Cloudy
ENG
IN
18:24
30-01-2023

RESULT

We have loaded in-build dataset mtcars and visualized data using visualization library ggplot and we have visualized Weight and Mileage using various plots like scatter plots,etc with many features

Ex.No.2

TITLE OF EXPERIMENT: Statistical Analysis – Univariate, Bivariate, Multivariate – plotting and coloring for maps

1. AIM: Load the gapminder dataset and perform statistical analysis using tidyverse and dplyr libraries

CODE:

```
install.packages('gapminder')
install.packages('tidyr')
install.packages('dplyr')
install.packages('ggplot2')
library('gapminder')
View(gapminder)
library('tidyr')
library('dplyr')
library('ggplot2')
#extract continent asia
gapminder%>%filter(continent=='Asia')
#extract year 1957
gapminder%>%filter(year==1957)
#extract year as 2002 and country china
```

```
gapminder%>%filter(year==2002 & country=="China")

#sort lifeExp in desc order

gapminder%>%arrange(desc(lifeExp))

#year 1957 and pop in desc order

gapminder%>%filter(year==1957)%>%arrange(desc(pop))

#lifeExp in months

gapminder%>%mutate(lifeExp*12)

gapminder%>%mutate(lifeExpInMonths=lifeExp*12)

#gapminder_1952

gapminder_1952<-gapminder%>%filter(year==1952)

View(gapminder_1952)

#VISUALIZE scatter plot for pop and gdpPercap for gapminder_1952

ggplot(data=gapminder_1952,aes(x=pop,y=gdpPercap,color=continent))
+geom_point()+scale_x_log10()+scale_y_log10()+ggtitle("20BDS0146"
)

#scale_x is used to expand data in x axis

#color is used to give color to continent ,its in 8 bit color range in form
of VIBGYOR

#SCATTER plot pop and lifexp group by continent based on population
size

ggplot(data=gapminder_1952,aes(x=pop,y=lifeExp,color=continent,size
=gdpPercap))+geom_point()+scale_x_log10()+ggtitle("20BDS0146")

#sub graph-splits graph by continent using facet_wrap function
```

```
ggplot(data=gapminder_1952,aes(x=pop,y=lifeExp,color=continent))+geom_point()+scale_x_log10()+facet_wrap(~continent)+ggtitle("20BDS0146")

#sub graph for entire data structure:gapminder

ggplot(data=gapminder,aes(x=pop,y=gdpPercap,color=continent))+geom_point()+scale_x_log10()+scale_y_log10()+ggtitle("20BDS0146")

ggplot(data=gapminder,aes(x=pop,y=lifeExp,color=continent,size=gdpPercap))+geom_point()+scale_x_log10()+ggtitle("20BDS0146")

ggplot(data=gapminder,aes(x=pop,y=lifeExp,color=continent))+geom_point()+scale_x_log10()+facet_wrap(~year)+ggtitle("20BDS0146")

ggplot(data=gapminder,aes(x=pop,y=lifeExp,color=continent))+geom_point()+scale_x_log10()+facet_wrap(~continent)+ggtitle("20BDS0146")

#Summarize -median lifeExp,DS: GAPMINDER

gapminder%>%summarize(medianLIfeExp=median(lifeExp))

#1957, median-lifeExp,max-gdpPercap

gapminder%>%filter(year==1957)%>%summarize(medianLIfeEXp=median(lifeExp),maxgdpPercap=max(gdpPercap))

#group by year,median lifeExp

#store in object by_year

by_year<-
gapminder%>%group_by(year)%>%summarize(MedianLifeExp=median(lifeExp))

#Visualize year Vs MedianLifeExp:#by_year

ggplot(data=by_year,aes(x=year,y=MedianLifeExp))+geom_point()+expand_limits(y=0)+ggtitle("20BDS0146")

#summarize median gdpPercap
```

#by year and continent and save in by_year_continent

by_year_continent<-gapminder%>%group_by(year,continent)%>%summarize(MedianGdpPercap=median(gdpPercap))

#VIsualize year vs medianGdp

#Line plot

ggplot(data=by_year_continent,aes(x=year,y=MedianGdpPercap))+geom_line()+ggtitle("20BDS0146")

ggplot(data=by_year_continent,aes(x=year,y=MedianGdpPercap,color=continent))+geom_line()+ggtitle("20BDS0146")

OUTPUT:

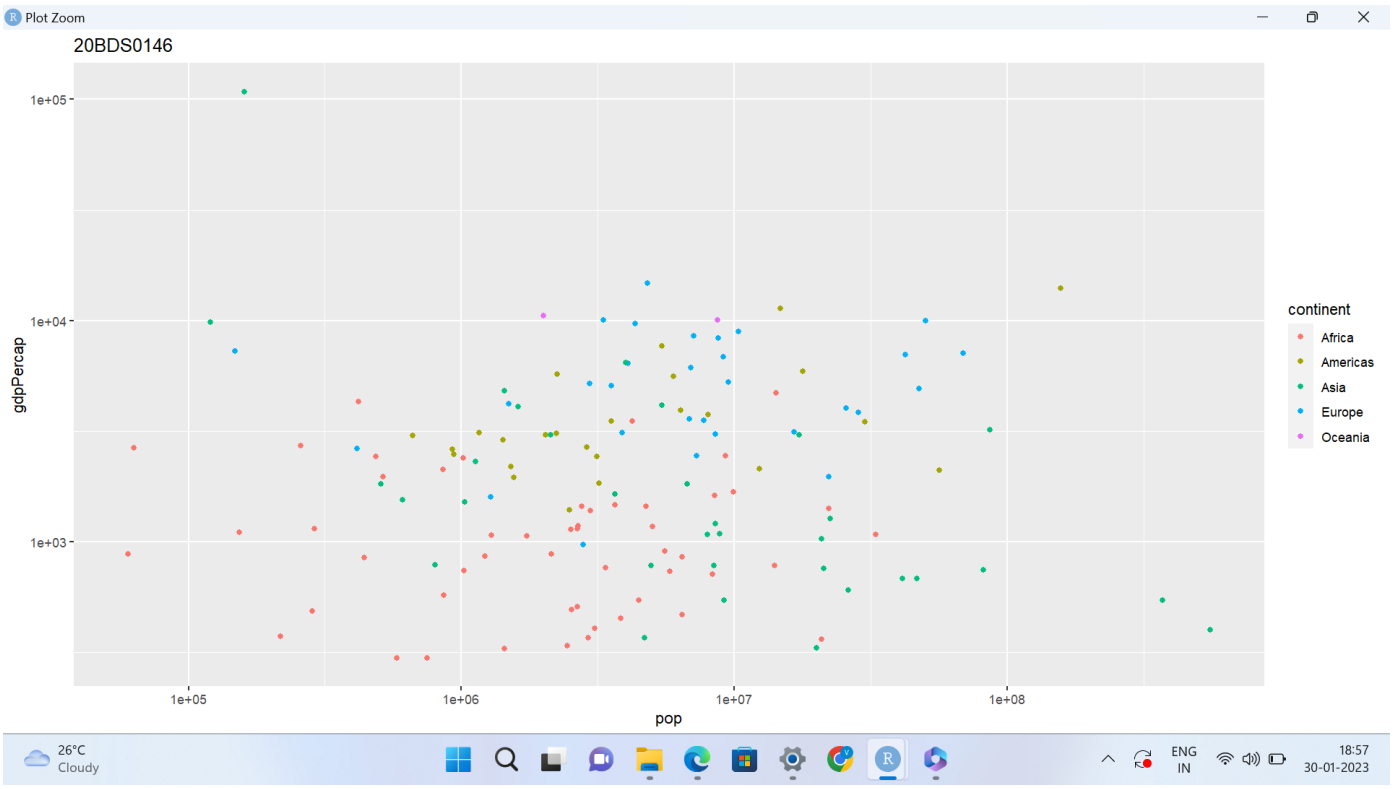
```
> gapminder%>%filter(year==2002 & country=="China")
# A tibble: 1 × 6
  country continent  year lifeExp        pop gdpPercap
  <fct>   <fct>     <int>   <dbl>      <int>     <dbl>
1 China   Asia       2002    72.0 1280400000     3119.
> #sort lifeExp in desc order
> gapminder%>%arrange(desc(lifeExp))
# A tibble: 1,704 × 6
   country          continent  year lifeExp       pop gdpPercap
   <fct>            <fct>     <int>   <dbl>     <int>     <dbl>
 1 Japan            Asia       2007    82.6 127467972    31656.
 2 Hong Kong, China Asia       2007    82.2   6980412    39725.
 3 Japan            Asia       2002    82   127065841    28605.
 4 Iceland          Europe     2007    81.8    301931    36181.
 5 Switzerland      Europe     2007    81.7   7554661    37506.
 6 Hong Kong, China Asia       2002    81.5   6762476    30209.
 7 Australia        Oceania    2007    81.2  20434176    34435.
 8 Spain            Europe     2007    80.9  40448191    28821.
 9 Sweden           Europe     2007    80.9   9031088    33860.
10 Israel           Asia       2007    80.7   6426679    25523.
# … with 1,694 more rows
# ℹ Use `print(n = ...)` to see more rows
>
```

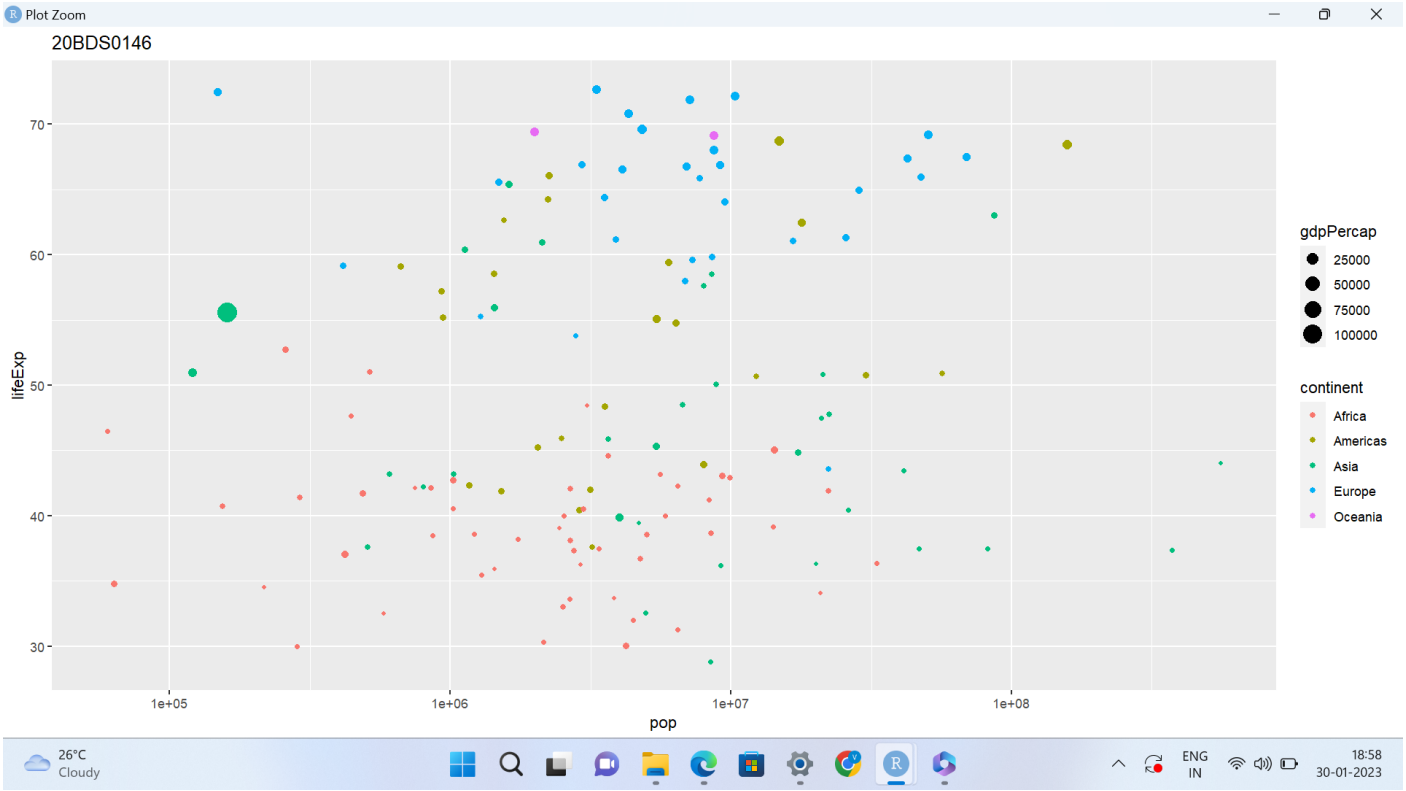
Console | Terminal | Background Jobs

R 4.2.2 · ~/

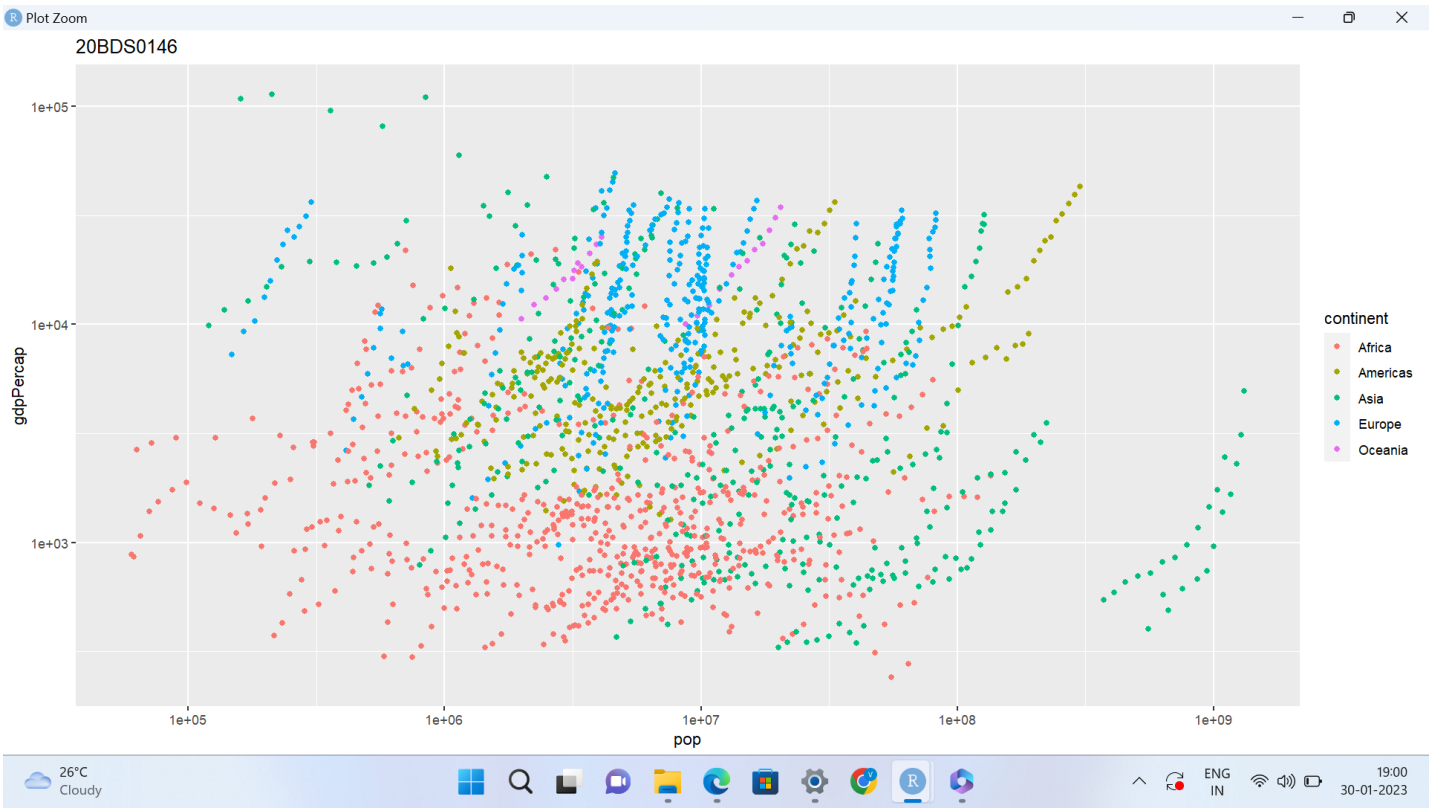
```
> gapminder%>%filter(year==1957)%>%arrange(desc(pop))
# A tibble: 142 × 6
   country         continent  year lifeExp       pop gdpPercap
   <fct>           <fct>     <int>   <dbl>     <int>     <dbl>
 1 China           Asia       1957    50.5 637408000      576.
 2 India           Asia       1957    40.2 409000000      590.
 3 United States   Americas   1957    69.5 171984000    14847.
 4 Japan           Asia       1957    65.5  91563009     4318.
 5 Indonesia       Asia       1957    39.9  90124000      859.
 6 Germany         Europe     1957    69.1  71019069    10188.
 7 Brazil          Americas   1957    53.3  65551171     2487.
 8 United Kingdom  Europe     1957    70.4  51430000    11283.
 9 Bangladesh      Asia       1957    39.3  51365468      662.
10 Italy           Europe     1957    67.8  49182000     6249.
# … with 132 more rows
# ℹ Use `print(n = ...)` to see more rows
```

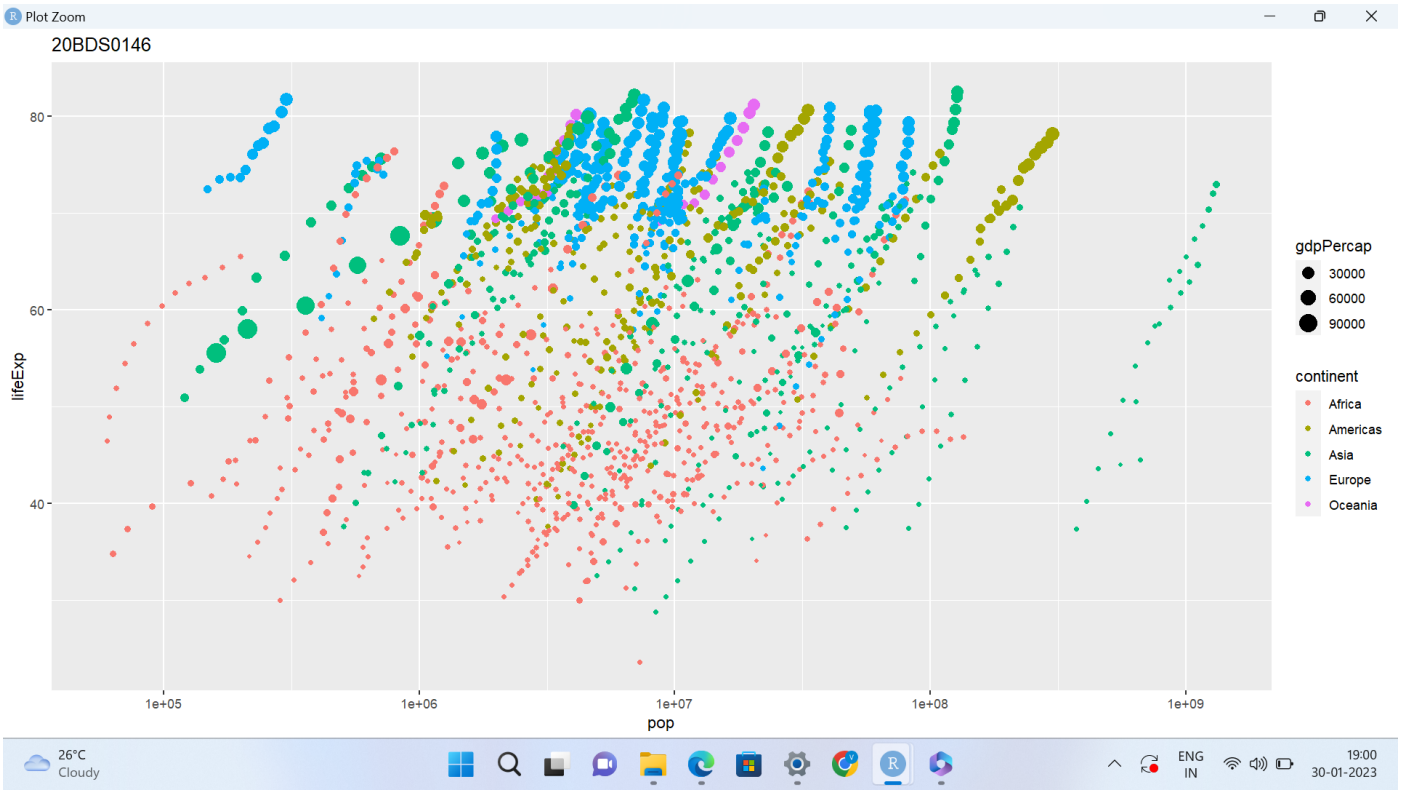
```
> gapminder%>%mutate(lifeExpInMonths=lifeExp*12)
# A tibble: 1,704 × 7
   country     continent  year lifeExp      pop gdpPercap lifeExpInMonths
   <fct>       <fct>     <int>   <dbl>    <int>     <dbl>           <dbl>
 1 Afghanistan Asia       1952    28.8  8425333      779.            346.
 2 Afghanistan Asia       1957    30.3  9240934      821.            364.
 3 Afghanistan Asia       1962    32.0 10267083      853.            384.
 4 Afghanistan Asia       1967    34.0 11537966      836.            408.
 5 Afghanistan Asia       1972    36.1 13079460      740.            433.
 6 Afghanistan Asia       1977    38.4 14880372      786.            461.
 7 Afghanistan Asia       1982    39.9 12881816      978.            478.
 8 Afghanistan Asia       1987    40.8 13867957      852.            490.
 9 Afghanistan Asia       1992    41.7 16317921      649.            500.
10 Afghanistan Asia       1997    41.8 22227415      635.            501.
# … with 1,694 more rows
# ℹ Use `print(n = ...)` to see more rows
```

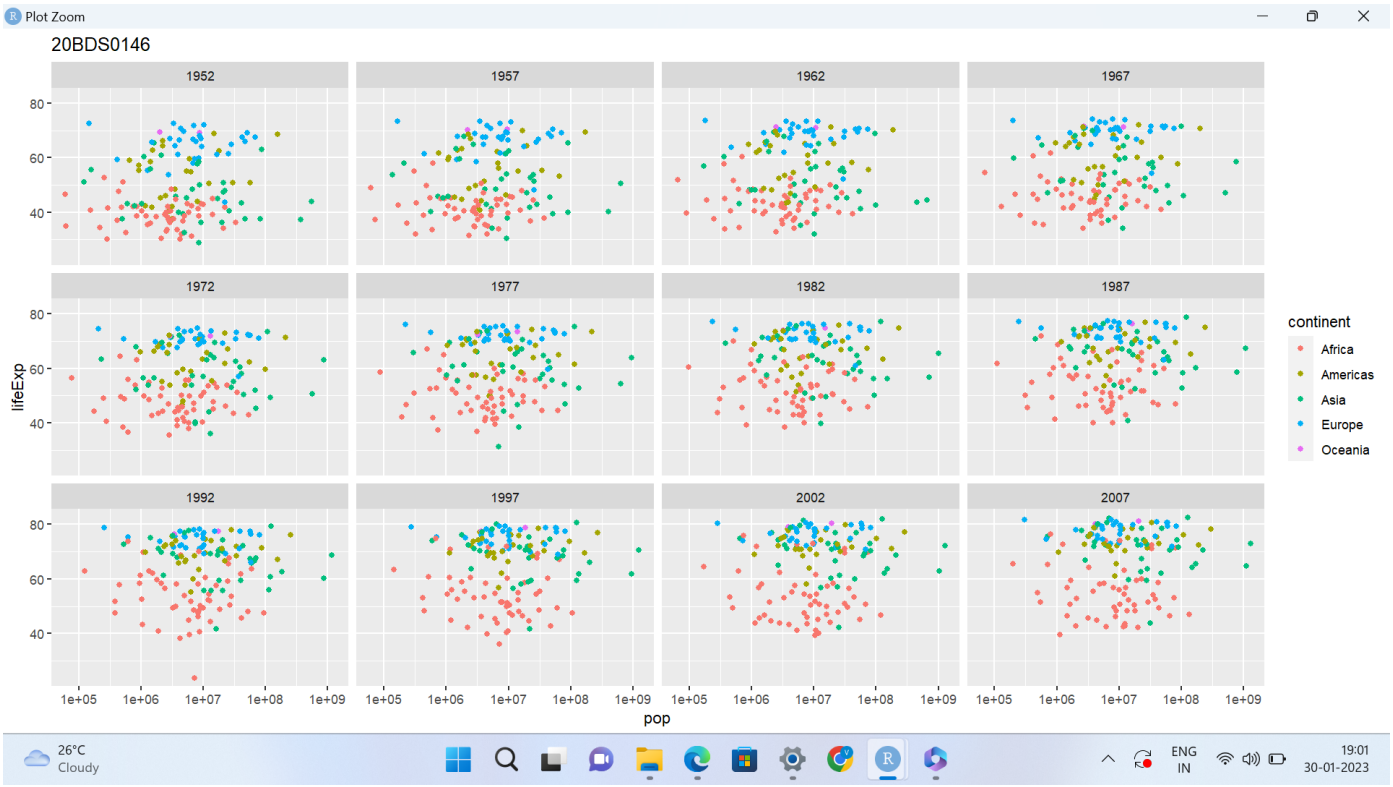

Plot Zoom
20BDS0146
gdpPercap
1e+05
1e+04
1e+03
1e+05
1e+06
1e+07
1e+08
pop
continent
Africa
Americas
Asia
Europe
Oceania
26°C
Cloudy
ENG
IN
18:57
30-01-2023

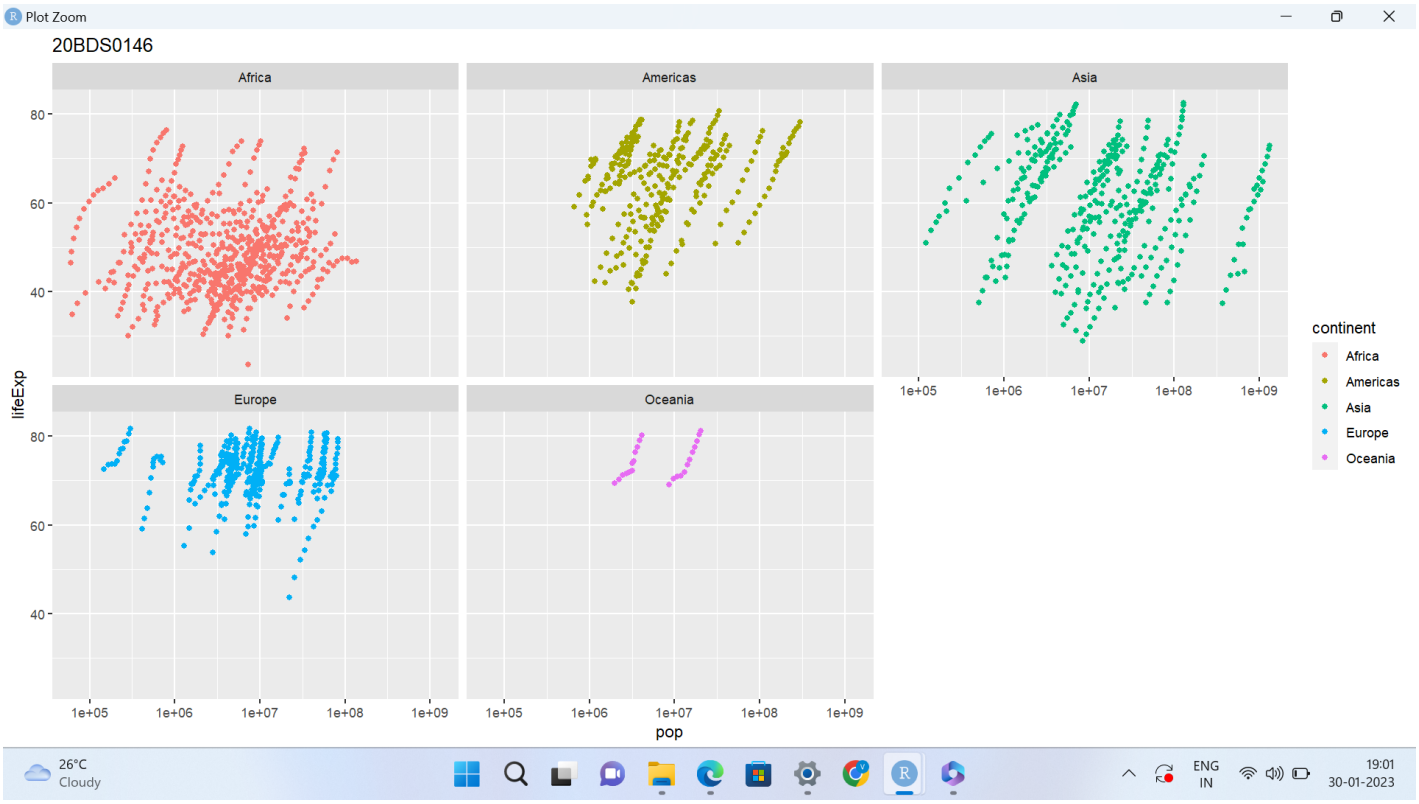

Plot Zoom
20BDS0146
lifeExp
70
60
50
40
30
1e+05
1e+06
1e+07
1e+08
pop
gdpPercap
25000
50000
75000
100000
continent
Africa
Americas
Asia
Europe
Oceania
26°C
Cloudy
ENG
IN
18:58
30-01-2023


Plot Zoom
20BDS0146
Africa
Americas
Asia
Europe
Oceania
lifeExp
pop
continent
Africa
Americas
Asia
Europe
Oceania
26°C
Cloudy
ENG
IN
18:59
30-01-2023

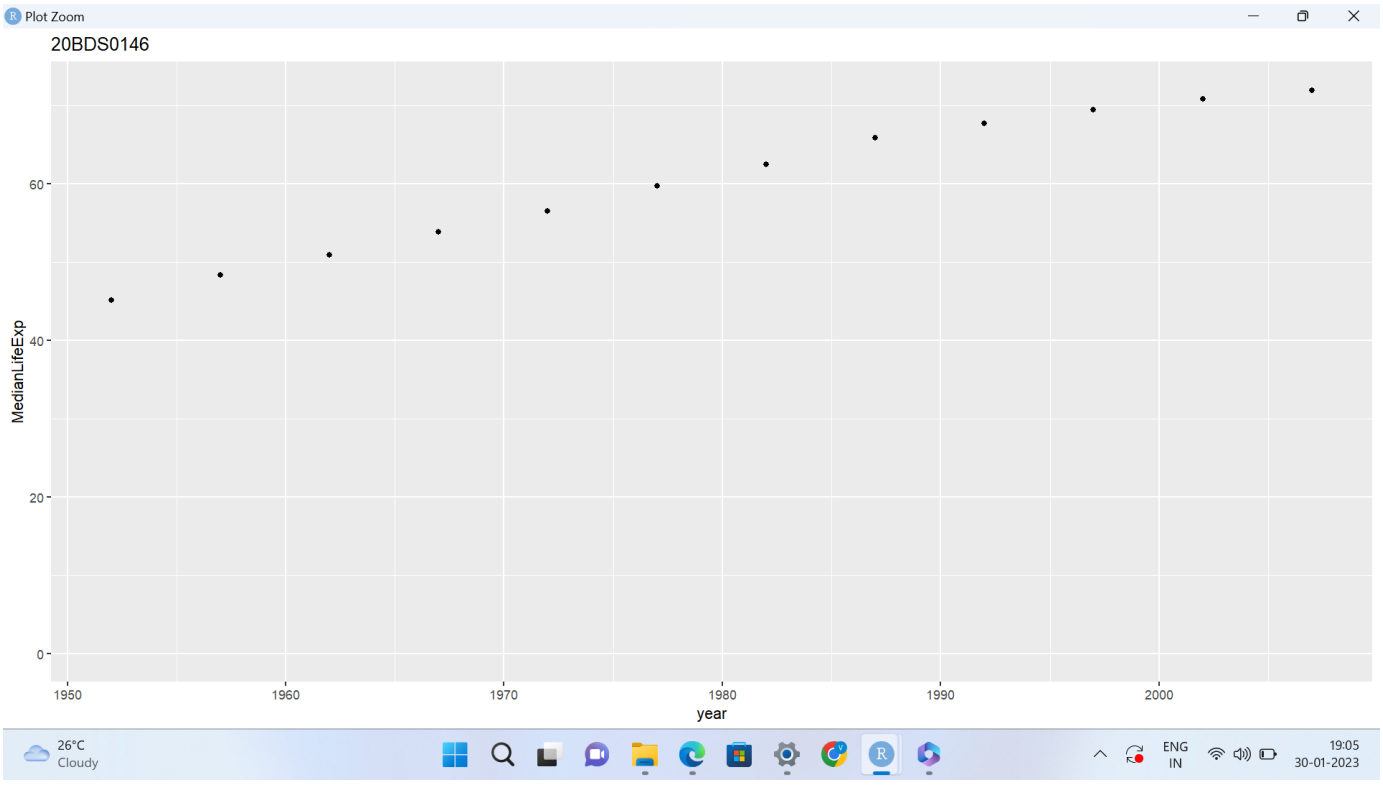

Plot Zoom
20BDS0146
gdpPercap
pop
continent
Africa
Americas
Asia
Europe
Oceania
26°C
Cloudy
ENG
IN
19:00
30-01-2023

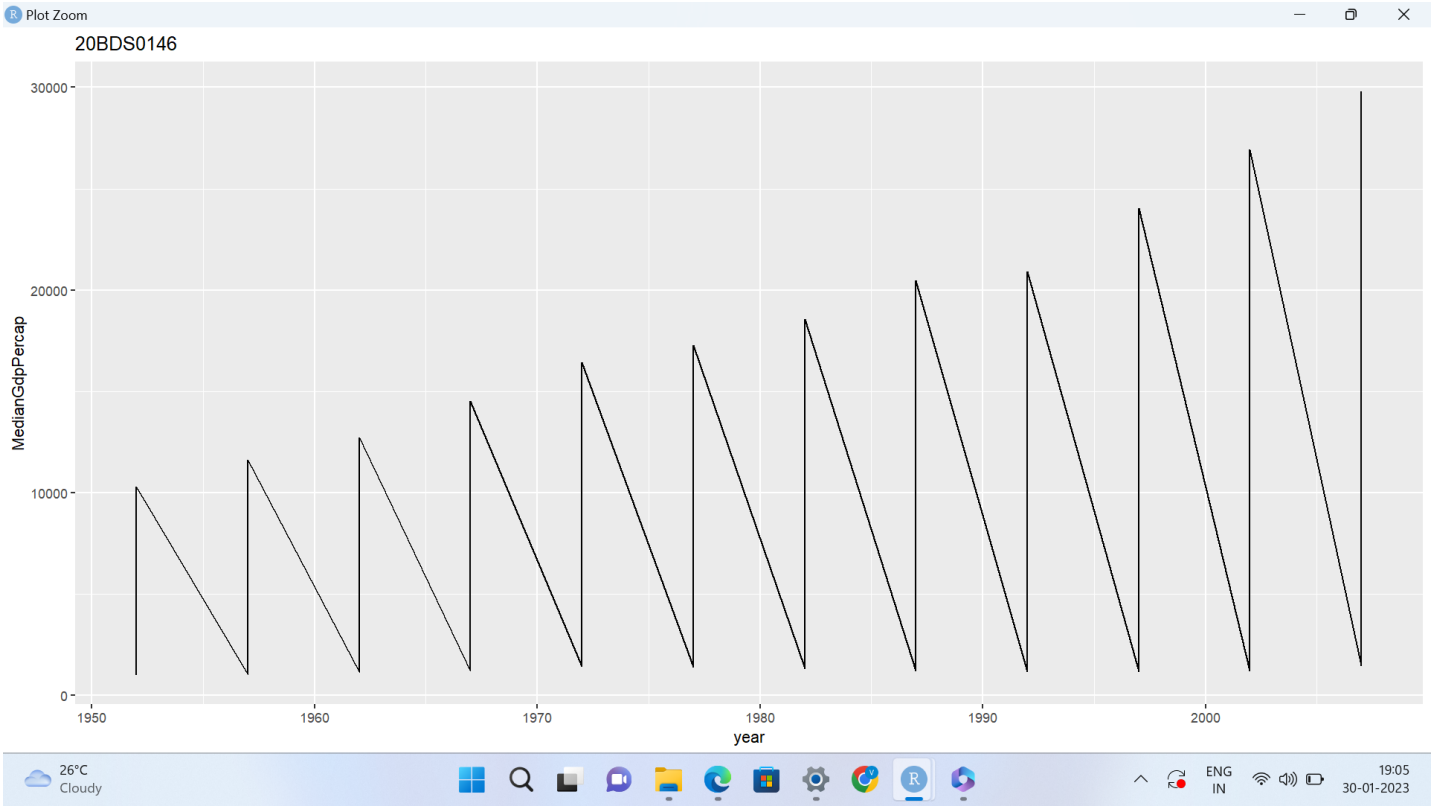

Plot Zoom
20BDS0146
lifeExp
pop
80
60
40
1e+05
1e+06
1e+07
1e+08
1e+09
gdpPercap
30000
60000
90000
continent
Africa
Americas
Asia
Europe
Oceania
26°C
Cloudy
ENG
IN
19:00
30-01-2023

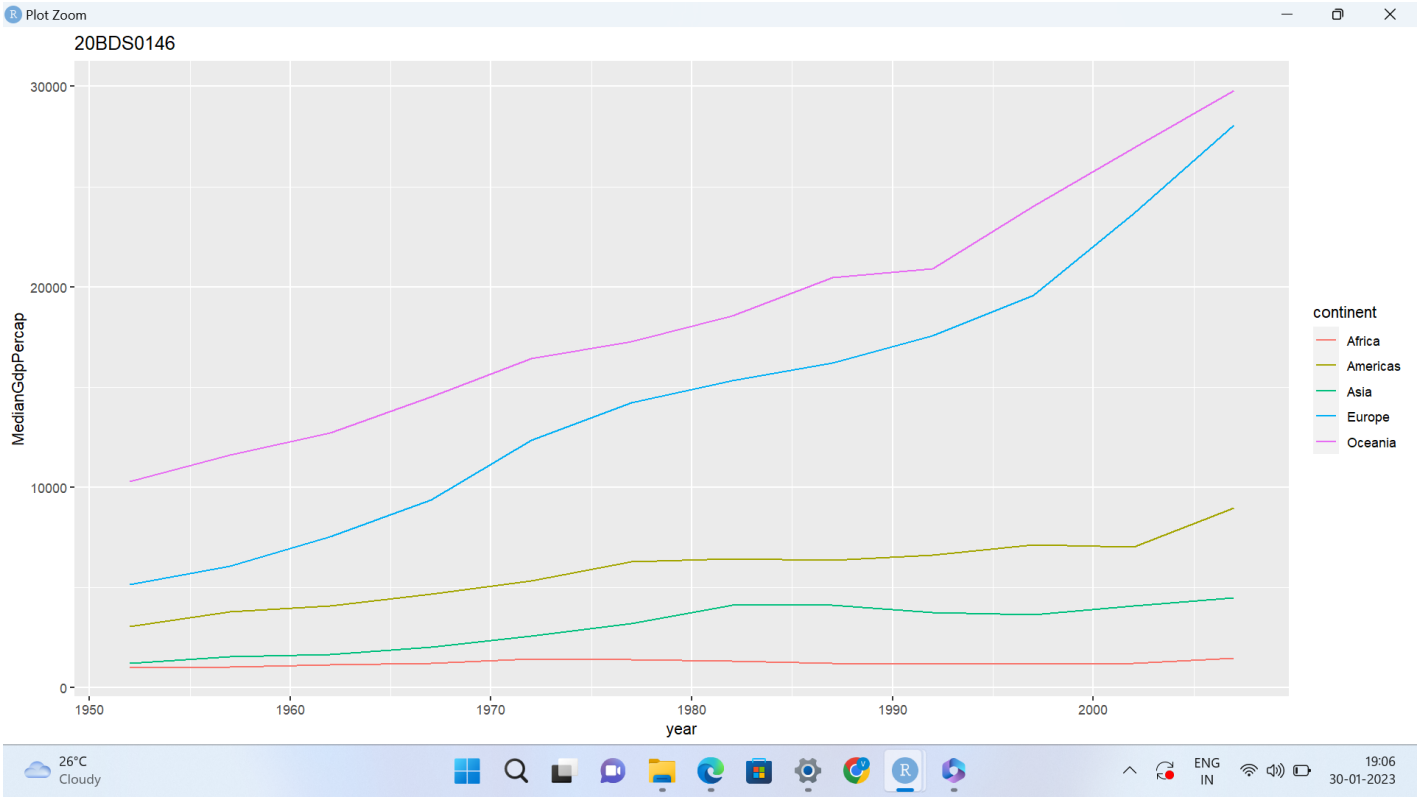

Plot Zoom
20BDS0146
1952
1957
1962
1967
1972
1977
1982
1987
1992
1997
2002
2007
lifeExp
pop
80
60
40
1e+05
1e+06
1e+07
1e+08
1e+09
continent
Africa
Americas
Asia
Europe
Oceania
26°C
Cloudy
ENG
IN
19:01
30-01-2023




```
> gapminder%>%summarize(medianLIfeExp=median(lifeExp))
# A tibble: 1 × 1
  medianLIfeExp
          <dbl>
1          60.7
> #1957, median-lifeExp,max-gdpPercap
> gapminder%>%filter(year==1957)%>%summarize(medianLIfeExp=median(lifeExp),maxgdpP
rcap=max(gdpPercap))
# A tibble: 1 × 2
  medianLIfeExp maxgdpPercap
          <dbl>        <dbl>
1          48.4      113523.
> #group by year,median lifeExp
> #store in object by_year
> by_year<-gapminder%>%group_by(year)%>%summarize(MedianLifeExp=median(lifeExp))
> #summarize median gdpPercap
> #by year and continent and save in by_year_continent
> by_year_continent<-gapminder%>%group_by(year,continent)%>%summarize(MedianGdpPer
ap=median(gdpPercap))
```


Plot Zoom
20BDS0146
MedianLifeExp
0
20
40
60
1950
1960
1970
1980
1990
2000
year
26°C
Cloudy
ENG
IN
19:05
30-01-2023


Plot Zoom
20BDS0146
MedianGdpPercap
0
10000
20000
30000
1950
1960
1970
1980
1990
2000
year
26°C
Cloudy
ENG
IN
19:05
30-01-2023



RESULT

We have loaded the gapminder dataset and perform statistical analysis using tidyverse and dplyr libraries and plotted various graphs to understand different aspects of the gapminder dataset.

2. AIM: Using RClolorBrewer visualize mpg data.

CODE:

install.packages(“gridExtra”)

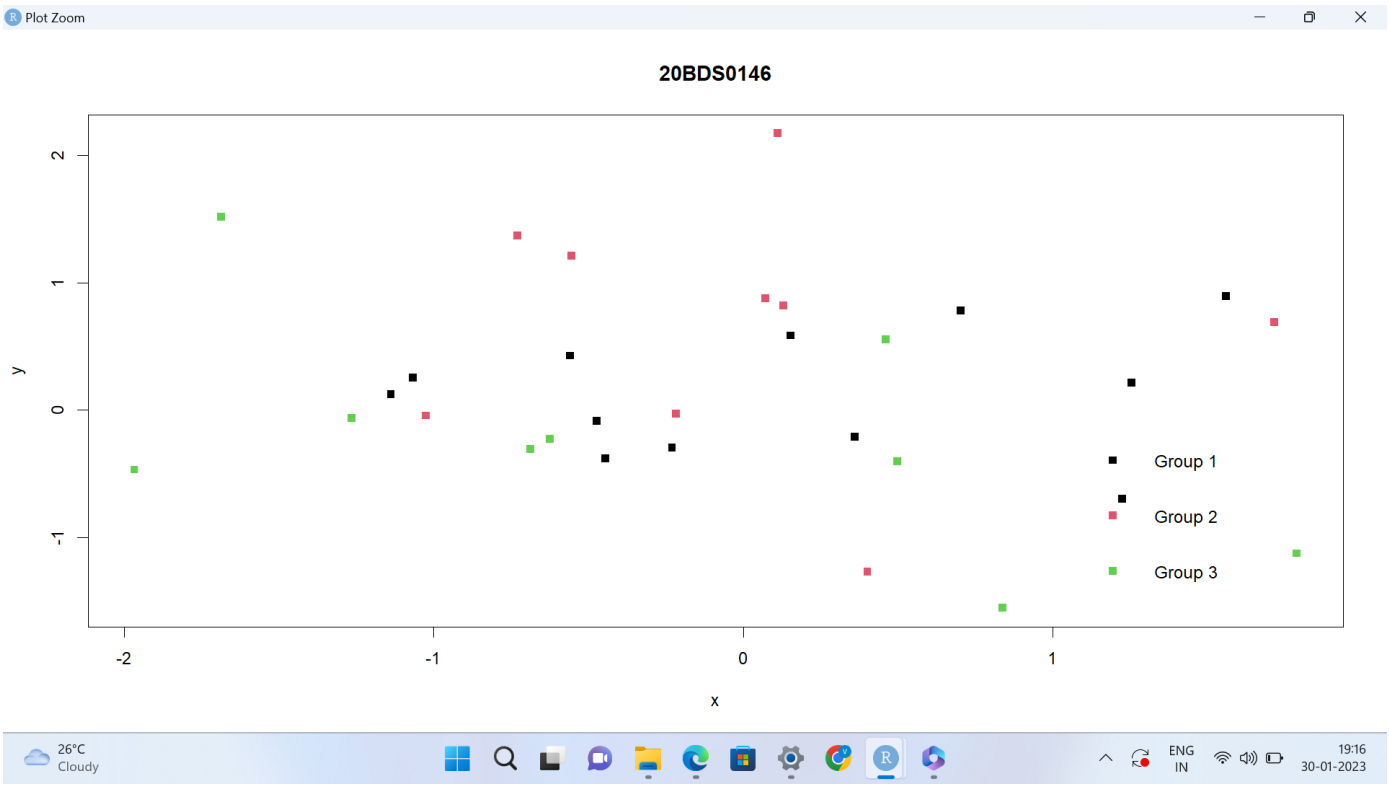
set.seed(123)

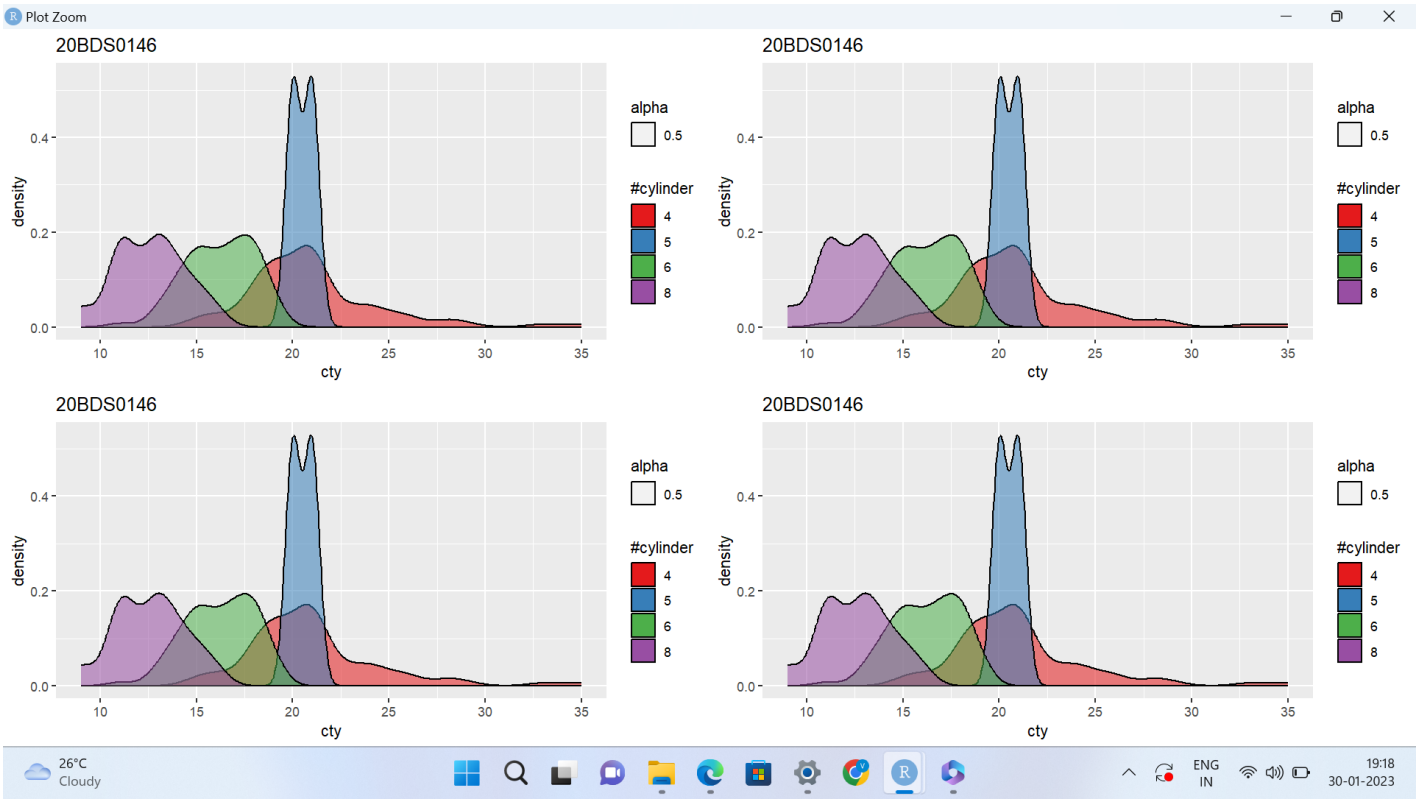
//to set values in this range

```
x <- rnorm(30)
y <- rnorm(30)
//rnorm to normalize 30 values in range x--1 to 1
//rnorm to normalize 30 values in range y--1 to 2
//rep(1:3) means reptitive 1 2 3
//pch 19 means shaded circle
//pch 25 means inverted triangle
//pch range[1-25]
plot(x,y,col=rep(1:3,each=3),pch=15,main="20BDS0146")
//legend box to represent details of colors used
legend("bottomright",legend=paste("Group",1:3),col=1:3,pch=15,bty="n
")
rep(1:3,each=3)
paste("Group",1:3)
library(RColorBrewer)
library(ggplot2)
#Color Visualization using RColorBrewer
View(mpg)
//to display in form of grid,all graphs in one page
//density plot using geom_density
//fill()to fill color
//alpha to give transparency in R
//gridExtra to arrange in form of grid similar to par
```

```
library(gridExtra)

//scale fill brewer used to provide similar colors to all values
differentiating with light and dark

p1=p2=p3=p4=ggplot(data=mpg,aes(x=cty))+geom_density(aes(fill=factor(cyl),alpha=0.5))+labs(title="XXXX",X="City Mileage",fill="#cylinder")+scale_fill_brewer(palette="Set1")+ggtitle("20BDS0146")

grid.arrange(p1,p2,p3,p4,nrow=2)

//total 234 cylinders

str(mpg)

factor(mpg$cyl)

//factor gives values distinct 4,5,6,8
```

OUTPUT:


Plot Zoom
20BDS0146
y
x
-2
-1
0
1
2
Group 1
Group 2
Group 3
26°C
Cloudy
ENG
IN
19:16
30-01-2023


Plot Zoom
20BDS0146
density
cty
alpha
0.5
#cylinder
4
5
6
8
26°C
Cloudy
ENG
IN
19:18
30-01-2023

```
> factor(mpg$cyl)
  [1] 4 4 4 4 6 6 6 4 4 4 4 6 6 6 6 6 6 8 8 8 8 8 8 8 8 8 8 8 8 8 8 4 4 6 6 6
 [38] 4 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8 8 6 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
 [75] 8 8 8 6 6 6 6 8 8 6 6 8 8 8 8 8 6 6 6 6 8 8 8 8 4 4 4 4 4 4 4 4 4 4 4 4
[112] 4 6 6 6 4 4 4 4 6 6 6 6 6 6 8 8 8 8 8 8 8 8 8 8 8 8 6 6 8 8 4 4 4 4 6 6 6
[149] 6 6 6 6 6 8 6 6 6 6 8 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 6 6 6 8 4 4 4 4 6 6
[186] 6 4 4 4 4 6 6 6 4 4 4 4 4 8 8 4 4 4 6 6 6 6 4 4 4 4 6 4 4 4 4 4 5 5 6 6 4
[223] 4 4 4 5 5 4 4 4 4 6 6 6
Levels: 4 5 6 8
> 
```

RESULT

We have used RColorBrewer to visualize mpg data and also we used scale_fill_brewer() to explore features of RColorBrewer.

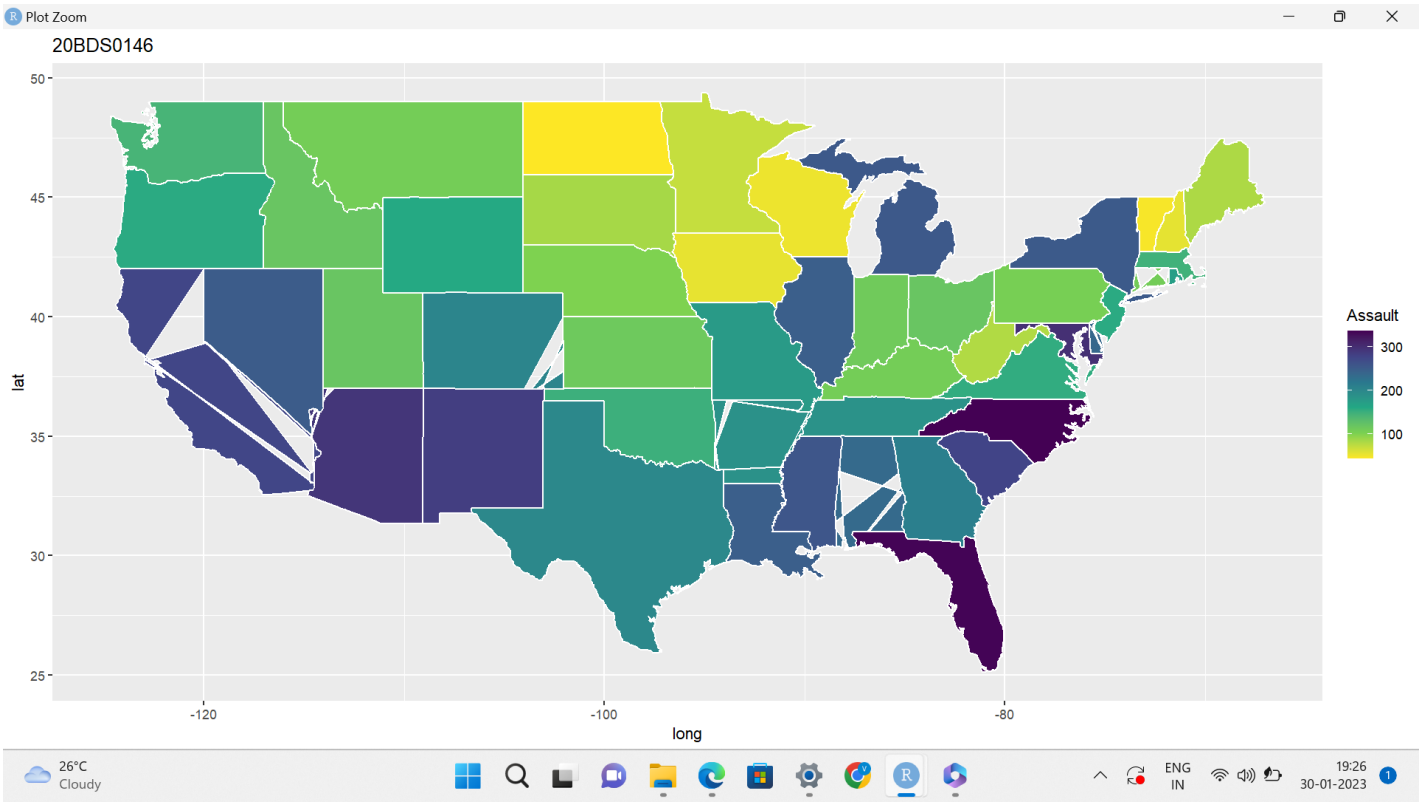
3. AIM: Load USArrests in-build dataset and correlate in the maps with anyone fields. Display the maps using colormapping.

CODE:

```
install.packages("viridisLite")
library(viridis)
 install.packages("maps")
install.packages("dplyr")
library(dplyr)
library(maps)
#Preparing dataset
View(USArrests)
arrests<-USArrests
```

```
//rownames to lower case

tolower(rownames(USArrests))

arrests$region=tolower(rownames(USArrests))

View(arrests)//new row added of region names

states_map<-map_data("state")

//all data to represent US map

View(map_data("state"))

//combine two arrests and map_data state using left join as region is common

arrest_map=merge(states_map,arrests,by="region")

View(arrest_map)

#Create the map

ggplot(data=arrest_map,aes(x=long,y=lat,group=group))+geom_polygon(aes(fill=Assault),color="white")+scale_fill_viridis_c(option="D",direction=-1)+ggtitle("20BDS0146")

//color white splits colors
```

OUTPUT:



RESULT

We have loaded USArrests in-build dataset and correlated in the maps with lat, long fields,etc. And also displayed the map using colormapping.