Set up:

1) CheckeitherJava1.8.0isalreadyinstalledonyoursystemornot,use“Javac version" to check Java version

2) IfJava isnotinstalledonyoursystemthenfirst installjavaunder"C:\JAVA"Java setup

3) Extract filesHadoop2.8.0.tar.gzorHadoop-2.8.0.zipandplaceunder"C:\Hadoop- 2.8.0" Hadoop

4) SetthepathHADOOP_HOMEEnvironment variableonwindows10(seeStep1,2, 3 and 4 below) hadoop

5) SetthepathJAVA_HOMEEnvironment variableonwindows10(seeStep1,2,3 and 4 below) java

6) Nextwesetthe HadoopbindirectorypathandJAVA bindirectorypath

Configuration:

a) FileC:/Hadoop-2.8.0/etc/hadoop/core-site.xml,pastebelowxmlparagraphandsave this file.

```
<configuration>
 <property>
  <name>fs.defaultFS</name>
```

```
    <value>hdfs://localhost:9000</value>
  </property>
</configuration>
```

b)Rename"mapred-site.xml.template"to"mapred-site.xml"andedit thisfileC:/Hadoop- 2.8.0/etc/hadoop/mapred-site.xml, paste below xml paragraph and save this file.

```
<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
</configuration>
```

c) Createfolder"data"under"C:\Hadoop-2.8.0"

1) Createfolder"datanode"under"C:\Hadoop-2.8.0\data"

2) Createfolder"namenode"under"C:\Hadoop-2.8.0\data" data

d) EditfileC:\Hadoop-2.8.0/etc/hadoop/hdfs-site.xml,pastebelowxmlparagraphandsave this file.

```
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
```

```
        <value>C:\hadoop-2.8.0\data\namenode</value>
      </property>
      <property>
        <name>dfs.datanode.data.dir</name>
        <value>C:\hadoop-2.8.0\data\datanode</value>
    </property>
</configuration>
```

e)Edit file C:/Hadoop-2.8.0/etc/hadoop/yarn-site.xml, paste below xml paragraph and save this file.

```
 <configuration>
      <property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
      </property>
      <property>
         <name>yarn.nodemanager.auxservices.mapreduce.shuffle.class</name>
         <value>org.apache.hadoop.mapred.ShuffleHandler</value>
       </property>
</configuration>
```

f)Edit file C:/Hadoop-2.8.0/etc/hadoop/hadoop-env.cmd by closing the command line"JAVA_HOME=%JAVA_HOME%"insteadofset"JAVA_HOME=C:\Java"(OnC:\java this is path to file jdk.18.0

Hadoop Configuration

7) Download file Hadoop Configuration.zip (Link: https://github.com/MuhammadBilalYar/HADOOP-INSTALLATION-ONWINDOW- 10/blob/master/Hadoop%20Configuration.zip)

8) Deletefile binonC:\Hadoop-2.8.0\bin,replacedbyfile binonfile just download(from Hadoop Configuration.zip).

9) Opencmdandtypingcommand"hdfsnamenode–format".Youwillsee hdfsnamenode–format

Testing

10) Opencmdandchangedirectoryto "C:\Hadoop-2.8.0\sbin"andtype"startall.cmd"to start apache.

Makesuretheseappsarerunning.a)Namenodeb)Hadoopdatanodec)YARN Resource Manager d)YARN Node Manager hadoop nodes

1. Open:http://localhost:8088
2. Open:http://localhost:50070

Prepare:

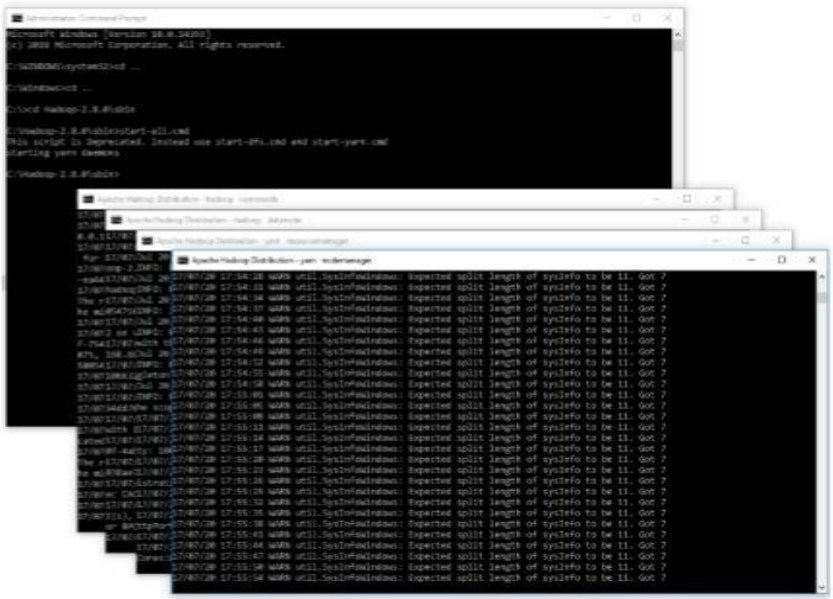
1. DownloadMapReduceClient.jar (Link:https://github.com/MuhammadBilalYar/HADOOPINSTALLATION-ON- WINDOW-10/blob/master/MapReduceClient.jar)
2. DownloadInput_file.txt (Link:https://github.com/MuhammadBilalYar/HADOOPINSTALLATION-ON- WINDOW-10/blob/master/input_file.txt)

   Placebothfiles in"C:/"

HadoopOperation:

1. OpencmdinAdministrativemodeandmoveto"C:/Hadoop-2.8.0/sbin"andstartcluster

Start-all.cmd



1. Create an input directory in HDFS.

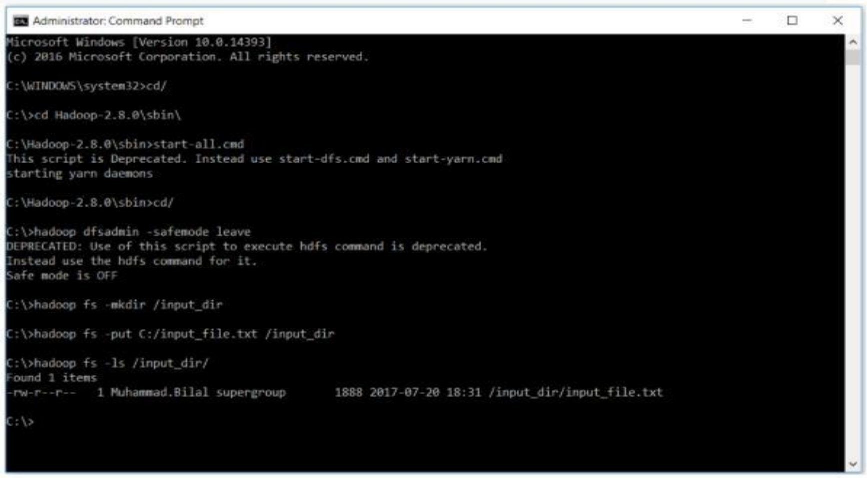
Hadoop fs-mkdir/input_dir

2. Copy the input text file named input_file.txt in the input directory (input_dir) of HDFS.

hadoop fs-putC:/input_file.txt/input_dir

3. Verify input_file.txt available in HDFS input directory(input_dir).

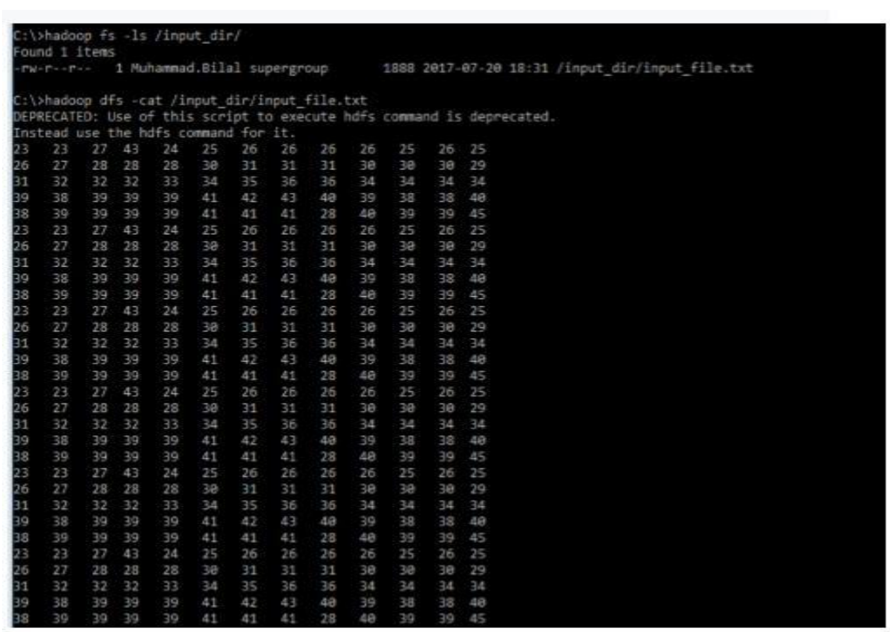
hadoop fs-ls /input_dir/



```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd/

C:\>cd Hadoop-2.8.0\sbin\

C:\Hadoop-2.8.0\sbin>start-all.cmd
This script is Deprecated. Instead use start-dfs.cmd and start-yarn.cmd
starting yarn daemons

C:\Hadoop-2.8.0\sbin>cd/

C:\>hadoop dfsadmin -safemode leave
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.
Safe mode is OFF

C:\>hadoop fs -mkdir /input_dir

C:\>hadoop fs -put C:/input_file.txt /input_dir

C:\>hadoop fs -ls /input_dir/
Found 1 items
-rw-r--r--   1 Muhammad.Bilal supergroup       1888 2017-07-20 18:31 /input_dir/input_file.txt

C:\>
```

4. Verify content of the copied file.

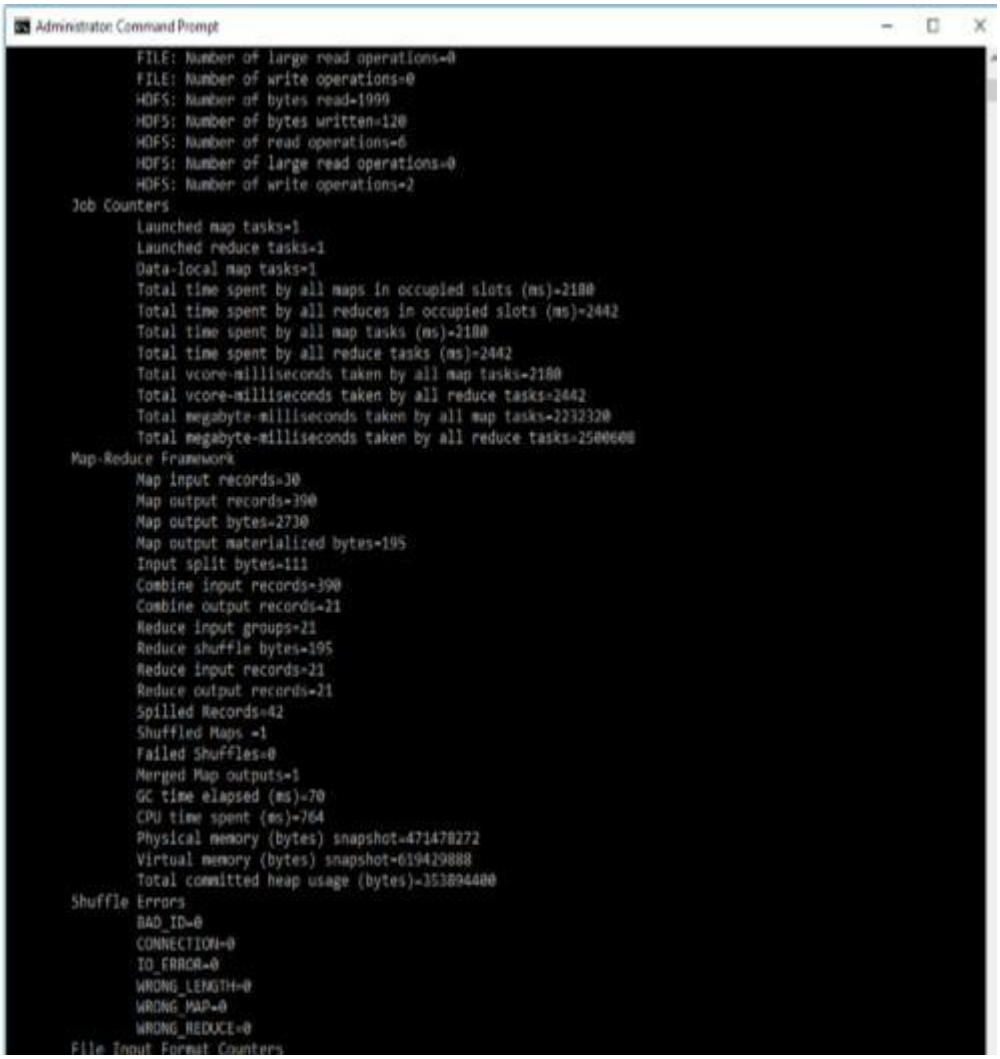
Hadoop dfs -cat /input_dir/input_file.txt



```
C:\>hadoop fs -ls /input_dir/
Found 1 items
-rw-r--r--   1 Muhammad.Bilal supergroup       1888 2017-07-20 18:31 /input_dir/input_file.txt

C:\>hadoop dfs -cat /input_dir/input_file.txt
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
```

5. Run Map Reduce Client.jar and also provide input and out directories.

hadoop jarC:/MapReduceClient.jar wordcount /input_dir/output_dir



6. Verify content for generated output file.

hadoop dfs -cat /output_dir/*

Some Other useful commands

8) To leave Safe mode

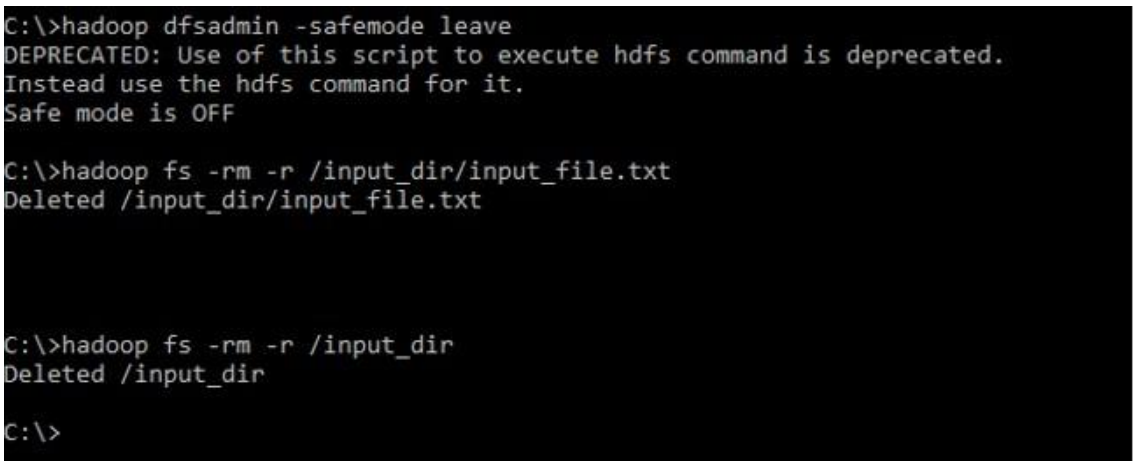
hadoop dfs admin–safemode leave

9) To delete file from HDFS directory

hadoop fs –rm -r/iutput_dir/input_file.txt

10) To delete directory from HDFS directory

hadoop fs –rm -r/iutput_dir



Program:

AverageMapper.java

```
importorg.apache.hadoop.io.*;
importorg.apache.hadoop.mapre
duce.*; import
java.io.IOException;
publicclassAverageMapperextendsMapper<LongWritable, Text,Text,IntWritable>
 {
 publicstaticfinalintMISSING=9999;
 publicvoidmap(LongWritablekey,Textvalue,Contextcontext)throws
IOException,InterruptedException
        {
                String line =
                value.toString();
                Stringyear=line.substring(1
                5,19); int temperature;
                  if(line.charAt(87)=='+')
                      temperature=Integer.parseInt(line.substring(88,92));
                  else
                      temperature=Integer.parseInt(line.substring(87,92));
                Stringquality= line.substring(92,93);
                if(temperature!=MISSING&&quality.matches("[01459]"))
```

```
                context.write(new Text(year),new IntWritable(temperature));
        }

        }
```

AverageReducer.java

```
importorg.apache.hadoop.mapre
duce.*; import
java.io.IOException;
publicclassAverageReducerextendsReducer<Text,IntWritable,Text,IntWritable>
 {
 publicvoidreduce(Textkey,Iterable<IntWritable>values,Contextcontext)throws
IOException,
 InterruptedException
          {
                intmax_temp
                =0; int count
                = 0;
                for(IntWritablevalue:values)
                   {
                          max_temp+=value.get
                          (); count+=1;
                   }
          context.write(key,newIntWritable(max_temp/count));          }
}
```

AverageDriver.java

```
importorg.apache.hadoo
p.io.*;
importorg.apache.hadoo
p.fs.*;
importorg.apache.hadoop.mapreduce.*;
importorg.apache.hadoop.mapreduce.lib.input.FileInputFormat;
importorg.apache.hadoop.mapreduce.lib.output.FileOutputF
ormat; public class AverageDriver
 {

        publicstaticvoidmain(String[]args)throwsException
         {
                 if(args.length!=2)
                 {
                       System.err.println("PleaseEntertheinputandoutputparameter
                       s"); System.exit(-1);
                 }
                Job job = new Job();
                job.setJarByClass(AverageDriver.class);
                job.setJobName("Max temperature");
                FileInputFormat.addInputPath(job,new
                Path(args[0]));
                FileOutputFormat.setOutputPath(job,newPath(args[1
```

```
        ])); job.setMapperClass(AverageMapper.class);
        job.setReducerClass(AverageReducer.class);
      job.setOutputKeyClass(Text.class);
      job.setOutputValueClass(IntWritable.class);
      System.exit(job.waitForCompletion(true)?0:1);
    }
}
```

OUTPUT:

Year	Average Temperature
1950	23
1951	18
1952	25

PROGRAM:

SIMPLE LINEAR REGRESSION

```
dataset=read.csv("data-marketing-budget-

12mo.csv",header=T, colClasses = c("numeric",

"numeric", "numeric"))head(dataset,5)

#/////SimpleRegression/////

simple.fit=lm(Sales~Spend,data=dataset)

summary(simple.fit)
```

OUTPUT:

```
Call:
lm(formula = Sales ~ Spend, data = dataset)

Residuals:
   Min     1Q Median     3Q    Max
 -3385  -2097    258   1726   3034

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1383.4714  1255.2404   1.102    0.296
Spend         10.6222     0.1625  65.378 1.71e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2313 on 10 degrees of freedom
Multiple R-squared:  0.9977, Adjusted R-squared:  0.9974
F-statistic:  4274 on 1 and 10 DF,  p-value: 1.707e-14
```

PROGRAM:

Logistic Regression

```
#selects some column from

mtcars input<-

mtcars[,c("am","cyl","hp","wt")]

print(head(input))

input<- mtcars[,c("am","cyl","hp","wt")]

am.data=glm(formula=am~cyl+hp+wt,data=input,family=binomial)

print(summary(am.data))
```

OUTPUT:

```
> source("~/Logistic Regression.r", echo=TRUE)

> #selects some column from mtcars
> input<- mtcars [,c("am","cyl","hp","wt")]

> print(head(input))
                  am cyl  hp    wt
Mazda RX4          1   6 110 2.620
Mazda RX4 Wag      1   6 110 2.875
Datsun 710         1   4  93 2.320
Hornet 4 Drive     0   6 110 3.215
Hornet Sportabout  0   8 175 3.440
Valiant            0   6 105 3.460

> input<- mtcars [,c("am","cyl","hp","wt")]

> am.data =glm(formula = am ~ cyl+hp+wt,data = input,family = binomial)

> print(summary(am.data))

Call:
glm(formula = am ~ cyl + hp + wt, family = binomial, data = input)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) 19.70288    8.11637   2.428   0.0152 *
cyl          0.48760    1.07162   0.455   0.6491
hp           0.03259    0.01886   1.728   0.0840 .
wt          -9.14947    4.15332  -2.203   0.0276 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 43.2297  on 31  degrees of freedom
Residual deviance:  9.8415  on 28  degrees of freedom
AIC: 17.841

Number of Fisher Scoring iterations: 8
```

PROGRAM

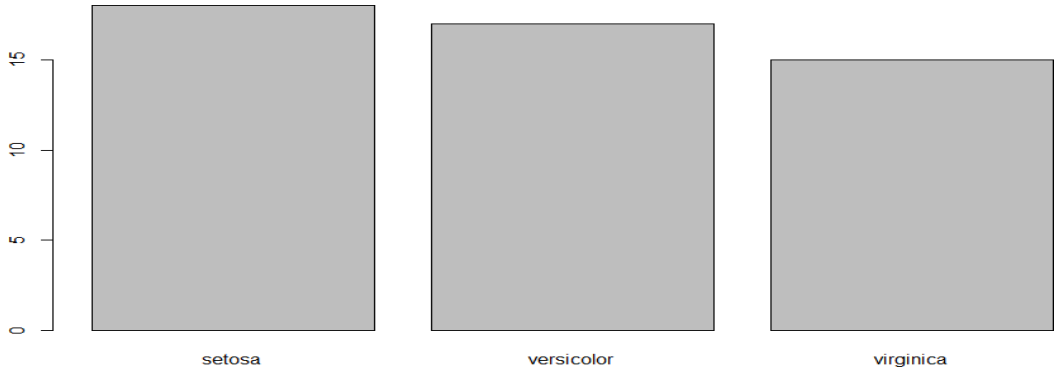
```
library(e1071)

plot(iris)

iris

plot(iris$Sepal.Length,iris$Sepal.width,col=iris$S

pecies) plot(iris$Petal.Length, iris$Petal.width,

col=iris$Species) s<-sample(150,100)

col<-c("Petal.Length","Petal.Width","Species")

iris_train<- iris[s,col]

iris_test<-iris[-s,col]

svmfit<- svm(Species~., data=iris_train, kernel= "linear", cost=.1,scale=FALSE)

print(svmfit)

plot(svmfit,iris_train[,col])

tuned<-tune(svm,Species~.,data=iris_train,kernel="linear",ranges=
list(cost=c(0.001,0.01,.1,.1,10,100)))

summary(tuned)

p<-predict(svmfit,iris_test[,col],type="class")

plot(p)
```

table(p,iris_test[,3])

mean(p==iris_test[,3])

OUTPUT:

```
> source("~/SVM Classification Techniques.R", echo=TRUE)

> library(e1071)

> plot(iris)

> iris
    Sepal.Length Sepal.Width Petal.Length Petal.Width     Species
1            5.1         3.5          1.4         0.2      setosa
2            4.9         3.0          1.4         0.2      setosa
3            4.7         3.2          1.3         0.2      setosa
4            4.6         3.1          1.5         0.2      setosa
5            5.0         3.6          1.4         0.2      setosa
6            5.4         3.9          1.7         0.4      setosa
7            4.6         3.4          1.4         0.3      setosa
8            5.0         3.4          1.5         0.2      setosa
9            4.4         2.9          1.4         0.2      setosa
10           4.9         3.1          1.5         0.1      setosa
```

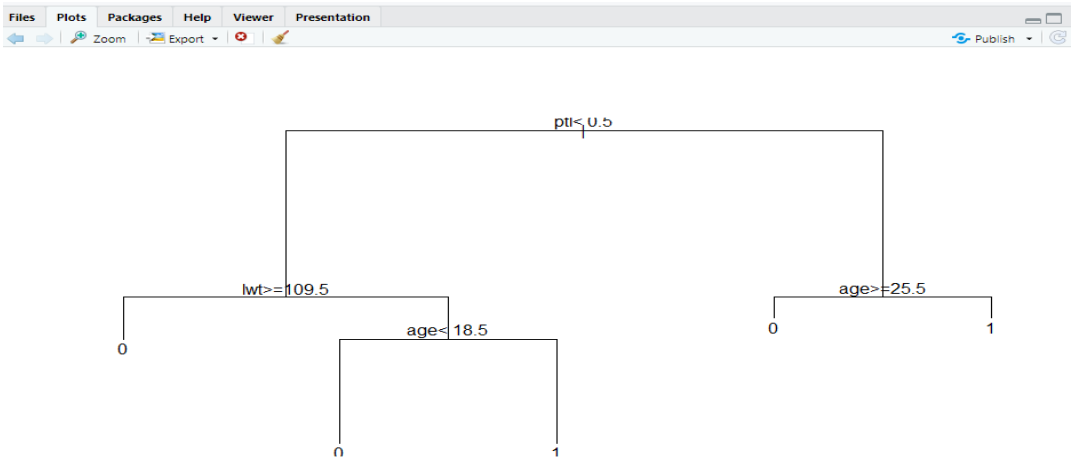


PROGRAM

```
library(MASS)
library(rpart)
head(birthwt)
hist(birthwt$b
wt)
table(birthwt$l
ow)
cols <- c('low', 'race', 'smoke', 'ht', 'ui')
birthwt[cols]<-
lapply(birthwt[cols],as.factor) set.seed(1)
train<-sample(1:nrow(birthwt),0.75*nrow(birthwt))
birthwtTree<-rpart(low~.-bwt,data=birthwt[train,],method='class')
plot(birthwtTree)
text(birthwtTree,pretty=0)
summary(birthwtTree)
birthwtPred<-predict(birthwtTree,birthwt[-train,],type='class')
table(birthwtPred, birthwt[-train, ]$low)
```

OUTPUT:

```
> source("~/Decision Tree.R", echo=TRUE)

> library(MASS)

Attaching package: 'MASS'

The following object is masked _by_ '.GlobalEnv':

    birthwt


> library(rpart)

> head(birthwt)
   low age lwt race smoke ptl ht ui ftv  bwt
85   0  19 182    2     0   0  0  1   0 2523
86   0  33 155    3     0   0  0  0   3 2551
87   0  20 105    1     1   0  0  0   1 2557
88   0  21 108    1     1   0  0  1   2 2594
89   0  18 107    1     1   0  0  1   0 2600
91   0  21 124    3     0   0  0  0   0 2622
```



PROGRAM:

```
library(dataset)
head(iris)
library(ggplot2)
ggplot(iris,aes(Petal.Length,Petal.Width,color=Species))+geom_point()
set.seed(20)
irisCluster<-kmeans(iris[,3:4],3,nstart=20)
irisCluster
table(irisCluster$cluster, iris$Species)
```

OUTPUT:

```
> source("~/Clustering Techniques.R", echo=TRUE)

> input<- mtcars [,c("am","cyl","hp","wt")]

> print(head(input))
                  am cyl  hp    wt
Mazda RX4          1   6 110 2.620
Mazda RX4 Wag      1   6 110 2.875
Datsun 710         1   4  93 2.320
Hornet 4 Drive     0   6 110 3.215
Hornet Sportabout  0   8 175 3.440
Valiant            0   6 105 3.460

> input<- mtcars [,c("am","cyl","hp","wt")]

> am.data =glm(formula = am ~ cyl+hp+wt,data = input,family = binomial)

> print(summary(am.data))

Call:
glm(formula = am ~ cyl + hp + wt, family = binomial, data = input)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) 19.70288    8.11637   2.428   0.0152 *
cyl          0.48760    1.07162   0.455   0.6491
hp           0.03259    0.01886   1.728   0.0840 .
wt          -9.14947    4.15332  -2.203   0.0276 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 43.2297  on 31  degrees of freedom
Residual deviance:  9.8415  on 28  degrees of freedom
AIC: 17.841

Number of Fisher Scoring iterations: 8
```

1.Histogram

Histogramis basicallya plot that breaks the data into bins (or breaks) and shows frequency distribution of these bins. You can change the breaks also and see the effect it has data visualization in terms of understandability.

Note: We have used par(mfrow=c(2,5)) command to fit multiple graphs in same page for sake of clarity(see the code below).

PROGRAM:

```
library(RColorBrewer)

data(VADeaths) par(mfrow=c(2,3)) hist(VADeaths,breaks=10,
col=brewer.pal(3,"Set3"),main="Set33colors")hist(VADeaths,breaks=3
,col=brewer.pal(3,"Set2"),main="Set2 3 colors") hist(VADeaths,breaks=7,
col=brewer.pal(3,"Set1"),main="Set13colors")hist(VADeaths,,breaks=2,
col=brewer.pal(8,"Set3"),main="Set38colors")
hist(VADeaths,col=brewer.pal(8,"Greys"),main="Greys8colors")
hist(VADeaths,col=brewer.pal(8,"Greens"),main="Greens8colors")
```

OUTPUT:

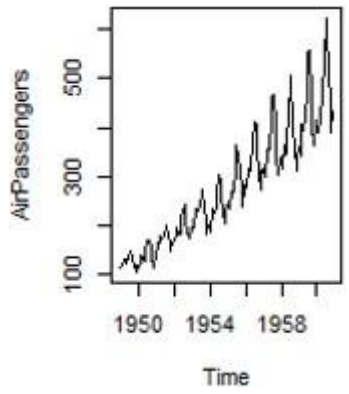
```
> source("~/Visualize data using plotting framework.R", echo=TRUE)

> library(e1071)

> plot(iris)

> iris
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
1            5.1         3.5          1.4         0.2     setosa
2            4.9         3.0          1.4         0.2     setosa
3            4.7         3.2          1.3         0.2     setosa
4            4.6         3.1          1.5         0.2     setosa
5            5.0         3.6          1.4         0.2     setosa
6            5.4         3.9          1.7         0.4     setosa
7            4.6         3.4          1.4         0.3     setosa
8            5.0         3.4          1.5         0.2     setosa
9            4.4         2.9          1.4         0.2     setosa
10           4.9         3.1          1.5         0.1     setosa
```

2.1. Line Chart

Below is the line chart showing the increase in air passengers over given time period. Line Charts are commonly preferred when we are to analyses a trend spread over a time period. Furthermore, line plot is also suitable to plots where we need to compare relative changes in quantities across some variable (like time). Below is the code:

PROGRAM:

```
data(AirPassengers)

plot(AirPassengers,type="l")#Simple Line Plot
```



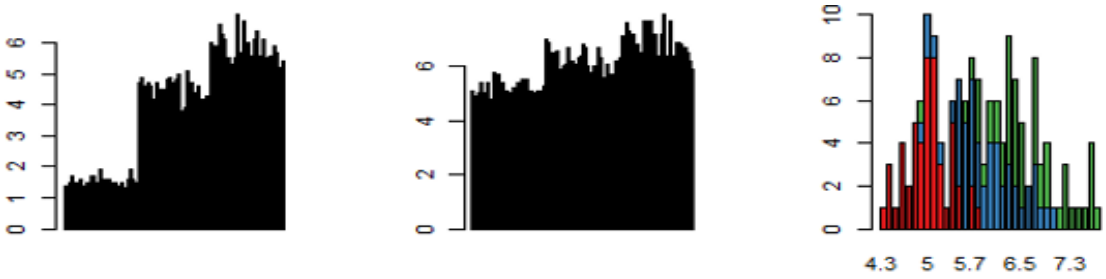
2.2. Bar Chart

Bar Plots are suitable for showing comparison between cumulative totals across several groups. Stacked Plots are used for bar plots for various categories. Here's the code:

PROGRAM:

```
data("iris")

barplot(iris$Petal.Length)#Creating simple Bar Graph

barplot(iris$Sepal.Length,col=brewer.pal(3,"Set1"))
barplot(table(iris$Species,iris$Sepal.Length),col=brewer.pal(3,"Set1"))#StackedPlot
```

OUTPUT:

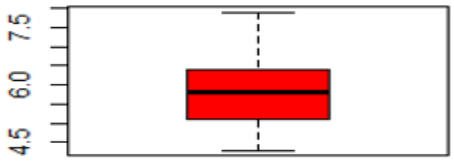


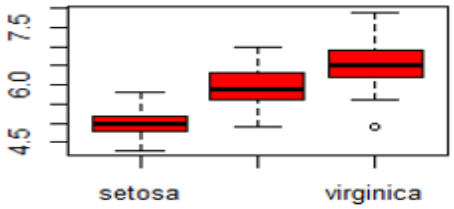
3. **Box Plot** Box Plot shows 5 statistically significant numbers the minimum, the 25th percentile, the median, the 75th percentile and the maximum. It is thus useful for visualizing the spread of the data is and deriving inferences accordingly.

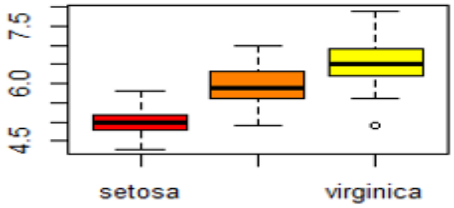
PROGRAM:

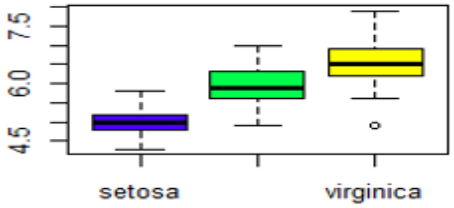
```
data(iris)
par(mfrow=c
(2,2))
boxplot(iris$Sepal.Length,col="red")
boxplot(iris$Sepal.Length~iris$Species,col="red")
boxplot(iris$Sepal.Length~iris$Species,col=heat.colors(3))
boxplot(iris$Sepal.Length~iris$Species,col=topo.colors(3))
boxplot(iris$Petal.Length~iris$Species)#CreatingBoxPlotbetweentwovariable
```

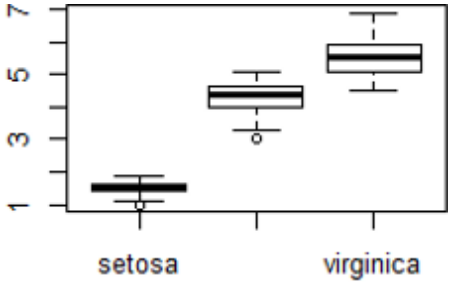
OUTPUT:









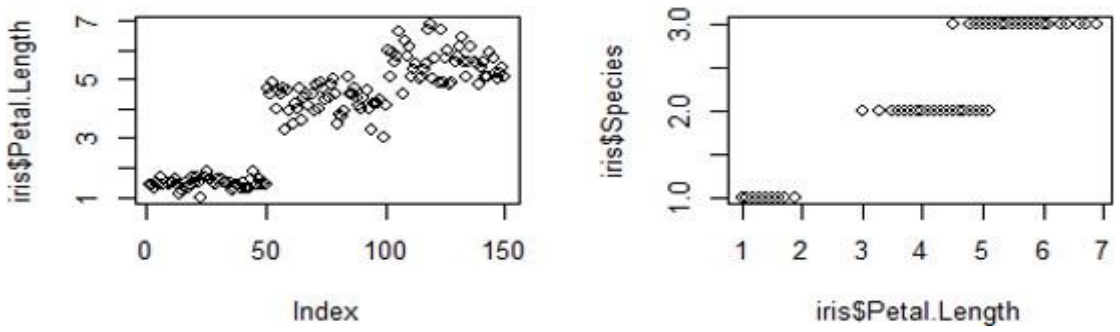


4. Scatter Plot(including3Dandother features)Scatter plots help in visualizing data easily and for simple data inspection. Here's the code for simple scatter and multivariate scatter plot:

PROGRAM:

```
plot(x=iris$Petal.Length) #Simple Scatter Plot

plot(x=iris$Petal.Length,y=iris$Species)#MultivariateScatterPlot
```

OUTPUT:

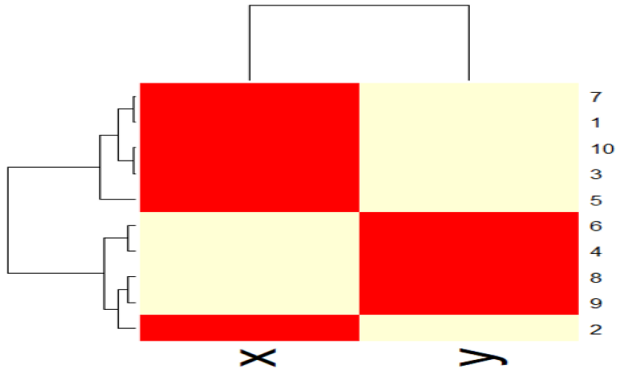


5. **Heat Map** : one of the most innovative data visualizations in R, the heat map emphasizes color intensity to visualize relationships between multiple variables. The result is an attractive 2D image that is easy to interpret. As a basic example, a heat map highlights the popularity of competingitems by ranking them according to their original market launch date. It breaks it down further by providing sales statistics and figures over the course of time.

PROGRAM:

```
# simulate a dataset of 10 points

x<-rnorm(10,mean=rep(1:5,each=2),sd=0.7)

y<-rnorm(10,mean=rep(c(1,9),each=5),sd=0.1)

dataFrame<-data.frame(x=x,y=y)

set.seed(143)

dataMatrix<-as.matrix(dataFrame)[sample(1:10),] # convert to class 'matrix', then shuffle the rows of the matrix

heatmap(dataMatrix)#visualize hierarchical clustering via a heatmap
```

OUTPUT:

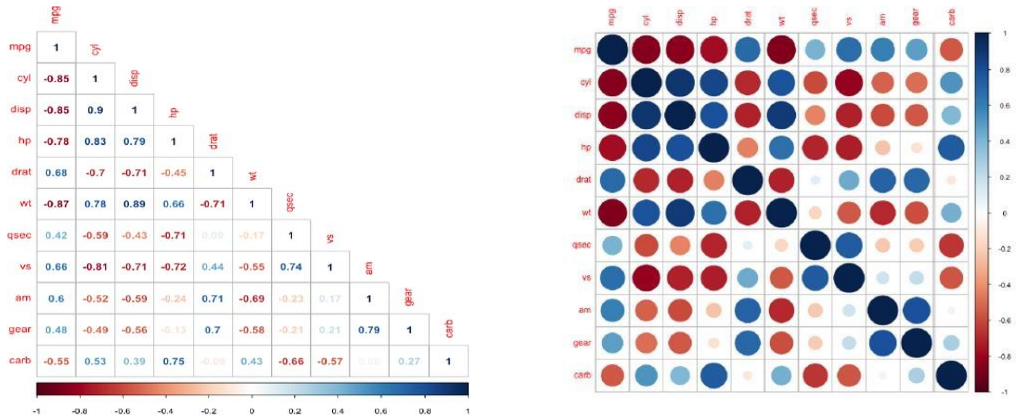


6. **Correlogram** : Correlated data is best visualized through corrplot. The 2D format is similar to a heat map, but it highlights statistics that are directly related. Most correlograms highlight theamount of correlation between datasets at various points in time. Comparing sales data between different months or years is a basic example.

PROGRAM:

```
#data("mtcars")

corr_matrix<-cor(mtcars)

#with circles corrplot(corr_matrix)
#with numbers and lower corrplot(corr_matrix,method='number',type="lower")
```

OUTPUT:

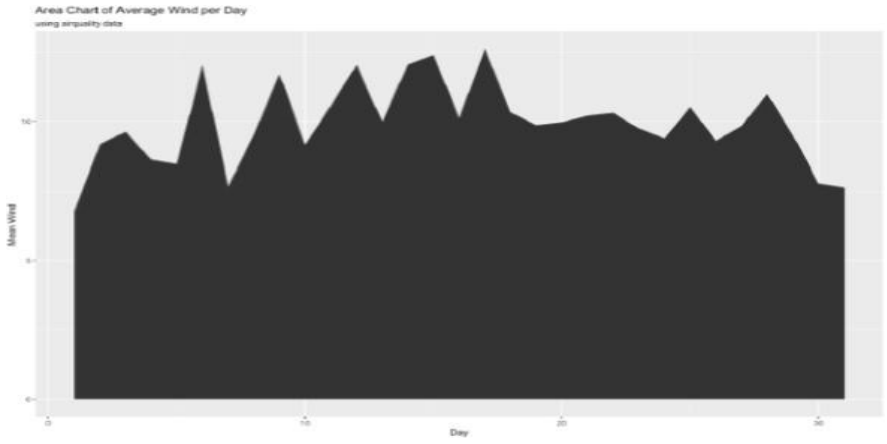


7. **Area Chart:** Area charts express continuitybetween different variables or data sets. It'sakintothe traditional line chart you know fromgrade school and is used in a similar fashion. Most area charts highlight trends and their evolution over the course of time, making them highly effective when trying to expose underlying trends whether they're positive or negative.

PROGRAM:

```
data("airquality")#dataset used

airquality %>%

group_by(Day) %>%

summarise(mean_wind=mean(Wind))%>%

ggplot() +

geom_area(aes(x = Day, y = mean_wind)) +

labs(title= "Area Chart of AverageWindper Day",

subtitle="usingairqualitydata",y="MeanWind")
```

OUTPUT:



1) To use MongoDB with R, first, we have to download and install MongoDB Next, start MongoDB. We can start MongoDB like so:

mongod

2) Insertingdata

Let's insert the crimes data from data.gov to MongoDB. The dataset reflects reported incidents of crime (with the exception of murders where data exists for each victim) that occurred in the City of Chicago since 2001.

```
library (ggplot2)

library (dplyr)

library (maps)

library (ggmap)

library (mongolite)

library (lubridate)

library (gridExtra)

crimes=data.table::fread("Crimes_2001_to_present.csv")

names (crimes)
```

OUTPUT:

ID' 'Case Number' 'Date' 'Block' 'IUCR' 'Primary Type' 'Description' 'LocationDescription' 'Arrest''Domestic' 'Beat' 'District' 'Ward' 'Community Area' 'FBI Code' 'X Coordinate' 'Y Coordinate' 'Year' 'Updated On' 'Latitude' 'Longitude' 'Location'.

3) Let’s remove spaces in the column names to avoid anyproblems when we query it from MongoDB.

```
names(crimes)=gsub("","",names(crimes))

names(crimes)
```

4) Let’s use the insert function fromthe mongolite package to insert rowsto a collection in MongoDB.Let’s create a database called Chicago and call the collection crimes.

```
my_collection=mongo(collection="crimes",db="Chicago")#createconnection,datab
ase and collection

my_collection$insert(crimes)
```

OUTPUT:

```
'ID' 'CaseNumber' 'Block' 'PrimaryType 'Description
'LocationDescription''Arrest''Domestic''Beat''District''Ward''CommunityArea'
'FBICode''XCoordinate''YCoordinate''Year''UpdatedOn''Latitude''Longitude'
'Location'
```

5) Let’s check if we have inserted the“crimes”data.

```
my_collection$count()
```

OUTPUT: 6261148

We see that the collection has 6261148 records.

6) First,let’s look what the data looks like by displaying one record:

```
my_collection$iterate()$one()
```

OUTPUT:

```
$ID
 1454164
 $CaseNumber
' G185744'
 $Date
'04/01/200106:00:00PM'
 $Block
'049XXNMENARDAV'
```

```
 $IUCR

'0910'
 $PrimaryType
'MOTORVEHICLETHEFT'
 $Description
'AUTOMOBILE'
 $LocationDescription
' STREET'
 $Arrest
' false'
 $Domestic
' false'
 $Beat
 1622
 $District
 16
 $FBICode
' 07'
 $XCoordinate
 1136545

 $YCoordinate
 1932203
 $Year
 2001
 $UpdatedOn
'08/17/201503:03:40PM'
 $Latitude
 41.970129962
 $Longitude
 87.773302309
 $Location
'(41.970129962,-87.773302309)'
```

7) How many distinct “PrimaryType”do we have?

length(my_collection$distinct("PrimaryType"))

OUTPUT:

35

As shown above, there are 35 different crime primary types in the database. Wewill seethe patterns of the most common crime types below.

8) Now, let's see how many domestic assaults there are in the collection.

```
my_collection$count('{"PrimaryType":"ASSAULT","Domestic":"true"}')
```

OUTPUT:

8247

9) To get the filtered data and we can also retrieve onlythe columns of

interest.

```
query1= my_collection$find('{"PrimaryType" : "ASSAULT", "Domestic" :
"true" }')

query2= my_collection$find('{"PrimaryType" : "ASSAULT", "Domestic" :
"true" }',

fields = '{"_id":0, "PrimaryType":1, "Domestic":1}')

ncol(query1) # with all the columns

ncol(query2) # only the selected columns
```

OUTPUT:

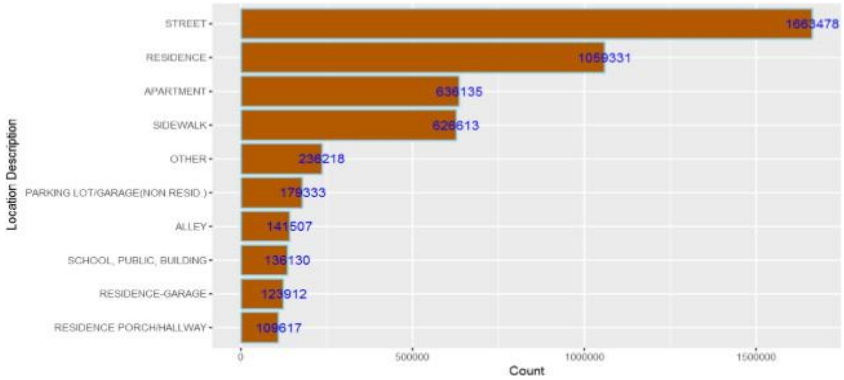
22

2

10) To find out "Where do most crimes take place?" use the following command.

```
my_collection$aggregate('[{"$group":{"_id":"$LocationDescription",
"Count":

{"$sum":1}}}]')%>%na.omit()%>%
arrange(desc(Count))%>%head(10)%>%ggplot(aes(x=reorder(`_id`,Count),
y=Count))+geom_bar(stat="identity",color='skyblue',fill='#b35900')+geom_
text(aes(label = Count), color = "blue")
+coord_flip()+xlab("Locationdescription")
```

OUTPUT:



11)If loading the entire dataset we are working with does not slow downour analysis, we can use data.table or dplyr but when dealing with big data, using MongoDB can give us performanceboost asthewholedatawillnot beloaded intomemory. Wecanreproducethe above plot without using MongoDB, like so:

```
crimes%>%group_by(`LocationDescription`)%>%summarise(Total=n())%>%

arrange(desc(Total))%>%head(10)%>%

ggplot(aes(x=reorder(`LocationDescription`,Total),y=Total))+

geom_bar(stat="identity",color='skyblue',fill='#b35900')+geom_text(aes(label

= Total), color = "blue") +coord_flip()+xlab("Location Description")
```

OUTP