Geethanjali College of Engineering and Technology

(UGC AUTONOMOUS INSTITUTION)

(Accredited by NBA and NAAC with ‘A’ grade, Approved by AICTE New Delhi and Affiliated to JNTUH)
Cheeryal (V), Keesara (M), Medchal (Dist.), Telangana – 501 301.

DATA ANALYTICS LAB
(18CS41L1)
Laboratory Manual

IV Year B.Tech. CSE I Semester

Geethanjali

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

2021-2022

Lab Incharge **HOD-CSE**

Dr. A. Sree Lakshmi

Geethanjali College of Engineering and Technology
(UGC AUTONOMOUS INSTITUTION)

(Accredited by NBA and NAAC with 'A' grade, Approved by AICTE New Delhi and Affiliated to JNTUH) Cheeryal (V), Keesara (M), Medchal (Dist.), Telangana – 501 301.

DATA ANALYTICS LAB

(18CS41L1)

Laboratory Workbook

Name: ________________________________

Roll No: ___________ Class: ______________ Branch: _______________

Academic Year: 2021 – 2022

Geethanjali College of Engineering and Technology
(UGC AUTONOMOUS INSTITUTION)

(Accredited by NBA and NAAC with ‘A’ grade, Approved by AICTE New Delhi and Affiliated to JNTUH)

Cheeryal (V), Keesara (M), Medchal (Dist.), Telangana – 501 301.

Geethanjali

CERTIFICATE

This is to certify that Mr. / Miss ____________________________________ has satisfactorily completed________ number of experiments in the ***Data Analytics Laboratory.***

Roll No: ____________ Branch: ______________
Section: _____________ Year: ________________
Academic Year: ______________

Head **Faculty**

Dept. of CSE **In-Charge**

Internal Examiner **External Examiner**

GEETHANJALI COLLEGE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Cheeryal (V), Keesara (M), Medchal Dist., Telangana-501301

18CS41L1-DATA ANALYTICS LAB

IV Year. B.Tech. (CSE) – I Sem

L	T	P/D	C
-	-	2/-	1

Prerequisite(s):

- 18CS2102 - Object Oriented Programming using Java
- 18CS2203 -Database Management Systems

Course Objectives:

Develop ability to

1. Know the basic elements of Big Data and Data science to handle huge amount of data.
2. Gain knowledge of basic mathematics behind the Big data.
3. Understand the different Big data processing technologies.
4. Apply the Analytical concepts of Big data using R and Python.
5. Visualize the Big Data using different tools.

Course Outcomes (COs):

At the end of the course, student would be able to:

CO1: Observe Big Data elements and Architectures.
CO2: Apply different mathematical models for Big Data.
CO3: Demonstrate their Big Data skills by developing different applications.
CO4: Apply each learning model for different datasets.
CO5: Analyze needs, challenges and techniques for big data visualization.

LIST OF EXPERIMENTS

Week 1: Installation, Configuration, and Running of Hadoop and HDFS.

Week 2: Implementation of Word Count / Frequency Programs using MapReduce.

Week 3: Implementation of MR Program that processes a Weather Dataset.

Week 4: Implementation of Linear and Logistic Regression.

Week 5: Implementation of SVM Classification Technique.

Week 6: Implementation of Decision Tree Classification Technique.

Week 7: Implementation of Hierarchical Clustering.

Week 8: Implementation of Partitioning Clustering.

Week 9: Data Visualization using Pie, Bar, Boxplot Chart Plotting Framework.

Week 10: Data Visualization using Histogram Plotting Framework.

Week 11: Data Visualization using Line Graph Plotting, Scatterplot Plotting Framework.

Week 12: Application to analyze Stock Market Data using R Language.

Vision of the Institution

Geethanjali visualizes dissemination of knowledge and skills to students, who eventually contribute to well-being of the people of the nation and global community.

Mission of the Institution

- To impact adequate fundamental knowledge in all basic science and engineering technical and Inter - personal skills so students
- To bring out creativity in students that would promote innovation, research and entrepreneurship.
- To Preserve and promote cultural heritage, humanistic and spiritual values promoting peace and harmony in society.

Vision of the Department

To produce globally competent and socially responsible computer science engineers contributing to the advancement of engineering and technology which involves creativity and innovation by providing excellent learning environment with world class facilities

Mission of the Department

- To be a center of excellence in instruction, innovation in research and scholarship, and service to the stake holders, the profession, and the public.
- To prepare graduates to enter a rapidly changing field as a competent computer science engineer.
- To prepare graduate capable in all phases of software development, possess a firm understanding of hardware technologies, have the strong mathematical background necessary for scientific computing, and be sufficiently well versed in general theory to allow growth within the discipline as it advances.
- To prepare graduates to assume leadership roles by possessing good communication skills, the ability to work effectively as team members, and an appreciation for their social and ethical responsibility in a global setting.

PROGRAM EDUCATIONAL OBJECTIVES

- To provide graduates with a good foundation in mathematics, sciences and engineering fundamentals required to solve engineering problems that will facilitate them to find employment in industry and / or to pursue postgraduate studies with an appreciation for lifelong learning.

- To provide graduates with analytical and problem-solving skills to design algorithms, other hardware / software systems, and inculcate professional ethics, inter-personal skills to work in a multi-cultural team.

- To facilitate graduates to get familiarized with the art software / hardware tools, imbibing creativity and innovation that would enable them to develop cutting-edge technologies of multi-disciplinary nature for societal development.

PROGRAM OUTCOMES (POs)

Program Outcomes (POs) describe what students are expected to know and be able to do by the time of graduation to accomplish Program Educational Objectives (PEOs). The Program Outcomes for Computer Science and Engineering graduates are:

Engineering Graduates would be able to:

PO 1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO 2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO 3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO 4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO 5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO 6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO 7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO 8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO 9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO 10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO 11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO 12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO 1: To identify and define the computing requirements appropriate for its solution under given constraints.

PSO 2: To follow the best practices, namely, SEI-CMM levels and 6-sigma which varies from time to time for software development projects using open-ended programming environments to produce software deliverables as per customer needs.

6. Course Objectives and Course Outcomes

Course Objectives:

Develop ability to

1. Know the basic elements of Big Data and Data science to handle huge amount of data.
2. Gain knowledge of basic mathematics behind the Big data.
3. Understand the different Big data processing technologies.
4. Apply the Analytical concepts of Big data using R and Python.
5. Visualize the Big Data using different tools.

Course Outcomes (COs):

At the end of the course, student would be able to:

CO1. Observe Big Data elements and Architectures.
CO2. Apply different mathematical models for Big Data.
CO3. Demonstrate their Big Data skills by developing different applications.
CO4. Apply each learning model for different datasets.
CO5. Analyze needs, challenges and techniques for big data visualization.

Mapping of Lab Course with Programme Educational Objectives

Course	Course Code	PEOs	POs & PSOs
DATA ANALYTICS LAB	**18CS41L1**	**PEO1, PEO2, PEO3**	**PO1, PO2, PO3, PO4, PO5, PO11, PO12, PSO1, PSO2**

Mapping of Lab Course outcomes with Programme outcomes:

Course Outcomes - DATA ANALYTICS (18CS41L1)	**Program Outcomes and Program Specific Outcomes**													
	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1. Observe Big Data elements and Architectures.	2	1	1	1	1	-	-	-	-	-	2	2	1	2
CO2. Apply different mathematical models for Big Data.	1	1	2	3	2	-	-	-	-	-	2	2	1	2
CO3. Demonstrate their Big Data skills by developing different applications.	2	1	1	2	1	-	-	-	-	-	1	2	1	2
CO4. Apply each learning model for different datasets.	1	1	1	2	1	-	-	-	-	-	1	2	1	2
CO5. Analyze needs, challenges and techniques for big data visualization	2	1	1	2	1	-	-	-	-	-	1	2	1	2

Prerequisites:

- 18CS2102 - Object Oriented Programming using Java
- 18CS2203 -Database Management Systems

INSTRUCTIONS TO THE STUDENTS:

1. Students are required to attend all labs.
2. Students should be dressed in formals when attending the laboratory sessions.
3. Students will work individually in computer laboratories.
4. While coming to the lab bring the observation book and Work book etc.
5. Before coming to the lab, prepare the pre-lab questions. Read through the lab experiment to familiarize you.
6. Utilize 3 hours' time properly to perform the experiment and noting down the outputs.
7. If the experiment is not completed in the prescribed time, the pending work has to be done in the leisure hour or extended hours.
8. You will be expected to submit the completed work book according to the deadlines set up by your instructor.

INSTRUCTIONS TO LABORATORY TEACHERS:

- Observation book and lab records submitted for the lab work are to be checked and signed before the next lab session.
- Students should be instructed to switch ON the power supply after the connections are checked by the lab assistant / teacher.
- The promptness of submission should be strictly insisted by awarding the marks accordingly.
- Ask viva questions at the end of the experiment.
- Do not allow students who come late to the lab class.
- Encourage the students to do the experiments innovatively.
- Fill continuous Evaluation sheet, on regular basis.
- Ensure that the students are dressed in formals

Scheme of Lab Exam Evaluation:

Evaluation of Internal Marks:

a) 15 Marks are awarded for day-to-day work

1) Record and Observation book --------- 5Marks

2) Attendance and behavior of student --------- 5 Marks

3) Viva and performance ----------------5 Marks

b) 15 Marks are awarded for conducting laboratory test as follows:

1) Write up and program--------5 Marks

2) Execution of Program ---------5 Marks

3) Viva and performance ----------------5 Marks

Evaluation of External Marks:

70 Marks are awarded for conducting laboratory test as follows:

1) Algorithm ------------------- 25 Marks.

2) Write up and program--------- 15 Marks

3) Execution of Program --------- 15 Marks

4) Viva ---------------------- 15 Marks

PERFORMANCE INDICATOR

S.No.	Name of Experiment	Date of Exp.	Date of Submission	Marks	Signature	Remarks

S.No.	Name of Experiment	Date of Exp.	Date of Submission	Marks	Signature	Remarks

WEEK 1

INSTALLATION, CONFIGURATION AND RUNNING HADOOP AND HDFS.

PROCEDURE:

1. Installing Java

```
prince@prince-VirtualBox:~$ cd ~
```

Update the source list

```
prince@prince-VirtualBox:~$ sudo apt-get update
```

The OpenJDK project is the default version of Java
that is provided from a supported Ubuntu repository.

```
prince@prince-VirtualBox:~$ sudo apt-get install default-jdk
```

```
prince@prince-VirtualBox:~$ java -version
```

```
java version "1.7.0_65"
OpenJDK Runtime Environment (IcedTea 2.5.3) (7u71-2.5.3-0ubuntu0.14.04.1)
OpenJDK 64-Bit Server VM (build 24.65-b04, mixed mode)
```

2. Adding a dedicated Hadoop user

```
prince@prince-VirtualBox:~$ sudo add group hadoop
```

```
 Adding group `hadoop' (GID 1002) ...
 Done.
```

```
prince@prince-VirtualBox:~$ sudo add user –ingroup hadoop hduser
  Adding user `hduser' ...
  Adding new user `hduser' (1001) with group `hadoop' ...
  Creating home directory `/home/hduser' ...
  Copying files from `/etc/skel' ...
  Enter new UNIX password:
  Retype new UNIX password:
passwd: password updated successfully
  Changing the user information for hduser
  Enter the new value, or press ENTER for the default
          Full Name []:
          Room Number []:
          Work Phone []:
          Home Phone []:
          Other []:
Is the information correct? [Y/n] Y
```

3. Installing SSH

ssh has two main components:

1. **ssh** : The command we use to connect to remote machines - the client.
2. **sshd** : The daemon that is running on the server and allows clients to connect to the server.

The **ssh** is pre-enabled on Linux, but in order to start **sshd** daemon, we need to install **ssh** first. Use this command to do that :

```
prince@prince-VirtualBox:~$ sudo apt-get install ssh
```

This will install ssh on our machine. If we get something similar to the following, we can think it is setup properly:

```
prince@prince-VirtualBox:~$ which ssh

/usr/bin/ssh

prince@prince-VirtualBox:~$ which sshd
/usr/sbin/sshd
```

4. Create and Setup SSH Certificates

Hadoop requires SSH access to manage its nodes, i.e. remote machines plus our local

machine. For our single-node setup of Hadoop, we therefore need to configure SSH access to localhost.

So, we need to have SSH up and running on our machine and configured it to allow SSH public key authentication.

Hadoop uses SSH (to access its nodes) which would normally require the user to enter a password. However, this requirement can be eliminated by creating and setting up SSH certificates using the following commands. If asked for a filename just leave it blank and press the enter key to continue.

```
prince@prince-VirtualBox:~$ suhduser
Password:
prince@prince-VirtualBox:~$ ssh-keygen -t rsa -P ""

Generating public/private rsa key pair.
Enter file in which to save the key (/home/hduser/.ssh/id_rsa):
Created directory '/home/hduser/.ssh'.
Your identification has been saved in /home/hduser/.ssh/id_rsa.
Your public key has been saved in /home/hduser/.ssh/id_rsa.pub.
The key fingerprint is:
50:6b:f3:fc:0f:32:bf:30:79:c2:41:71:26:cc:7d:e3
```

```
hduser@prince-VirtualBox
The key's randomart image is:
+--[ RSA 2048]----+
|       .oo.o    |
|      . .o=. o  |
|     . + .  o . |
|      o =    E  |
|       S +      |
|        . +     |
|         O +    |
|          O o   |
|           o..  |
+-----------------+
```

```
hduser@prince-VirtualBox:/home/k$ cat $HOME/.ssh/id_rsa.pub >>
$HOME/.ssh/authorized_keys
```

The second command adds the newly created key to the list of authorized keys so that Hadoop can use ssh without prompting for a password.

We can check if ssh works:

```
hduser@prince-VirtualBox:/home/k$ sshlocalhost
```

```
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is e1:8b:a0:a5:75:ef:f4:b4:5e:a9:ed:be:64:be:5c:2f.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-40-generic x86_64)
...
```

5. Install Hadoop

```
hduser@prince-VirtualBox:~$ wget
http://mirrors.sonic.net/apache/hadoop/common/hadoop-2.6.0/hadoop-2.6.0.tar.gz
hduser@prince-VirtualBox:~$ tar xvzf hadoop-2.6.0.tar.gz
```

We want to move the Hadoop installation to the **/usr/local/hadoop** directory using the following command:

```
hduser@prince-VirtualBox:~/hadoop-2.6.0$ sudo mv * /usr/local/hadoop
```

```
[sudo] password for hduser:
hduser is not in the sudoers file.  This incident will be reported.
```

Oops!... We got:

"hduser is not in the sudoers file. This incident will be reported."

This error can be resolved by logging in as a root user, and then add **hduser** to **sudo**:

```
hduser@prince-VirtualBox:~/hadoop-2.6.0$ su prince
Password:
```

```
prince@prince-VirtualBox:/home/hduser$ sudo add user hduser sudo
```

```
[sudo] password for prince:
Adding user `hduser' to group `sudo' ...
Adding user hduser to group sudo
Done.
```

Now, the **hduser** has root priviledge, we can move the Hadoop installation to the

/usr/local/hadoop directory without any problem:

```
prince@prince-VirtualBox:/home/hduser$ sudosuhduser
```

```
hduser@prince-VirtualBox:~/hadoop-2.6.0$ sudo mv * /usr/local/hadoop
```

```
hduser@prince-VirtualBox:~/hadoop-2.6.0$ sudochown -R hduser:hadoop
/usr/local/hadoop
```

6. Setup Configuration Files

The following files will have to be modified to complete the Hadoop setup:

i.~/.bashrc

ii./usr/local/hadoop/etc/hadoop/hadoop-env.sh

iii./usr/local/hadoop/etc/hadoop/core-site.xml

iv./usr/local/hadoop/etc/hadoop/mapred-site.xml.template

v./usr/local/hadoop/etc/hadoop/hdfs-site.xml

i. ~/.bashrc:

Before editing the **.bashrc** file in our home directory, we need to find the path where Java has been installed to set the **JAVA_HOME** environment variable using the following command:

```
hduser@prince-VirtualBox:~$ update-alternatives --config java
```

```
There is only one alternative in link group java (providing /usr/bin/java): /usr/lib/jvm/java-7-
openjdk-amd64/jre/bin/java
Nothing to configure.
```

Now we can append the following to the end of **~/.bashrc**:

```
hduser@prince-VirtualBox:~$ nano ~/.bashrc
```

```
#HADOOP VARIABLES START
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
export HADOOP_INSTALL=/usr/local/hadoop
export PATH=$PATH:$HADOOP_INSTALL/bin
export PATH=$PATH:$HADOOP_INSTALL/sbin
export HADOOP_MAPRED_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_HOME=$HADOOP_INSTALL
export HADOOP_HDFS_HOME=$HADOOP_INSTALL
export YARN_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_INSTALL/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_INSTALL/lib"
#HADOOP VARIABLES END
```

```
hduser@prince-VirtualBox:~$ source ~/.bashrc
```

note that the JAVA_HOME should be set as the path just before the '.../bin/':

```
hduser@ubuntu-VirtualBox:~$ javac -version
javac 1.7.0_75
```

```
hduser@ubuntu-VirtualBox:~$ which javac
/usr/bin/javac
```

```
hduser@ubuntu-VirtualBox:~$ readlink -f /usr/bin/javac
/usr/lib/jvm/java-7-openjdk-amd64/bin/javac
```

ii. /usr/local/hadoop/etc/hadoop/hadoop-env.sh

We need to set **JAVA_HOME** by modifying **hadoop-env.sh** file.

```
hduser@prince-VirtualBox:~$ nano /usr/local/hadoop/etc/hadoop/hadoop-env.sh
```

```
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
```

Adding the above statement in the **hadoop-env.sh** file ensures that the value of JAVA_HOME variable will be available to Hadoop whenever it is started up.

iii. /usr/local/hadoop/etc/hadoop/core-site.xml:

The **/usr/local/hadoop/etc/hadoop/core-site.xml** file contains configuration properties that Hadoop uses when starting up.
This file can be used to override the default settings that Hadoop starts with.

```
hduser@prince-VirtualBox:~$ sudomkdir -p /app/hadoop/tmp
```

```
hduser@prince-VirtualBox:~$ sudochownhduser:hadoop /app/hadoop/tmp
```

Open the file and enter the following in between the <configuration></configuration> tag:

```
hduser@prince-VirtualBox:~$ nano /usr/local/hadoop/etc/hadoop/core-site.xml
```

```
<configuration>
<property>
<name>hadoop.tmp.dir</name>
<value>/app/hadoop/tmp</value>
<description>A base for other temporary directories.</description>
</property>

<property>
<name>fs.default.name</name>
<value>hdfs://localhost:54310</value>
<description>The name of the default file system.  A URI whose
scheme and authority determine the FileSystem implementation.  The
uri's scheme determines the config property (fs.SCHEME.impl) naming
theFileSystem implementation class.  The uri's authority is used to
determine the host, port, etc. for a filesystem.</description>
</property>
</configuration>
```

iv. /usr/local/hadoop/etc/hadoop/mapred-site.xml

By default, the **/usr/local/hadoop/etc/hadoop/** folder contains **/usr/local/hadoop/etc/hadoop/mapred-site.xml.template** file which has to be renamed/copied with the name **mapred-site.xml**:

```
hduser@prince-VirtualBox:~$ cp /usr/local/hadoop/etc/hadoop/mapred-site.xml.template /usr/local/hadoop/etc/hadoop/mapred-site.xml
```

The **mapred-site.xml** file is used to specify which framework is being used for MapReduce. We need to enter the following content in between the <configuration></configuration> tag:

```
<configuration>
<property>
<name>mapred.job.tracker</name>
<value>localhost:54311</value>
<description>The host and port that the MapReduce job tracker runs
at.  If "local", then jobs are run in-process as a single map
and reduce task.
</description>
</property>
</configuration>
```

v. /usr/local/hadoop/etc/hadoop/hdfs-site.xml

The **/usr/local/hadoop/etc/hadoop/hdfs-site.xml** file needs to be configured for each host in the cluster that is being used.
It is used to specify the directories which will be used as the **namenode** and the **datanode** on that host.

Before editing this file, we need to create two directories which will contain the namenode and the datanode for this Hadoop installation.
This can be done using the following commands:

```
hduser@prince-VirtualBox:~$ sudomkdir -p /usr/local/hadoop_store/hdfs/namenode
hduser@prince-VirtualBox:~$ sudomkdir -p /usr/local/hadoop_store/hdfs/datanode
hduser@prince-VirtualBox:~$ sudochown -R hduser:hadoop /usr/local/hadoop_store
```

Open the file and enter the following content in between the <configuration></configuration> tag:

```
hduser@prince-VirtualBox:~$ nano /usr/local/hadoop/etc/hadoop/hdfs-site.xml
```

```
<configuration>
<property>
<name>dfs.replication</name>
<value>1</value>
<description>Default block replication.
  The actual number of replications can be specified when the file is created.
  The default is used if replication is not specified in create time.
</description>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value>file:/usr/local/hadoop_store/hdfs/namenode</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>file:/usr/local/hadoop_store/hdfs/datanode</value>
</property>
</configuration>
```

7. Format the NewHadoop File system

Now, the Hadoop file system needs to be formatted so that we can start to use it. The format command should be issued with write permission since it creates **current** directory under **/usr/local/hadoop_store/hdfs/namenode** folder:

```
hduser@prince-VirtualBox:~$ hadoop namenode -format
```

```
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.

/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at laptop/192.168.1.1
************************************************************/
```

Note that **hadoopnamenode -format** command should be executed once before we start using Hadoop.

If this command is executed again after Hadoop has been used, it'll destroy all the data on the Hadoop file system.

8. Starting Hadoop

Now it's time to start the newly installed single node cluster.
We can use **start-all.sh** or (**start-dfs.sh** and **start-yarn.sh**)

```
prince@prince-VirtualBox:~$ cd /usr/local/hadoop/sbin
```

```
prince@prince-VirtualBox:/usr/local/hadoop/sbin$ ls
```

```
distribute-exclude.sh    start-all.cmd        stop-balancer.sh
hadoop-daemon.sh         start-all.sh         stop-dfs.cmd
hadoop-daemons.sh        start-balancer.sh    stop-dfs.sh
hdfs-config.cmd          start-dfs.cmd        stop-secure-dns.sh
hdfs-config.sh           start-dfs.sh         stop-yarn.cmd
httpfs.sh                start-secure-dns.sh  stop-yarn.sh
kms.sh                   start-yarn.cmd       yarn-daemon.sh
mr-jobhistory-daemon.sh  start-yarn.sh        yarn-daemons.sh
refresh-namenodes.sh     stop-all.cmd
slaves.sh                stop-all.sh
```

```
prince@prince-VirtualBox:/usr/local/hadoop/sbin$ sudosuhduser
```

```
hduser@prince-VirtualBox:/usr/local/hadoop/sbin$ start-all.sh
hduser@prince-VirtualBox:~$ start-all.sh
```

```
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh
15/04/18 16:43:13 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting namenodes on [localhost]
localhost: starting namenode, logging to /usr/local/hadoop/logs/hadoop-hduser-namenode-laptop.out
localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-hduser-datanode-laptop.out
Starting secondary namenodes [0.0.0.0]
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-hduser-secondarynamenode-laptop.out
15/04/18 16:43:58 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
starting yarn daemons
startingresourcemanager, logging to /usr/local/hadoop/logs/yarn-hduser-resourcemanager-laptop.out
localhost: starting nodemanager, logging to /usr/local/hadoop/logs/yarn-hduser-nodemanager-laptop.out
```

We can check if it's really up and running:

```
hduser@prince-VirtualBox:/usr/local/hadoop/sbin$ jps

9026 NodeManager
7348 NameNode
9766 Jps
8887 ResourceManager
7507 DataNode
```

The output means that we now have a functional instance of Hadoop running on our VPS (Virtual private server).

9. Stopping Hadoop

```
$ pwd

/usr/local/hadoop/sbin

hduser@prince-VirtualBox:/usr/local/hadoop/sbin$ stop-all.sh

This script is Deprecated. Instead use stop-dfs.sh and stop-yarn.sh
15/04/18 15:46:31 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Stopping namenodes on [localhost]
localhost: stopping namenode
localhost: stopping datanode
Stopping secondary namenodes [0.0.0.0]
0.0.0.0: no secondarynamenode to stop
15/04/18 15:46:59 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
stopping yarn daemons
stoppingresourcemanager
localhost: stopping nodemanager
noproxyserver to stop
```

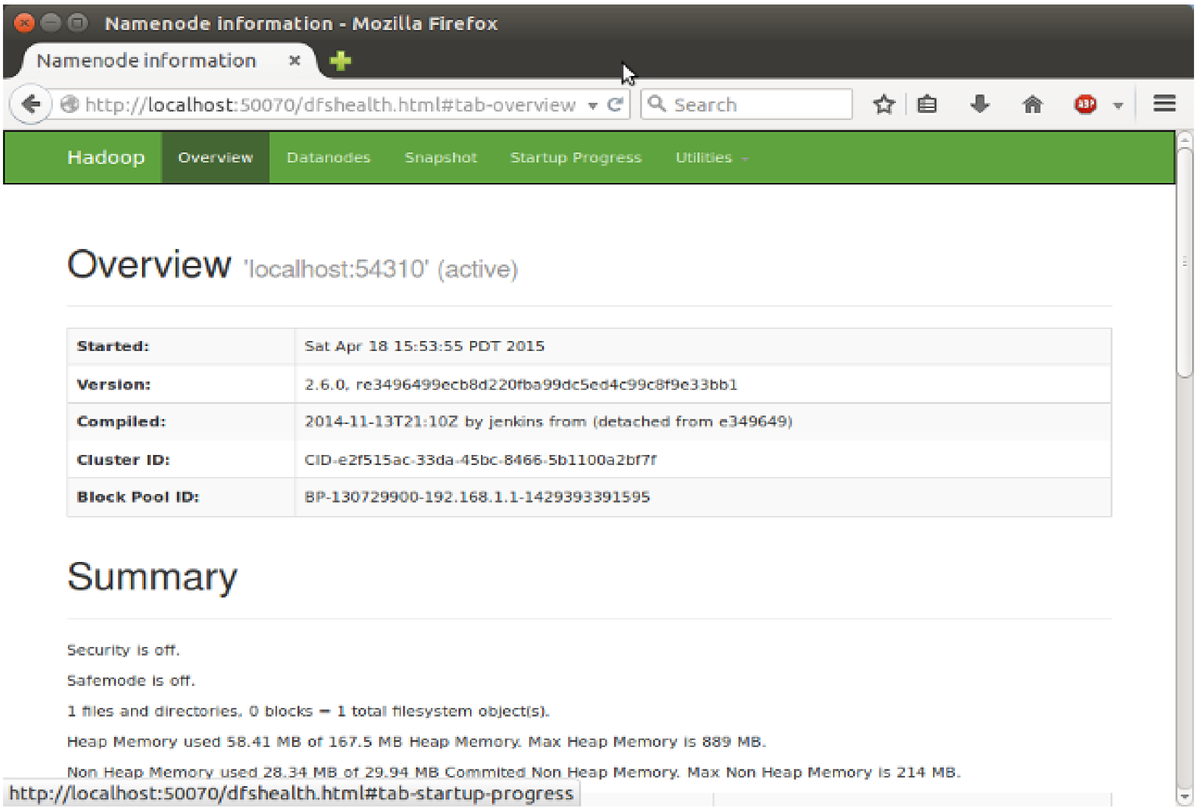
10.Hadoop Web Interfaces

Let's start the Hadoop again and see its Web UI:

hduser@prince-VirtualBox:/usr/local/hadoop/sbin$ start-all.sh

http://localhost:50070/ - web UI of the NameNode daemon

OUTPUT:



Namenode information - Mozilla Firefox
Namenode information
http://localhost:50070/dfshealth.html#tab-overview
Search
Hadoop
Overview
Datanodes
Snapshot
Startup Progress
Utilities

Overview 'localhost:54310' (active)

Started:	Sat Apr 18 15:53:55 PDT 2015
Version:	2.6.0, re3496499ecb8d220fba99dc5ed4c99c8f9e33bb1
Compiled:	2014-11-13T21:10Z by jenkins from (detached from e349649)
Cluster ID:	CID-e2f515ac-33da-45bc-8466-5b1100a2bf7f
Block Pool ID:	BP-130729900-192.168.1.1-1429393391595

Summary

Security is off.

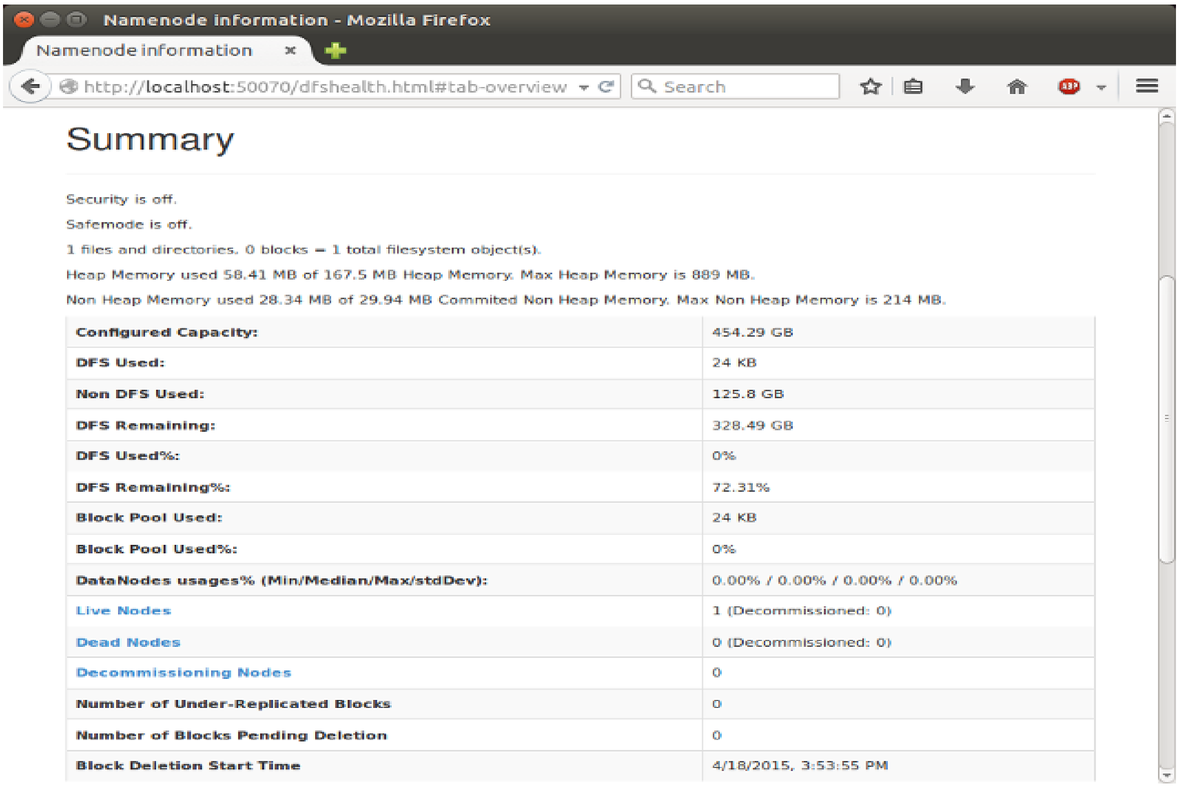
Safemode is off.

1 files and directories, 0 blocks = 1 total filesystem object(s).

Heap Memory used 58.41 MB of 167.5 MB Heap Memory. Max Heap Memory is 889 MB.

Non Heap Memory used 28.34 MB of 29.94 MB Commited Non Heap Memory. Max Non Heap Memory is 214 MB.

http://localhost:50070/dfshealth.html#tab-startup-progress


Namenode information - Mozilla Firefox
Namenode information
http://localhost:50070/dfshealth.html#tab-overview
Search

Summary

Security is off.

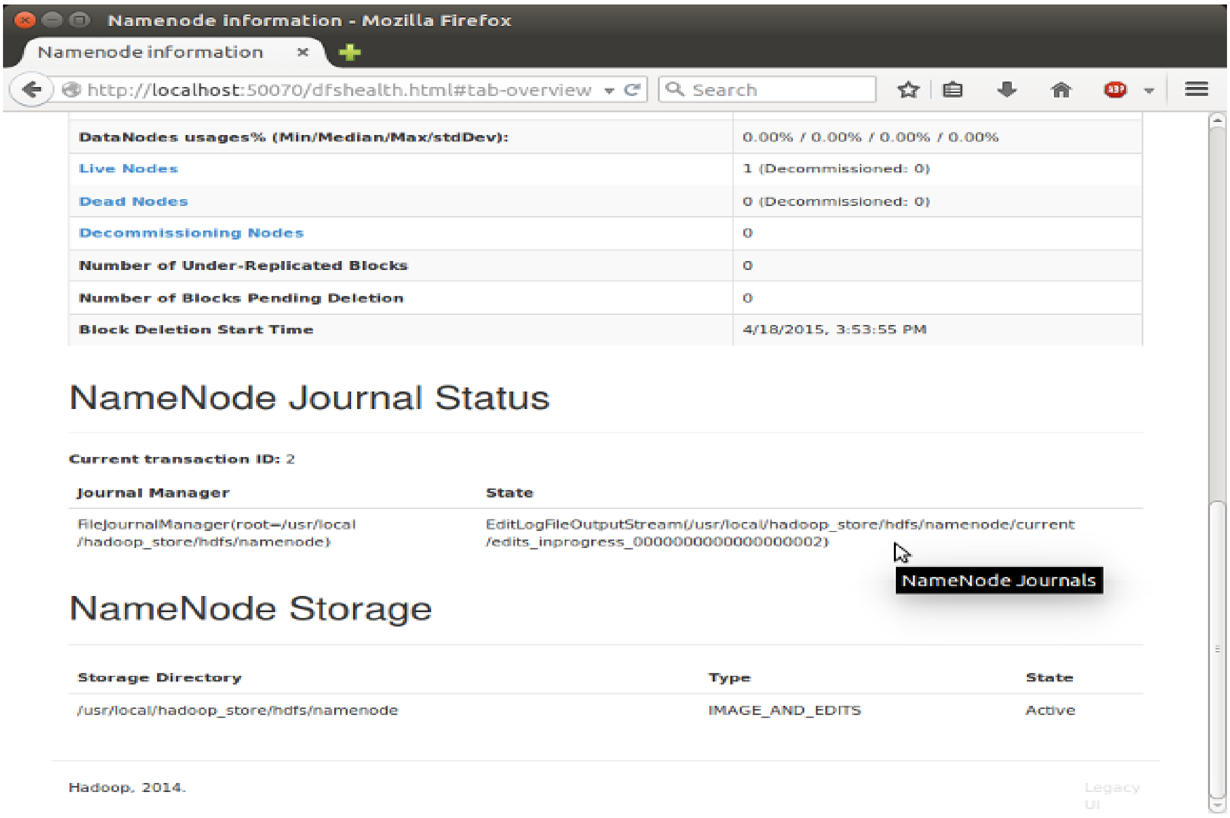
Safemode is off.

1 files and directories, 0 blocks = 1 total filesystem object(s).

Heap Memory used 58.41 MB of 167.5 MB Heap Memory. Max Heap Memory is 889 MB.

Non Heap Memory used 28.34 MB of 29.94 MB Commited Non Heap Memory. Max Non Heap Memory is 214 MB.

Configured Capacity:	454.29 GB
DFS Used:	24 KB
Non DFS Used:	125.8 GB
DFS Remaining:	328.49 GB
DFS Used%:	0%
DFS Remaining%:	72.31%
Block Pool Used:	24 KB
Block Pool Used%:	0%
DataNodes usages% (Min/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%
Live Nodes	1 (Decommissioned: 0)
Dead Nodes	0 (Decommissioned: 0)
Decommissioning Nodes	0
Number of Under-Replicated Blocks	0
Number of Blocks Pending Deletion	0
Block Deletion Start Time	4/18/2015, 3:53:55 PM


Namenode information - Mozilla Firefox
Namenode information
http://localhost:50070/dfshealth.html#tab-overview
Search

DataNodes usages% (Min/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%
Live Nodes	1 (Decommissioned: 0)
Dead Nodes	0 (Decommissioned: 0)
Decommissioning Nodes	0
Number of Under-Replicated Blocks	0
Number of Blocks Pending Deletion	0
Block Deletion Start Time	4/18/2015, 3:53:55 PM

NameNode Journal Status

Current transaction ID: 2

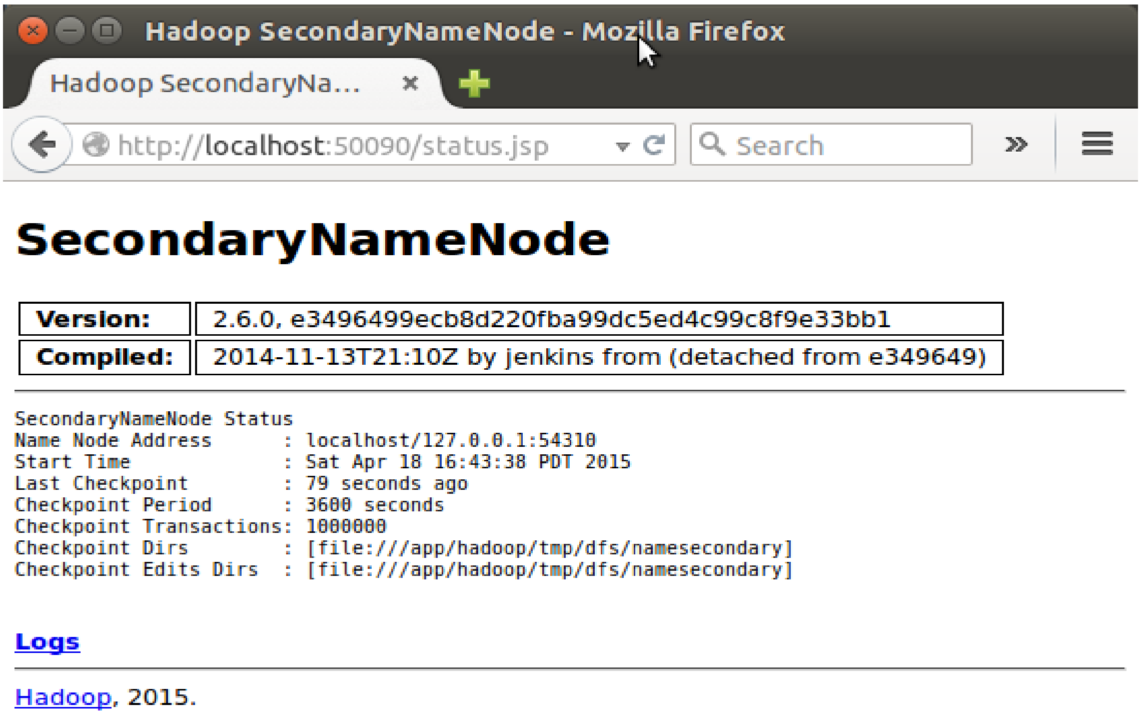
Journal Manager	State
FileJournalManager(root=/usr/local /hadoop_store/hdfs/namenode)	EditLogFileOutputStream(/usr/local/hadoop_store/hdfs/namenode/current /edits_inprogress_0000000000000000002)

NameNode Storage

Storage Directory	Type	State
/usr/local/hadoop_store/hdfs/namenode	IMAGE_AND_EDITS	Active

Hadoop, 2014.

Secondary Name Node


Hadoop SecondaryNameNode - Mozilla Firefox
Hadoop SecondaryNa...
http://localhost:50090/status.jsp
Search

SecondaryNameNode

Version:	2.6.0, e3496499ecb8d220fba99dc5ed4c99c8f9e33bb1
Compiled:	2014-11-13T21:10Z by jenkins from (detached from e349649)

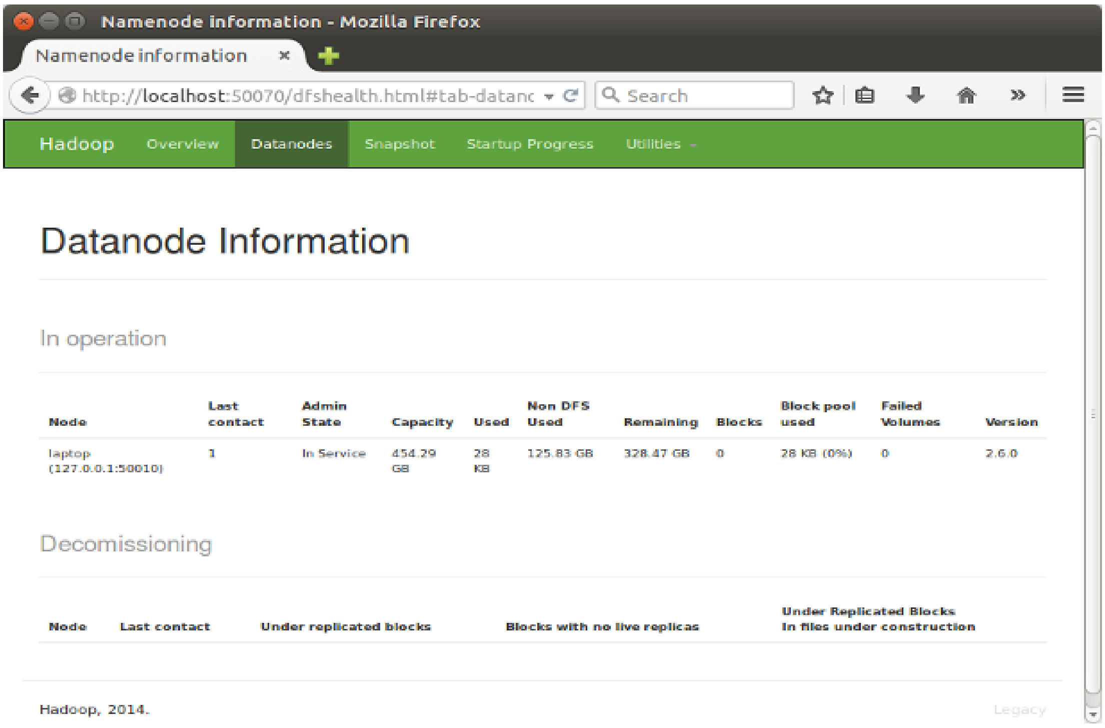
```
SecondaryNameNode Status
Name Node Address      : localhost/127.0.0.1:54310
Start Time             : Sat Apr 18 16:43:38 PDT 2015
Last Checkpoint        : 79 seconds ago
Checkpoint Period      : 3600 seconds
Checkpoint Transactions: 1000000
Checkpoint Dirs        : [file:///app/hadoop/tmp/dfs/namesecondary]
Checkpoint Edits Dirs  : [file:///app/hadoop/tmp/dfs/namesecondary]
```

Logs

Hadoop, 2015.

(Note) I had to restart Hadoop to get this Secondary Namenode.

Data Node


Namenode information - Mozilla Firefox
Namenode information
http://localhost:50070/dfshealth.html#tab-datanc
Search
Hadoop
Overview
Datanodes
Snapshot
Startup Progress
Utilities

Datanode Information

In operation

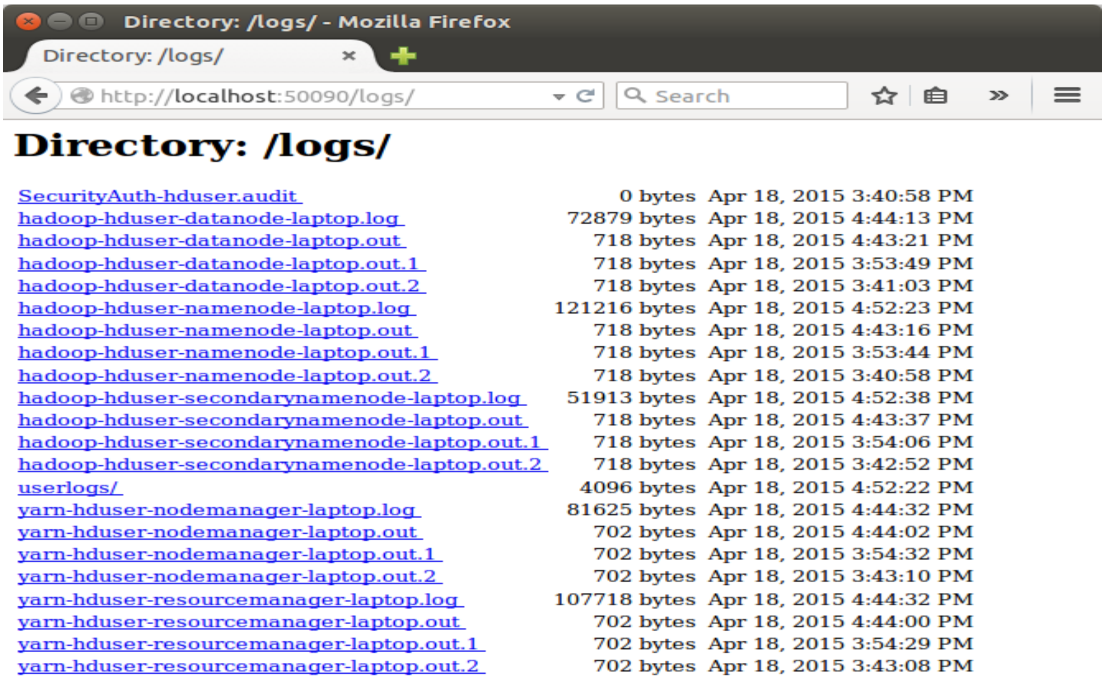
Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
laptop (127.0.0.1:50010)	1	In Service	454.29 GB	28 KB	125.83 GB	328.47 GB	0	28 KB (0%)	0	2.6.0

Decomissioning

Node	Last contact	Under replicated blocks	Blocks with no live replicas	Under Replicated Blocks In files under construction

Hadoop, 2014.

Legacy



Directory: /logs/ - Mozilla Firefox
Directory: /logs/
http://localhost:50090/logs/
Search

Directory: /logs/

```
SecurityAuth-hduser.audit                          0 bytes  Apr 18, 2015 3:40:58 PM
hadoop-hduser-datanode-laptop.log              72879 bytes  Apr 18, 2015 4:44:13 PM
hadoop-hduser-datanode-laptop.out                718 bytes  Apr 18, 2015 4:43:21 PM
hadoop-hduser-datanode-laptop.out.1              718 bytes  Apr 18, 2015 3:53:49 PM
hadoop-hduser-datanode-laptop.out.2              718 bytes  Apr 18, 2015 3:41:03 PM
hadoop-hduser-namenode-laptop.log             121216 bytes  Apr 18, 2015 4:52:23 PM
hadoop-hduser-namenode-laptop.out                718 bytes  Apr 18, 2015 4:43:16 PM
hadoop-hduser-namenode-laptop.out.1              718 bytes  Apr 18, 2015 3:53:44 PM
hadoop-hduser-namenode-laptop.out.2              718 bytes  Apr 18, 2015 3:40:58 PM
hadoop-hduser-secondarynamenode-laptop.log     51913 bytes  Apr 18, 2015 4:52:38 PM
hadoop-hduser-secondarynamenode-laptop.out       718 bytes  Apr 18, 2015 4:43:37 PM
hadoop-hduser-secondarynamenode-laptop.out.1     718 bytes  Apr 18, 2015 3:54:06 PM
hadoop-hduser-secondarynamenode-laptop.out.2     718 bytes  Apr 18, 2015 3:42:52 PM
userlogs/                                       4096 bytes  Apr 18, 2015 4:52:22 PM
yarn-hduser-nodemanager-laptop.log             81625 bytes  Apr 18, 2015 4:44:32 PM
yarn-hduser-nodemanager-laptop.out               702 bytes  Apr 18, 2015 4:44:02 PM
yarn-hduser-nodemanager-laptop.out.1             702 bytes  Apr 18, 2015 3:54:32 PM
yarn-hduser-nodemanager-laptop.out.2             702 bytes  Apr 18, 2015 3:43:10 PM
yarn-hduser-resourcemanager-laptop.log        107718 bytes  Apr 18, 2015 4:44:32 PM
yarn-hduser-resourcemanager-laptop.out           702 bytes  Apr 18, 2015 4:44:00 PM
yarn-hduser-resourcemanager-laptop.out.1         702 bytes  Apr 18, 2015 3:54:29 PM
yarn-hduser-resourcemanager-laptop.out.2         702 bytes  Apr 18, 2015 3:43:08 PM
```

VIVA QUESTIONS

1. What is Hadoop?

Apache Hadoop is an open-source framework that is used to efficiently store and process large datasets ranging in size from gigabytes to petabytes of data

2. What are the three modes in which Hadoop can run?

- Standalone Mode.
- Pseudo-distributed Mode.
- Fully-Distributed Mode.

3. What are the different Hadoop Configuration files?

HADOOP-ENV.sh, CORE-SITE.XML, HDFS-SITE.XML, MAPRED-SITE.XML, Masters and Slave

4. Difference between regular file system and HDFS?

Normal file systems have small block size of data. (Around 512 bytes) while HDFS has larger block sizes at around 64 MB) Multiple disks seek for larger files in normal file systems while in HDFS, data is read sequentially after every individual seek

5. What are the two main purpose of Hadoop?

Storing and processing big data

WEEK2:

IMPLEMENTATION OF WORD COUNT / FREQUENCY PROGRAMS USING MAPREDUCE

PROCEDURE:

1. Install hadoop.
2. Start all services using the command.

```
hduser@prince-VirtualBox:/usr/local/hadoop/bin$ jps
3242 Jps
```

```
hduser@prince-VirtualBox:/usr/local/hadoop/bin$ start-all.sh
```

```
hduser@prince-VirtualBox:/usr/local/hadoop/bin$ jps
16098 NameNode
16214 DataNode
16761 NodeManager
16636 ResourceManager
16429 SecondaryNameNode
19231 Jps
```

PROGRAM CODING:

```
hduser@prince-VirtualBox:/usr/local/hadoop/bin$ nano wordcount7.java
```

```
import java.io.IOException;
import java.util.StringTokenizer;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class wordcount7 {

public static class TokenizerMapper extends Mapper<Object, Text, Text, IntWritable>
{

private final static IntWritable one = new IntWritable (1);
private Text word = new Text ();

public void map (Object key, Text value, Context context) throws IOException,
InterruptedException {
StringTokenizeritr = new StringTokenizer(value.toString());
while (itr.hasMoreTokens ()) {
word.set (itr.nextToken ());
context.write (word, one);
}
}
}
public static class IntSumReducer
extends Reducer<Text,IntWritable,Text,IntWritable> {
privateIntWritable result = new IntWritable();

public void reduce(Text key, Iterable<IntWritable> values,
Context context
) throws IOException, InterruptedException {
int sum = 0;
for (IntWritableval : values) {
sum += val.get();
}
result.set(sum);
context.write(key, result);
}
}

public static void main(String[] args) throws Exception {
Configuration conf = new Configuration();
Job job = Job.getInstance (conf, "word count");
job.setJarByClass(wordcount7.class);
job.setMapperClass(TokenizerMapper.class);
job.setCombinerClass(IntSumReducer.class);
job.setReducerClass(IntSumReducer.class);
job.setOutputKeyClass (Text.class);
job.setOutputValueClass(IntWritable.class);
FileInputFormat.addInputPath(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));
System.exit (job.waitForCompletion (true) ? 0 : 1);
}
}
```

TO COMPILE:

```
hduser@prince-VirtualBox:/usr/local/hadoop/bin$ hadoopcom.sun.tools.javac.Main wordcount7.java
```

TO CREATE A JAR FILE:

```
hduser@prince-VirtualBox:/usr/local/hadoop/bin$ jar cf wc2.jar wordcount7*.java
```

TO CREATE A DIRECTORY IN HDFS:

```
hduser@prince-VirtualBox:/usr/local/hadoop/bin$hadoopdfs -mkdir /deepika
```

TO LOAD INPUT FILE:

```
hduser@prince-VirtualBox:/usr/local/hadoop/bin$hdfs -put
/home/prince/Downloads/wc.txt /deepika/wc1
```

TO EXECUTE:

```
hduser@prince-VirtualBox:/usr/local/hadoop/bin$ hadoop jar wc2.jar wordcount7
/deepika/wc1.txt /deepika/out2
```

```
16/09/16 14:34:23 INFO mapreduce.Job: map 0% reduce 0%
.
.
.
16/09/16 14:34:26 INFO mapreduce.Job: map 100% reduce 0%
.
.
.
16/09/16 14:34:30 INFO mapreduce.Job: map 100% reduce 100%
.
.
.
16/09/16 14:34:31 INFO mapreduce.Job: Job job_local364071501_0001 completed
successfully
16/09/16 14:34:31 INFO mapreduce.Job: Counters: 38
File System Counters
FILE: Number of bytes read=8552
FILE: Number of bytes written=507858
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=1424
HDFS: Number of bytes written=724
HDFS: Number of read operations=13
HDFS: Number of large read operations=0
HDFS: Number of write operations=4
Map-Reduce Framework
Map input records=10
Map output records=92
Map output bytes=1079
Map output materialized bytes=1018
Input split bytes=99
Combine input records=92
Combine output records=72
Reduce input groups=72
Reduce shuffle bytes=1018
Reduce input records=72
Reduce output records=72
Spilled Records=144
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=111
```

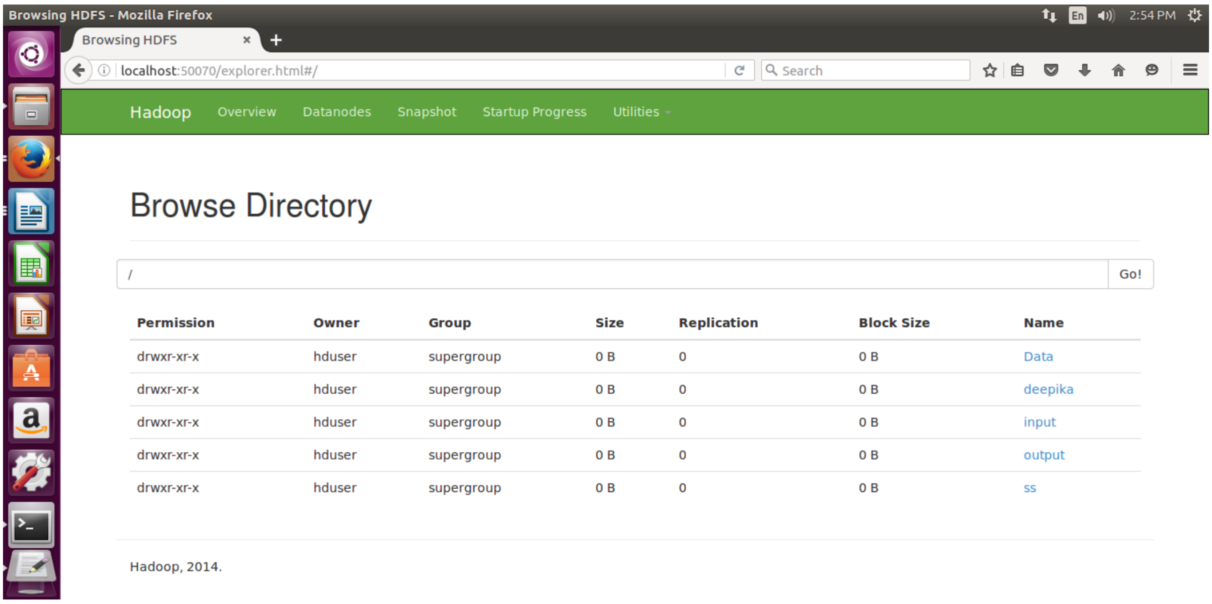
```
CPU time spent (ms)=0
Physical memory (bytes) snapshot=0
Virtual memory (bytes) snapshot=0
Total committed heap usage (bytes)=242360320
Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Input Format Counters
Bytes Read=712
File Output Format Counters
Bytes Written=724
```

INPUT FILE:
wc1.txt

STEPS:

1. Open an editor and type WordCount program and save as WordCount.java
2. Set the path as export HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar
3. To compile the program, bin/hadoopcom.sun.tools.javac.Main WordCount.java
4. Create a jar file, jar cf wc.jar WordCount*.class
5. Create input files input.txt,input1.txt and input2.txt and create a directory in hdfs, /mit/wordcount/input
6. Move these i/p files to hdfs system, bin/hadoopfs –put /opt/hadoop-2.7.0/input.txt /mit/wordcount/input/input.txt repeat this step for other two i/p files.
7. To execute, bin/hadoop jar wc.jar WordCount /mit/wordcount/input /mit/wordcount/output.
8. The mapreduce result will be available in the output directory.

OUTPUT:

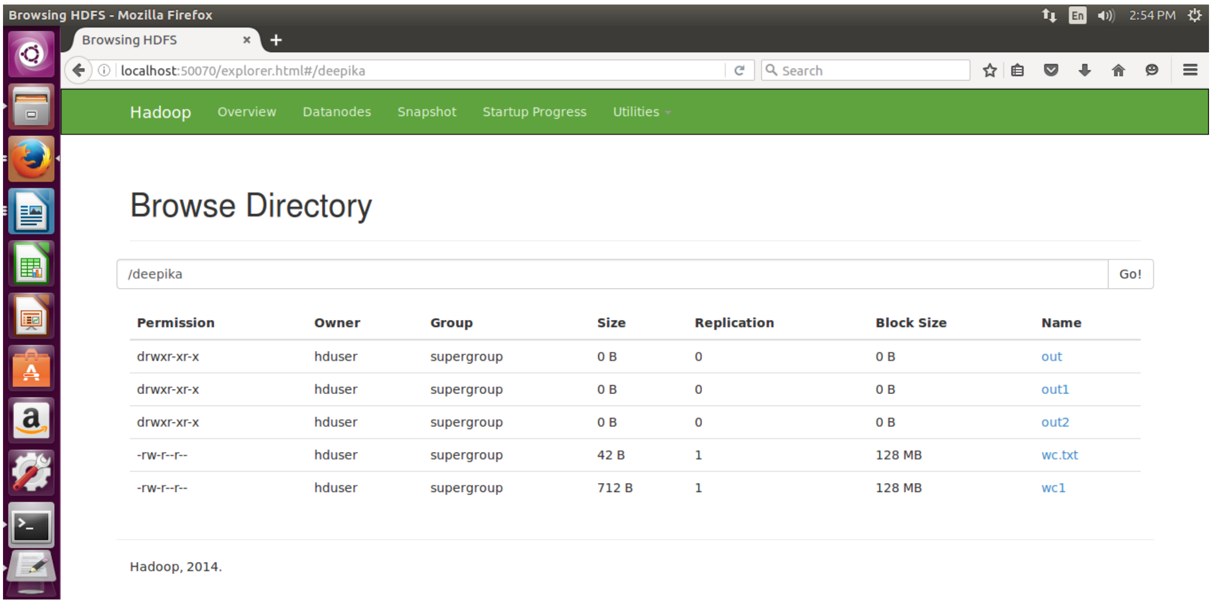


Browse Directory

/

Permission	Owner	Group	Size	Replication	Block Size	Name
drwxr-xr-x	hduser	supergroup	0 B	0	0 B	Data
drwxr-xr-x	hduser	supergroup	0 B	0	0 B	deepika
drwxr-xr-x	hduser	supergroup	0 B	0	0 B	input
drwxr-xr-x	hduser	supergroup	0 B	0	0 B	output
drwxr-xr-x	hduser	supergroup	0 B	0	0 B	ss

Hadoop, 2014.

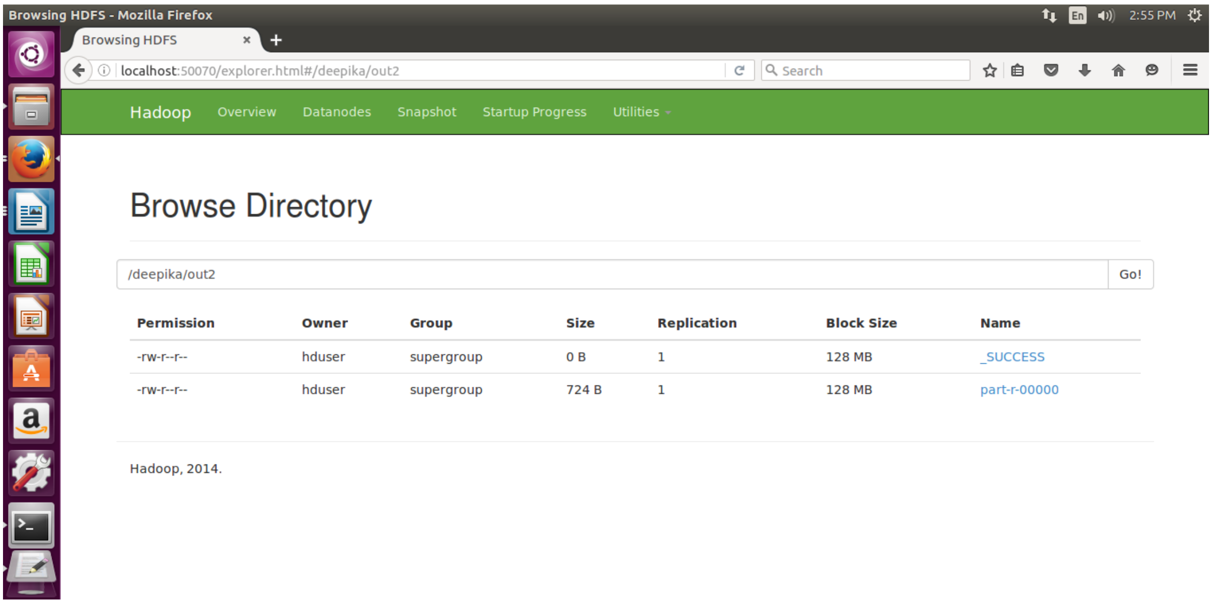


Browse Directory

/deepika

Permission	Owner	Group	Size	Replication	Block Size	Name
drwxr-xr-x	hduser	supergroup	0 B	0	0 B	out
drwxr-xr-x	hduser	supergroup	0 B	0	0 B	out1
drwxr-xr-x	hduser	supergroup	0 B	0	0 B	out2
-rw-r--r--	hduser	supergroup	42 B	1	128 MB	wc.txt
-rw-r--r--	hduser	supergroup	712 B	1	128 MB	wc1

Hadoop, 2014.

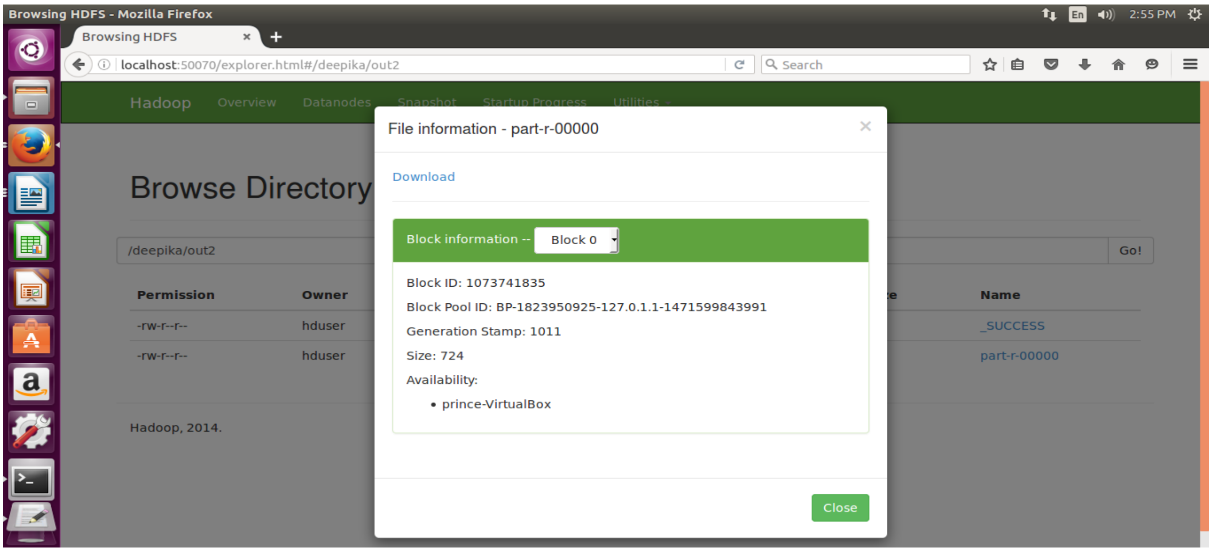


Browse Directory

/deepika/out2

Permission	Owner	Group	Size	Replication	Block Size	Name
-rw-r--r--	hduser	supergroup	0 B	1	128 MB	_SUCCESS
-rw-r--r--	hduser	supergroup	724 B	1	128 MB	part-r-00000

Hadoop, 2014.



File information - part-r-00000

Block ID: 1073741835
Block Pool ID: BP-1823950925-127.0.1.1-1471599843991
Generation Stamp: 1011
Size: 724
Availability:

- prince-VirtualBox

/mit/wordcount/input 2
/mit/wordcount/input/input.txt 1
/mit/wordcount/output. 1
/opt/hadoop-2.7.0/input.txt 1
1. 1
2. 1
3. 1
4. 1
5. 1
6. 1
7. 1
8. 1

Create 2

```
HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar 1
Move 1
Open 1
STEPS: 1
Set 1
The 1
To 2
WordCount 2
WordCount*.class 1
WordCount.java 2
a 2
an 1
and 4
as 2
available 1
be 1
bin/hadoop 3
cf 1
com.sun.tools.javac.Main 1
compile 1
create 1
directory 1
directory. 1
editor 1
execute, 1
export 1
file, 1
files 2
files. 1
for 1
fs 1
hdfs 1
hdfs, 1
i/p 2
in 2
input 1
input.txt,input1.txt 1
input2.txt 1
jar 3
mapreduce 1
other 1
output 1
path 1
program 1
program, 1
repeat 1
result 1
save 1
step 1
system, 1
```

```
the 3
these 1
this 1
to 1
two 1
type 1
wc.jar 2
will 1
–put 1
```

Viva Questions

1. What is Map Reduce?

MapReduce implements various mathematical algorithms to divide a task into small parts and assign them to multiple systems. In technical terms, MapReduce algorithm helps in sending the Map & Reduce tasks to appropriate servers in a cluster.

2. Mention three advantages of Map Reduce?

Parallel processing, Scalability, Fast

3. What are the main components of Map Reduce?

The two main components of the MapReduce Job are the JobTracker and TaskTracker.

4. What is HDFS?

HDFS is a distributed file system that handles large data sets running on commodity hardware. It is used to scale a single Apache Hadoop cluster to hundreds (and even thousands) of nodes. HDFS is one of the major components of Apache Hadoop, the others being MapReduce and YARN.

5. What is meant by Job Tracker?

The JobTracker is the service within Hadoop that farms out MapReduce tasks to specific nodes in the cluster, ideally the nodes that have the data, or at least are in the same rack. Client applications submit jobs to the Job tracker. The JobTracker talks to the NameNode to determine the location of the data.

WEEK 3

Implementation of MR program that processes a weather dataset

PROGRAM CODING:

```
import java.io.IOException;
import java.util.StringTokenizer;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.output.MultipleOutputs;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

/**
* @author devinline
*/
public class CalculateMaxAndMinTemeratureWithTime {
public static String calOutputName = "California";
public static String nyOutputName = "Newyork";
public static String njOutputName = "Newjersy";
public static String ausOutputName = "Austin";
public static String bosOutputName = "Boston";
public static String balOutputName = "Baltimore";

public static class WhetherForcastMapper extends
  Mapper<Object, Text, Text, Text> {

 public void map(Object keyOffset, Text dayReport, Context con)
   throws IOException, InterruptedException {
  StringTokenizer strTokens = new StringTokenizer(
    dayReport.toString(), "\t");
  int counter = 0;
  Float currnetTemp = null;
  Float minTemp = Float.MAX_VALUE;
  Float maxTemp = Float.MIN_VALUE;
  String date = null;
  String currentTime = null;
  String minTempANDTime = null;
  String maxTempANDTime = null;

  while (strTokens.hasMoreElements()) {
   if (counter == 0) {
    date = strTokens.nextToken();
   } else {
    if (counter % 2 == 1) {
     currentTime = strTokens.nextToken();
    } else {
     currnetTemp = Float.parseFloat(strTokens.nextToken());
     if (minTemp > currnetTemp) {
      minTemp = currnetTemp;
      minTempANDTime = minTemp + "AND" + currentTime;
     }
     if (maxTemp < currnetTemp) {
      maxTemp = currnetTemp;
      maxTempANDTime = maxTemp + "AND" + currentTime;
     }
    }
   }
   counter++;
  }
  // Write to context - MinTemp, MaxTemp and corresponding time
  Text temp = new Text();
  temp.set(maxTempANDTime);
  Text dateText = new Text();
  dateText.set(date);
  try {
   con.write(dateText, temp);
  } catch (Exception e) {
   e.printStackTrace();
  }

  temp.set(minTempANDTime);
  dateText.set(date);
  con.write(dateText, temp);

 }
}

public static class WhetherForcastReducer extends
  Reducer<Text, Text, Text, Text> {
 MultipleOutputs<Text, Text> mos;

 public void setup(Context context) {
  mos = new MultipleOutputs<Text, Text>(context);
 }

 public void reduce(Text key, Iterable<Text> values, Context context)
   throws IOException, InterruptedException {
  int counter = 0;
  String reducerInputStr[] = null;
  String f1Time = "";
  String f2Time = "";
  String f1 = "", f2 = "";
  Text result = new Text();
  for (Text value : values) {
   if (counter == 0) {
    reducerInputStr = value.toString().split("AND");
    f1 = reducerInputStr[0];
    f1Time = reducerInputStr[1];
   }

   else {
    reducerInputStr = value.toString().split("AND");
    f2 = reducerInputStr[0];
    f2Time = reducerInputStr[1];
   }

   counter = counter + 1;
  }
  if (Float.parseFloat(f1) > Float.parseFloat(f2)) {

   result = new Text("Time: " + f2Time + " MinTemp: " + f2 + "\t"
     + "Time: " + f1Time + " MaxTemp: " + f1);
  } else {

   result = new Text("Time: " + f1Time + " MinTemp: " + f1 + "\t"
     + "Time: " + f2Time + " MaxTemp: " + f2);
  }
  String fileName = "";
  if (key.toString().substring(0, 2).equals("CA")) {
   fileName = CalculateMaxAndMinTemeratureTime.calOutputName;
  } else if (key.toString().substring(0, 2).equals("NY")) {
   fileName = CalculateMaxAndMinTemeratureTime.nyOutputName;
  } else if (key.toString().substring(0, 2).equals("NJ")) {
   fileName = CalculateMaxAndMinTemeratureTime.njOutputName;
  } else if (key.toString().substring(0, 3).equals("AUS")) {
   fileName = CalculateMaxAndMinTemeratureTime.ausOutputName;
  } else if (key.toString().substring(0, 3).equals("BOS")) {
   fileName = CalculateMaxAndMinTemeratureTime.bosOutputName;
  } else if (key.toString().substring(0, 3).equals("BAL")) {
   fileName = CalculateMaxAndMinTemeratureTime.balOutputName;
  }
  String strArr[] = key.toString().split("_");
  key.set(strArr[1]); //Key is date value
  mos.write(fileName, key, result);
 }

 @Override
 public void cleanup(Context context) throws IOException,
   InterruptedException {
  mos.close();
 }
}

public static void main(String[] args) throws IOException,
  ClassNotFoundException, InterruptedException {
 Configuration conf = new Configuration();
 Job job = Job.getInstance(conf, "Wheather Statistics of USA");
 job.setJarByClass(CalculateMaxAndMinTemeratureWithTime.class);

 job.setMapperClass(WhetherForcastMapper.class);
 job.setReducerClass(WhetherForcastReducer.class);

 job.setMapOutputKeyClass(Text.class);
 job.setMapOutputValueClass(Text.class);

 job.setOutputKeyClass(Text.class);
 job.setOutputValueClass(Text.class);

 MultipleOutputs.addNamedOutput(job, calOutputName,
   TextOutputFormat.class, Text.class, Text.class);
 MultipleOutputs.addNamedOutput(job, nyOutputName,
   TextOutputFormat.class, Text.class, Text.class);
 MultipleOutputs.addNamedOutput(job, njOutputName,
   TextOutputFormat.class, Text.class, Text.class);
 MultipleOutputs.addNamedOutput(job, bosOutputName,
   TextOutputFormat.class, Text.class, Text.class);
 MultipleOutputs.addNamedOutput(job, ausOutputName,
   TextOutputFormat.class, Text.class, Text.class);
 MultipleOutputs.addNamedOutput(job, balOutputName,
   TextOutputFormat.class, Text.class, Text.class);
// FileInputFormat.addInputPath(job, new Path(args[0]));
 // FileOutputFormat.setOutputPath(job, new Path(args[1]));
 Path pathInput = new Path(
   "hdfs://192.168.213.133:54310/weatherInputData/input_temp.txt");
 Path pathOutputDir = new Path(
   "hdfs://192.168.213.133:54310/user/hduser1/testfs/output_mapred3");
 FileInputFormat.addInputPath(job, pathInput);
 FileOutputFormat.setOutputPath(job, pathOutputDir);
 try {
  System.exit(job.waitForCompletion(true) ? 0 : 1);
 } catch (Exception e) {
  // TODO Auto-generated catch block
  e.printStackTrace();
 }}}
```

OUTPUT:

whether output directory is in place on HDFS. Execute following command to verify the same.

```
hduser1@ubuntu:/usr/local/hadoop2.6.1/bin$ ./hadoop fs -ls
/user/hduser1/testfs/output_mapred3
```

```
Found 8 items
-rw-r--r--   3 zytham supergroup        438 2015-12-11 19:21
/user/hduser1/testfs/output_mapred3/Austin-r-00000
-rw-r--r--   3 zytham supergroup        219 2015-12-11 19:21
/user/hduser1/testfs/output_mapred3/Baltimore-r-00000
-rw-r--r--   3 zytham supergroup        219 2015-12-11 19:21
/user/hduser1/testfs/output_mapred3/Boston-r-00000
-rw-r--r--   3 zytham supergroup        511 2015-12-11 19:21
/user/hduser1/testfs/output_mapred3/California-r-00000
-rw-r--r--   3 zytham supergroup        146 2015-12-11 19:21
/user/hduser1/testfs/output_mapred3/Newjersy-r-00000
-rw-r--r--   3 zytham supergroup        219 2015-12-11 19:21
/user/hduser1/testfs/output_mapred3/Newyork-r-00000
-rw-r--r--   3 zytham supergroup          0 2015-12-11 19:21
/user/hduser1/testfs/output_mapred3/_SUCCESS
-rw-r--r--   3 zytham supergroup          0 2015-12-11 19:21
/user/hduser1/testfs/output_mapred3/part-r-00000
```

```
hduser1@ubuntu:/usr/local/hadoop2.6.1/bin$ ./hadoop fs -cat
/user/hduser1/testfs/output_mapred3/Austin-r-00000
25-Jan-2018 Time: 12:34:542 MinTemp: -22.3 Time: 05:12:345 MaxTemp: 35.7
26-Jan-2018 Time: 22:00:093 MinTemp: -27.0 Time: 05:12:345 MaxTemp: 55.7
27-Jan-2018 Time: 02:34:542 MinTemp: -22.3 Time: 05:12:345 MaxTemp: 55.7
29-Jan-2018 Time: 14:00:093 MinTemp: -17.0 Time: 02:34:542 MaxTemp: 62.9
30-Jan-2018 Time: 22:00:093 MinTemp: -27.0 Time: 05:12:345 MaxTemp: 49.2
31-Jan-2018 Time: 14:00:093 MinTemp: -17.0 Time: 03:12:187 MaxTemp: 56.0
```

VIVA QUESTIONS

1. What is Text Input format?

TextInputFormat is one of the file formats of Hadoop. As the name suggest, it is used to read lines of text files. Basically, it helps in generating key-value pairs from the text.

2. What is difference between HDFS block and InputSplit?

Block – HDFS Block is the physical representation of data in Hadoop. InputSplit – MapReduce InputSplit is the logical representation of data present in the block in Hadoop. It is basically used during data processing in MapReduce program or other processing techniques.

3. What is partitioning?

Hadoop Partitioning specifies that all the values for each key are grouped together. It also makes sure that all the values of a single key go to the same reducer. This allows even distribution of the map output over the reducer.

4. How to set which framework would use to run mapreduce program?

MapReduce is a programming model or pattern within the Hadoop framework that is used to access big data stored in the Hadoop File System (HDFS). It is a core component, integral to the functioning of the Hadoop framework.

5. What is shuffling in map reduce?

The process of transferring data from the mappers to reducers is shuffling. It is also the process by which the system performs the sort. Then it transfers the map output to the reducer as input.

WEEK4

IMPLEMENTATION OF LINEAR AND LOGISTIC REGRESSION

PROGRAM CODING :(LINEAR REGRESSION)

The aim of linear regression is to model a continuous variable Y as a mathematical function of one or more X variable(s), so that we can use this regression model to predict the Y when only the X is known. This mathematical equation can be generalized as follows:

$$Y = \beta 1 + \beta 2X + \epsilon$$

For this analysis, we will use the cars dataset that comes with R by default. cars is a standard built-in dataset, that makes it convenient to demonstrate linear regression in a simple and easy to understand fashion. You can access this dataset simply by typing in cars in your R console.

```
# display the first 6 observations

head(cars)
```

Scatter Plot:

```
# scatterplot

scatter.smooth(x=cars$speed, y=cars$dist, main="Dist ~ Speed")
```

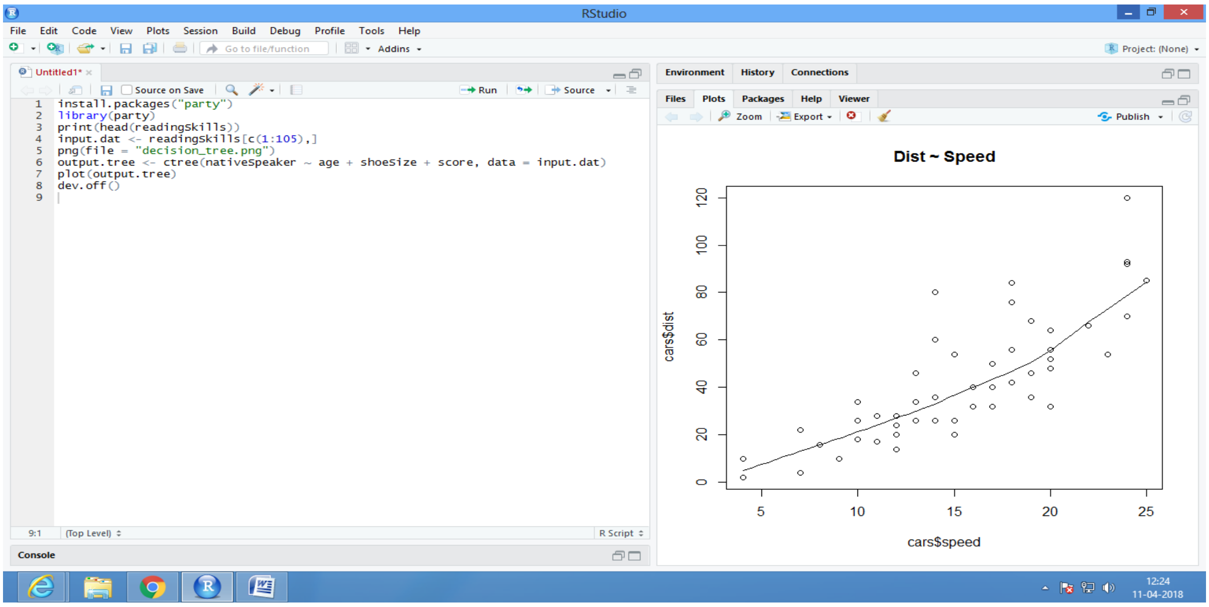
Build Linear Model:

```
# build linear regression model on full data
linearMod <- lm(dist ~ speed, data=cars)
print(linearMod)
```

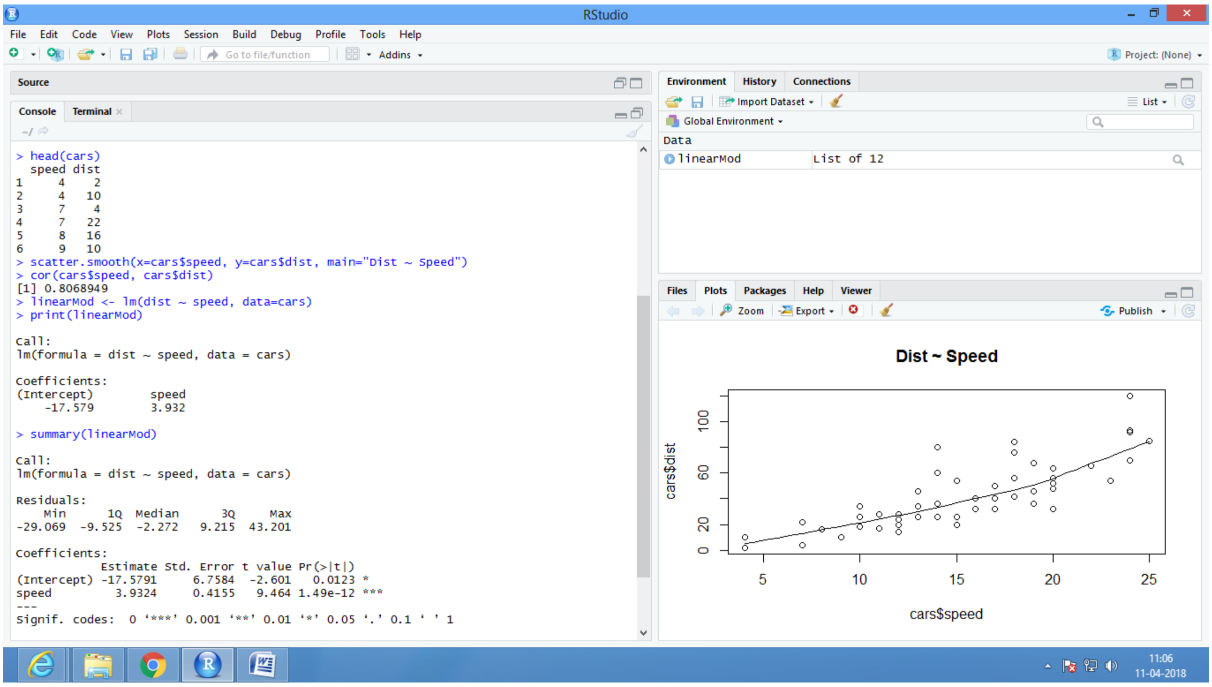
Linear Regression Diagnostics

```
summary(linearMod)
```

OUTPUT:



```
install.packages("party")
library(party)
print(head(readingSkills))
input.dat <- readingSkills[c(1:105),]
png(file = "decision_tree.png")
output.tree <- ctree(nativeSpeaker ~ age + shoeSize + score, data = input.dat)
plot(output.tree)
dev.off()
```



```
> head(cars)
  speed dist
1     4    2
2     4   10
3     7    4
4     7   22
5     8   16
6     9   10
> scatter.smooth(x=cars$speed, y=cars$dist, main="Dist ~ Speed")
> cor(cars$speed, cars$dist)
[1] 0.8068949
> linearMod <- lm(dist ~ speed, data=cars)
> print(linearMod)

Call:
lm(formula = dist ~ speed, data = cars)

Coefficients:
(Intercept)        speed
    -17.579        3.932

> summary(linearMod)

Call:
lm(formula = dist ~ speed, data = cars)

Residuals:
    Min      1Q  Median      3Q     Max
-29.069  -9.525  -2.272   9.215  43.201

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -17.5791     6.7584  -2.601   0.0123 *
speed         3.9324     0.4155   9.464 1.49e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Data
linearMod        List of 12
```

PROGRAM CODING:(LOGISTIC REGRESSION):

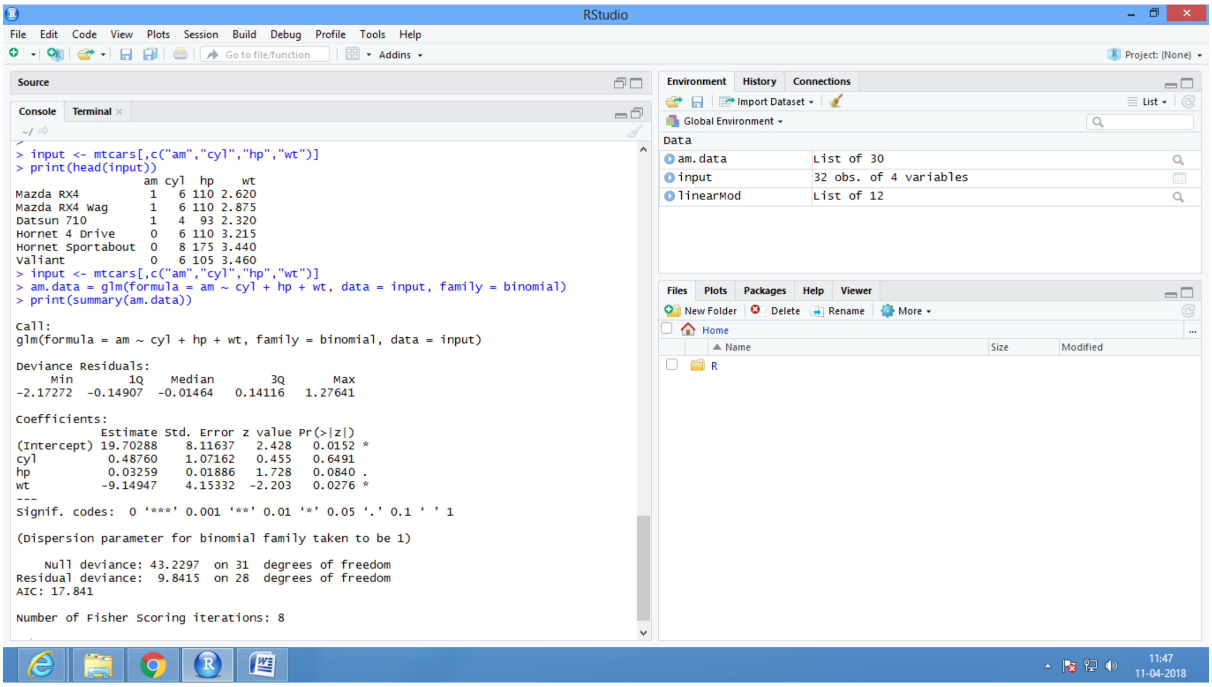
The general mathematical equation for logistic regression is
y = 1/(1+e^-(a+b1x1+b2x2+b3x3+...))

The basic syntax for glm() function in logistic regression is
glm(formula,data,family)

The in-built data set "mtcars" describes different models of a car with their various engine specifications. In "mtcars" data set, the transmission mode (automatic or manual) is described by the column am which is a binary value (0 or 1). We can create a logistic regression model between the columns "am" and 3 other columns - hp, wt and cyl.

# Select some columns form mtcars.
input <- mtcars[,c("am","cyl","hp","wt")]
print(head(input))
create regression model:
input <- mtcars[,c("am","cyl","hp","wt")]
am.data = glm(formula = am ~ cyl + hp + wt, data = input, family = binomial)
print(summary(am.data))

<u>OUTPUT:</u>



```
> 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)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.17272  -0.14907  -0.01464   0.14116   1.27641

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
```

```
Data
am.data     List of 30
input       32 obs. of 4 variables
linearMod   List of 12
```

VIVA QUESTIONS

1. **What is linear regression?**

Linear regression is a linear approach for modelling the relationship between a scalar response and one or more explanatory variables (also known as dependent and independent variables).

2. **What is logistic regression?**

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist.

3. **What are the assumptions of logistic regression?**

Basic assumptions that must be met for logistic regression include independence of errors, linearity in the logit for continuous variables, absence of multicollinearity, and lack of strongly influential outliers.

4. **Is logistic regression a type of supervised machine learning algorithm?**

Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable.

5. **Why logistic regression is called regression and not classification?**

Logistic regression uses the same basic formula as linear regression but it is regressing for the probability of a categorical outcome. Linear regression gives a continuous value of output y for a given input X. Whereas, logistic regression gives a continuous value of P(Y=1) for a given input X, which is later converted to Y=0 or Y=1 based on a threshold value. That’s the reason, logistic regression has “Regression” in its name.

WEEK5

IMPLEMENTATION OF SVM CLASSIFICATION TECHNIQUES

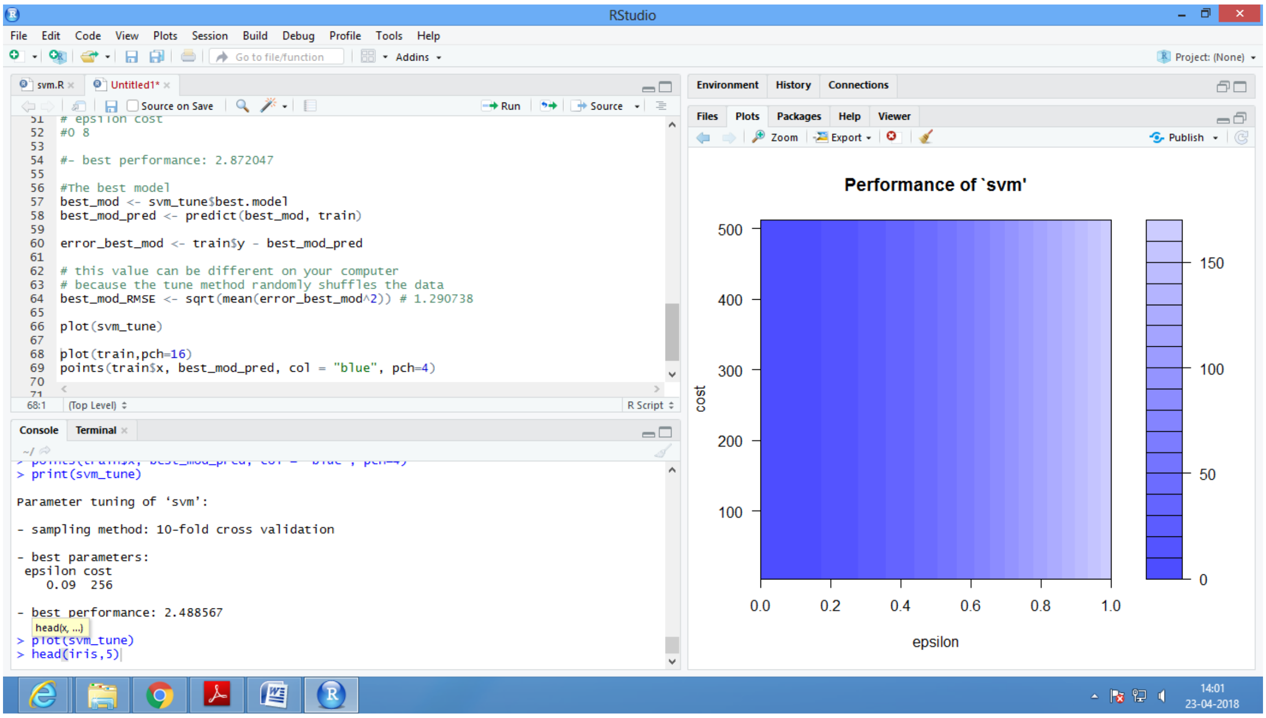
IMPLEMENTATION :(SVM)

To use SVM in R, we have a package e1071. The package is not preinstalled, hence one needs to run the line “install.packages(“e1071”) to install the package and then import the package contents using the library command--library(e1071).

R CODE:

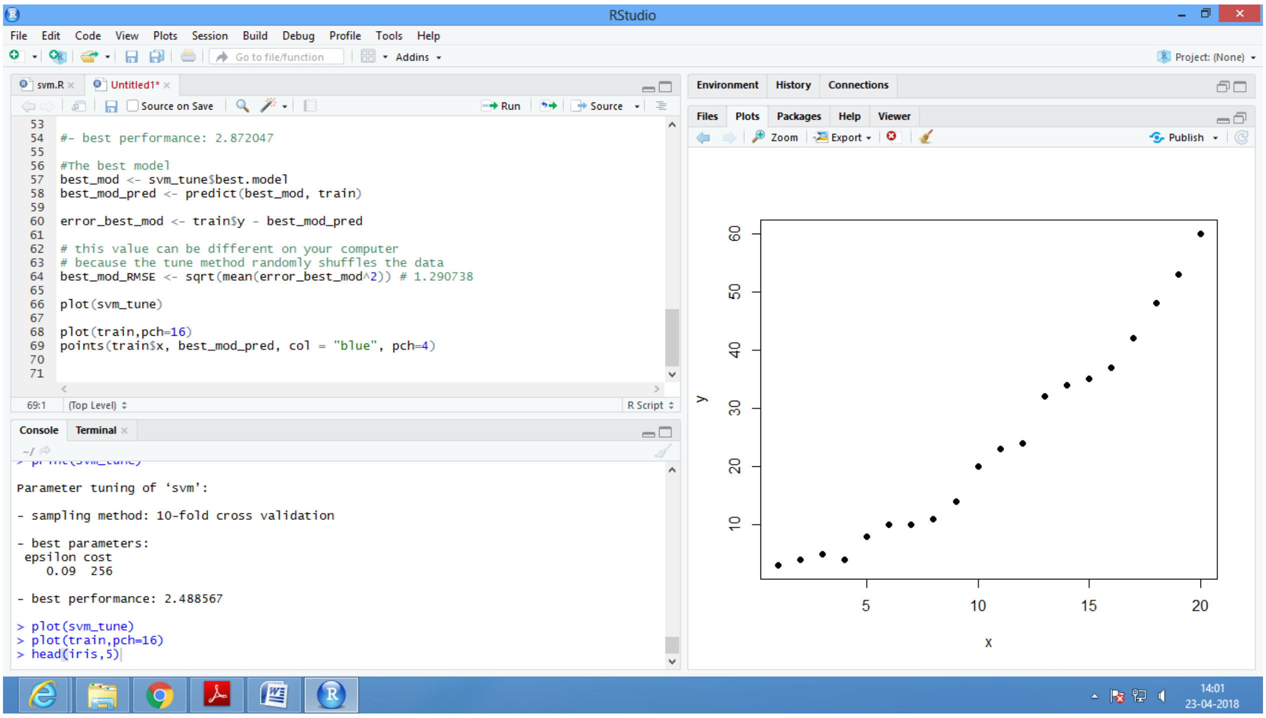
```
x=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)
y=c(3,4,5,4,8,10,10,11,14,20,23,24,32,34,35,37,42,48,53,60)

#Create a data frame of the data
train=data.frame(x,y)

#Plot the dataset
plot(train,pch=16)

#Linear regression
model <- lm(y ~ x, train)

#Plot the model using abline
abline(model)

#SVM
library(e1071)

#Fit a model. The function syntax is very similar to lm function
model_svm <- svm(y ~ x , train)

#Use the predictions on the data
pred <- predict(model_svm, train)

#Plot the predictions and the plot to see our model fit
points(train$x, pred, col = "blue", pch=4)

#Linear model has a residuals part which we can extract and directly calculate rmse
error <- model$residuals
lm_error <- sqrt(mean(error^2)) # 3.832974

#For svm, we have to manually calculate the difference between actual values (train$y) with
our predictions (pred)
error_2 <- train$y - pred
svm_error <- sqrt(mean(error_2^2)) # 2.696281


# perform a grid search
svm_tune <- tune(svm, y ~ x, data = train,
 ranges = list(epsilon = seq(0,1,0.01), cost = 2^(2:9))
)
print(svm_tune)

#Parameter tuning of ‘svm’:

# - sampling method: 10-fold cross validation

#- best parameters:
# epsilon cost
#0 8

#- best performance: 2.872047

#The best model
best_mod <- svm_tune$best.model
best_mod_pred <- predict(best_mod, train)

error_best_mod <- train$y - best_mod_pred

# this value can be different on your computer
# because the tune method randomly shuffles the data
best_mod_RMSE <- sqrt(mean(error_best_mod^2)) # 1.290738

plot(svm_tune)

plot(train,pch=16)
points(train$x, best_mod_pred, col = "blue", pch=4)
```

OUTPUT :



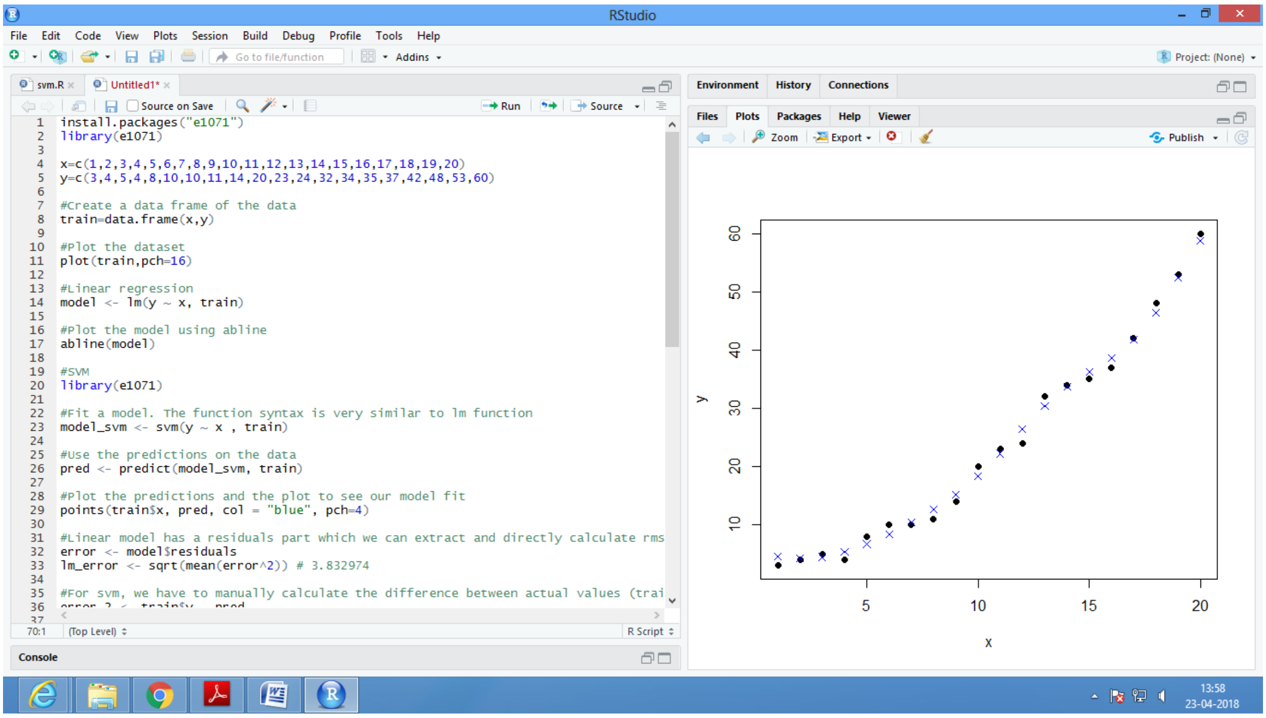
```
# epsilon cost
#0 8

#- best performance: 2.872047

#The best model
best_mod <- svm_tune$best.model
best_mod_pred <- predict(best_mod, train)

error_best_mod <- train$y - best_mod_pred

# this value can be different on your computer
# because the tune method randomly shuffles the data
best_mod_RMSE <- sqrt(mean(error_best_mod^2)) # 1.290738

plot(svm_tune)

plot(train,pch=16)
points(train$x, best_mod_pred, col = "blue", pch=4)
```

```
> points(train$x, best_mod_pred, col = "blue", pch=4)
> print(svm_tune)

Parameter tuning of ‘svm’:

- sampling method: 10-fold cross validation

- best parameters:
 epsilon cost
    0.09  256

- best performance: 2.488567
  head(x, ...)
> plot(svm_tune)
> head(iris,5)
```

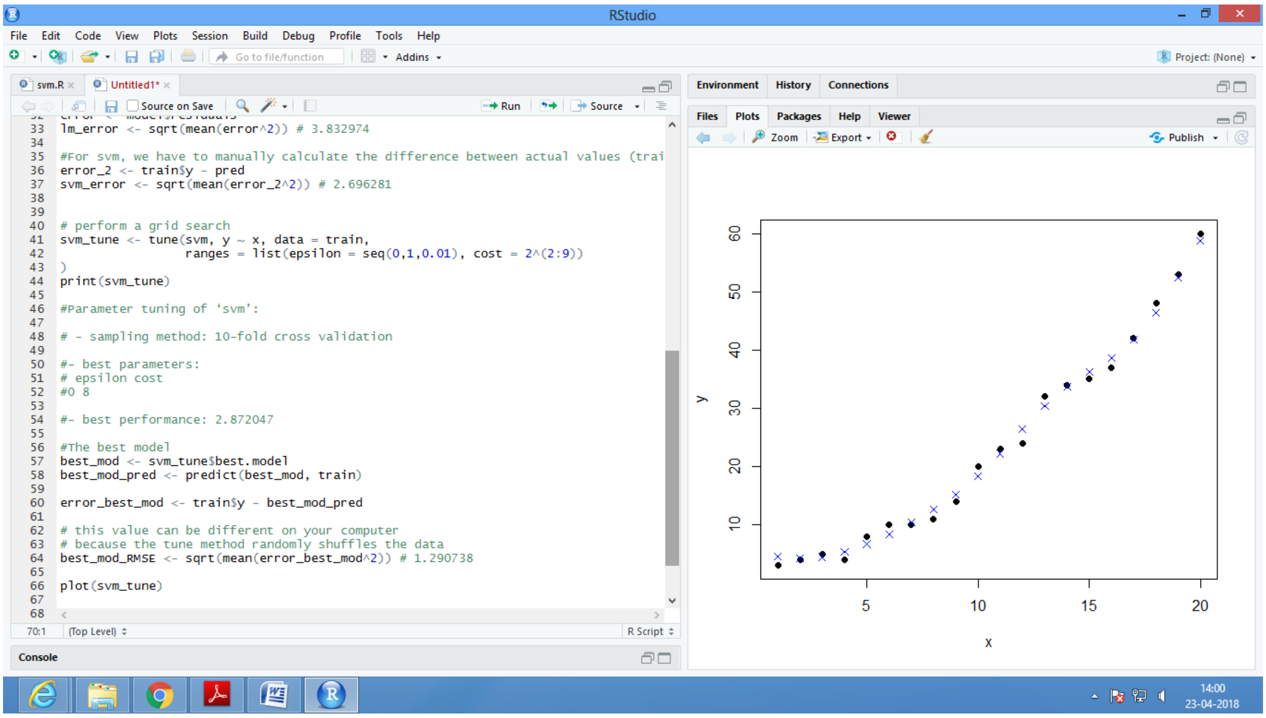


```

#- best performance: 2.872047

#The best model
best_mod <- svm_tune$best.model
best_mod_pred <- predict(best_mod, train)

error_best_mod <- train$y - best_mod_pred

# this value can be different on your computer
# because the tune method randomly shuffles the data
best_mod_RMSE <- sqrt(mean(error_best_mod^2)) # 1.290738

plot(svm_tune)

plot(train,pch=16)
points(train$x, best_mod_pred, col = "blue", pch=4)

```

```
> print(svm_tune)

Parameter tuning of ‘svm’:

- sampling method: 10-fold cross validation

- best parameters:
 epsilon cost
    0.09  256

- best performance: 2.488567

> plot(svm_tune)
> plot(train,pch=16)
> head(iris,5)
```



```r
install.packages("e1071")
library(e1071)

x=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)
y=c(3,4,5,4,8,10,10,11,14,20,23,24,32,34,35,37,42,48,53,60)

#Create a data frame of the data
train=data.frame(x,y)

#Plot the dataset
plot(train,pch=16)

#Linear regression
model <- lm(y ~ x, train)

#Plot the model using abline
abline(model)

#SVM
library(e1071)

#Fit a model. The function syntax is very similar to lm function
model_svm <- svm(y ~ x , train)

#Use the predictions on the data
pred <- predict(model_svm, train)

#Plot the predictions and the plot to see our model fit
points(train$x, pred, col = "blue", pch=4)

#Linear model has a residuals part which we can extract and directly calculate rms
error <- model$residuals
lm_error <- sqrt(mean(error^2)) # 3.832974

#For svm, we have to manually calculate the difference between actual values (trai
```



```r
lm_error <- sqrt(mean(error^2)) # 3.832974

#For svm, we have to manually calculate the difference between actual values (trai
error_2 <- train$y - pred
svm_error <- sqrt(mean(error_2^2)) # 2.696281


# perform a grid search
svm_tune <- tune(svm, y ~ x, data = train,
                 ranges = list(epsilon = seq(0,1,0.01), cost = 2^(2:9))
)
print(svm_tune)

#Parameter tuning of ‘svm’:

# - sampling method: 10-fold cross validation

#- best parameters:
# epsilon cost
#0 8

#- best performance: 2.872047

#The best model
best_mod <- svm_tune$best.model
best_mod_pred <- predict(best_mod, train)

error_best_mod <- train$y - best_mod_pred

# this value can be different on your computer
# because the tune method randomly shuffles the data
best_mod_RMSE <- sqrt(mean(error_best_mod^2)) # 1.290738

plot(svm_tune)
```

VIVA QUESTIONS

1. How SVM is used for classification?

SVM is a supervised machine learning algorithm which can be used for classification or regression problems. It uses a technique called the kernel trick to transform your data and then based on these transformations it finds an optimal boundary between the possible outputs.

2. Why is SVM used?

SVM can be used for classification (distinguishing between several groups or classes) and regression (obtaining a mathematical model to predict something). They can be applied to both linear and nonlinear problems.

3. What is the fundamental idea of SVM for classification?

SVM or Support Vector Machine is a linear model for classification and regression problems. It can solve linear and non-linear problems and work well for many practical problems.

4.Give some real-world applications of SVM?

Inverse Geosounding Problem, Texture Classification, Speech Recognition and more..

5. Explain about SVM regression?

Support Vector Regression is a supervised learning algorithm that is used to predict discrete values. Support Vector Regression uses the same principle as the SVMs. The basic idea behind SVR is to find the best fit line.

WEEK6

IMPLEMENTATION OF DECISION TREE CLASSIFICATION TECHNIQUES

Use the below command in R console to install the package. You also have to install the dependent packages if any.

```
install.packages("party")
```

The basic syntax for creating a decision tree in R is

```
ctree(formula, data)
```

input data:

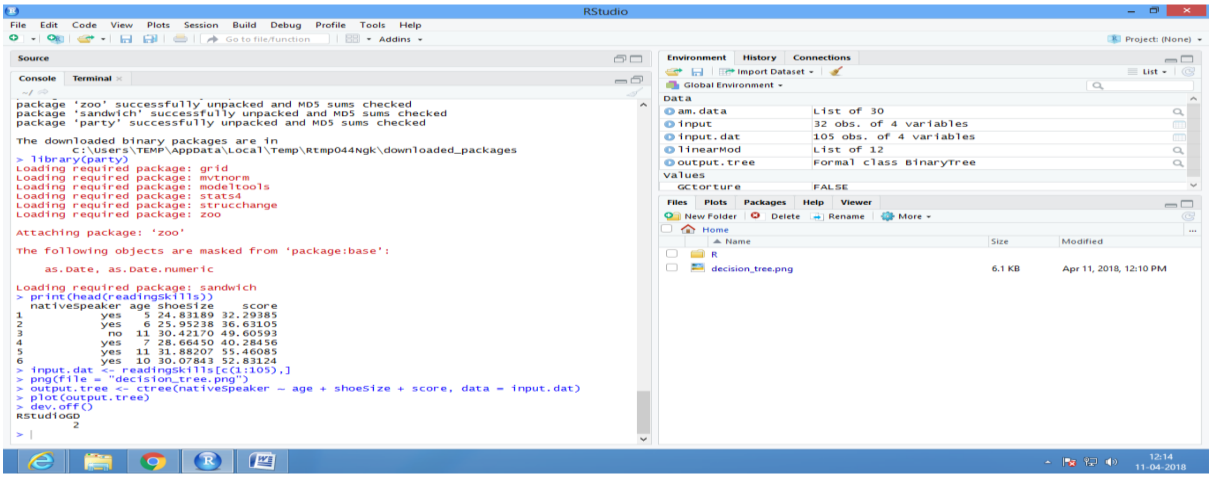
We will use the R in-built data set named readingSkills to create a decision tree. It describes the score of someone's readingSkills if we know the variables "age","shoesize","score" and whether the person is a native speaker or not.

```
# Load the party package. It will automatically load other dependent packages.
library(party)

# Print some records from data set readingSkills.
print(head(readingSkills))
# Load the party package. It will automatically load other dependent packages.
```

We will use the ctree() function to create the decision tree and see its graph.

```
# Load the party package. It will automatically load other dependent packages.
library(party)
# Create the input data frame.
input.dat <- readingSkills[c(1:105),]
# Give the chart file a name.
png(file = "decision_tree.png")
# Create the tree.
 output.tree <- ctree(nativeSpeaker ~ age + shoeSize + score, data = input.dat)
# Plot the tree.
plot(output.tree)
# Save the file.
dev.off()
```

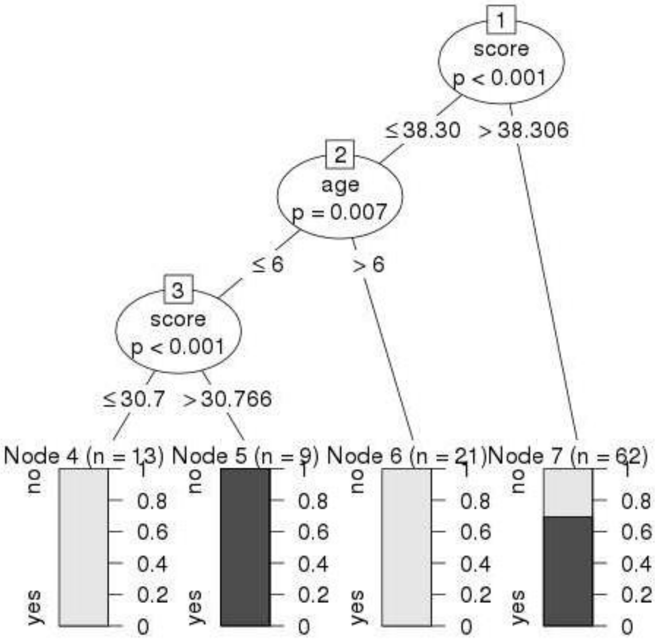
OUTPUT:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Source
Console
Terminal
~/
Environment History Connections
Import Dataset
List
Global Environment
Files Plots Packages Help Viewer
New Folder Delete Rename More
Home
12:14
11-04-2018

```
package ‘zoo’ successfully unpacked and MD5 sums checked
package ‘sandwich’ successfully unpacked and MD5 sums checked
package ‘party’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\TEMP\AppData\Local\Temp\Rtmp044Ngk\downloaded_packages
> library(party)
Loading required package: grid
Loading required package: mvtnorm
Loading required package: modeltools
Loading required package: stats4
Loading required package: strucchange
Loading required package: zoo

Attaching package: ‘zoo’

The following objects are masked from ‘package:base’:

    as.Date, as.Date.numeric

Loading required package: sandwich
> print(head(readingSkills))
  nativeSpeaker age shoeSize    score
1           yes   5 24.83189 32.29385
2           yes   6 25.95238 36.63105
3            no  11 30.42170 49.60593
4           yes   7 28.66450 40.28456
5           yes  11 31.88207 55.46085
6           yes  10 30.07843 52.83124
> input.dat <- readingSkills[c(1:105),]
> png(file = "decision_tree.png")
> output.tree <- ctree(nativeSpeaker ~ age + shoeSize + score, data = input.dat)
> plot(output.tree)
> dev.off()
RStudioGD
        2
> |
```

```
Data
am.data          List of 30
input            32 obs. of 4 variables
input.dat        105 obs. of 4 variables
linearMod        List of 12
output.tree      Formal class BinaryTree
Values
  GCtorture      FALSE
```

```
  Name                   Size     Modified
  R
  decision_tree.png      6.1 KB   Apr 11, 2018, 12:10 PM
```



VIVA QUESTIONS

1. Why are decisions trees the most popular classification technique?

Decision trees are able to generate understandable rules. Decision trees perform classification without requiring much computation. Decision trees are able to handle both continuous and categorical variables. Decision trees provide a clear indication of which fields are most important for prediction or classification.

2. Is decision tree a classification problem?

Decision trees are a rule-based approach to classification and regression problems. They use the values in each feature to split the dataset to a point where all data points that have the same class are grouped together.

3. Which algorithm is used in decision tree?

The basic algorithm used in decision trees is known as the ID3 (by Quinlan) algorithm. The ID3 algorithm builds decision trees using a top-down, greedy approach.

4. How decision trees used for classification?

Decision tree builds classification or regression models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed.

5. What is information gain?

Information gain is the reduction in entropy or surprise by transforming a dataset and is often used in training decision trees. Information gain is calculated by comparing the entropy of the dataset before and after a transformation.

WEEK7

IMPLEMENTATION OF HIERARCHICAL CLUSTERING

PROGRAM:

Installing and loading required R packages

install.packages("factoextra")

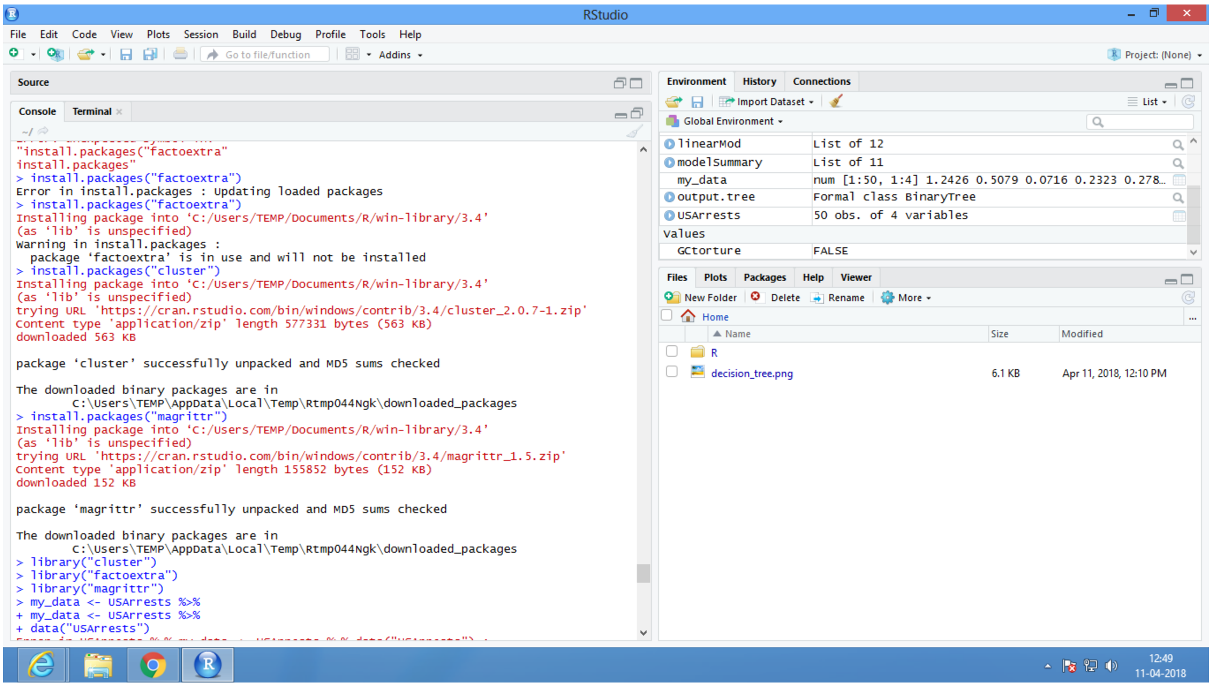
install.packages("cluster")

install.packages("magrittr")

library("cluster")

library("factoextra")

library("magrittr")


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Source
Console
Terminal
~/
Environment History Connections
Import Dataset
List
Global Environment
Files Plots Packages Help Viewer
New Folder Delete Rename More
Home
12:49
11-04-2018

```
"install.packages("factoextra"
install.packages"
> install.packages("factoextra")
Error in install.packages : Updating loaded packages
> install.packages("factoextra")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
Warning in install.packages :
  package ‘factoextra’ is in use and will not be installed
> install.packages("cluster")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/cluster_2.0.7-1.zip'
Content type 'application/zip' length 577331 bytes (563 KB)
downloaded 563 KB

package ‘cluster’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\TEMP\AppData\Local\Temp\Rtmp044Ngk\downloaded_packages
> install.packages("magrittr")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/magrittr_1.5.zip'
Content type 'application/zip' length 155852 bytes (152 KB)
downloaded 152 KB

package ‘magrittr’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\TEMP\AppData\Local\Temp\Rtmp044Ngk\downloaded_packages
> library("cluster")
> library("factoextra")
> library("magrittr")
> my_data <- USArrests %>%
+ my_data <- USArrests %>%
+ data("USArrests")
```

```
linearMod      List of 12
modelSummary   List of 11
my_data        num [1:50, 1:4] 1.2426 0.5079 0.0716 0.2323 0.278…
output.tree    Formal class BinaryTree
USArrests      50 obs. of 4 variables
Values
GCtorture      FALSE
```

	Name	Size	Modified
□	R		
□	decision_tree.png	6.1 KB	Apr 11, 2018, 12:10 PM

Data preparation

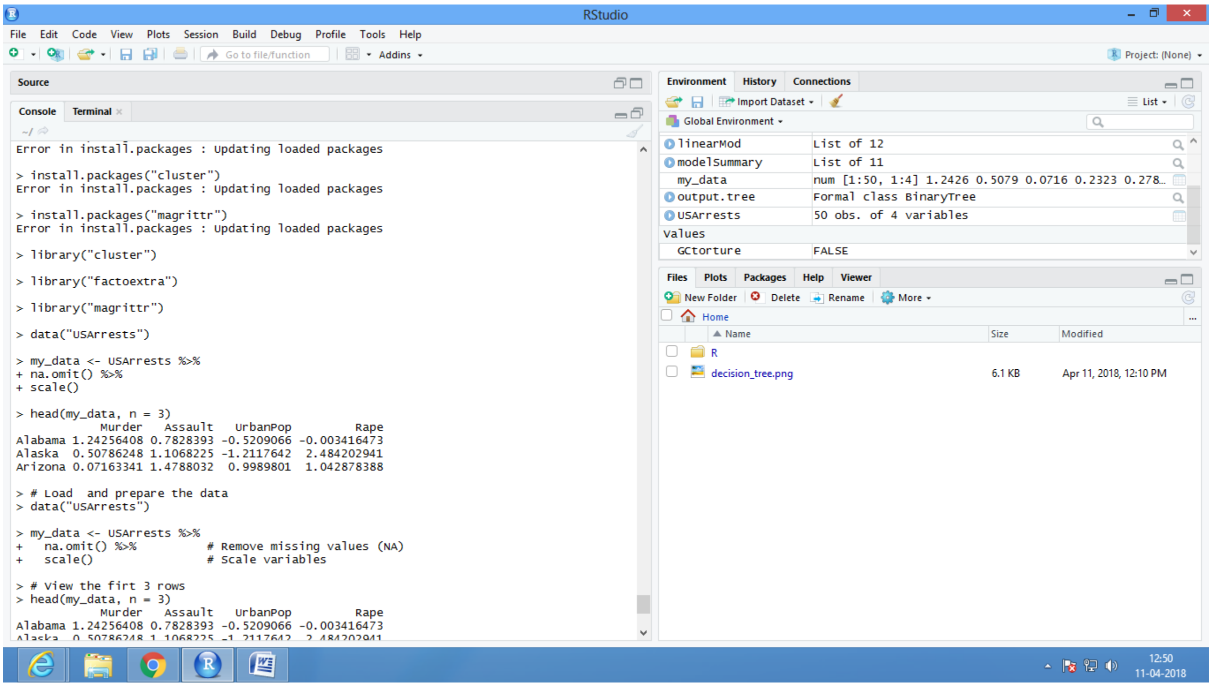
# Load and prepare the data

data("USArrests")

my_data <- **USArrests %>% na.omit() %>% # Remove missing values (NA) scale()** # Scale variables

View the firt 3 rows

head(my_data, n = 3)



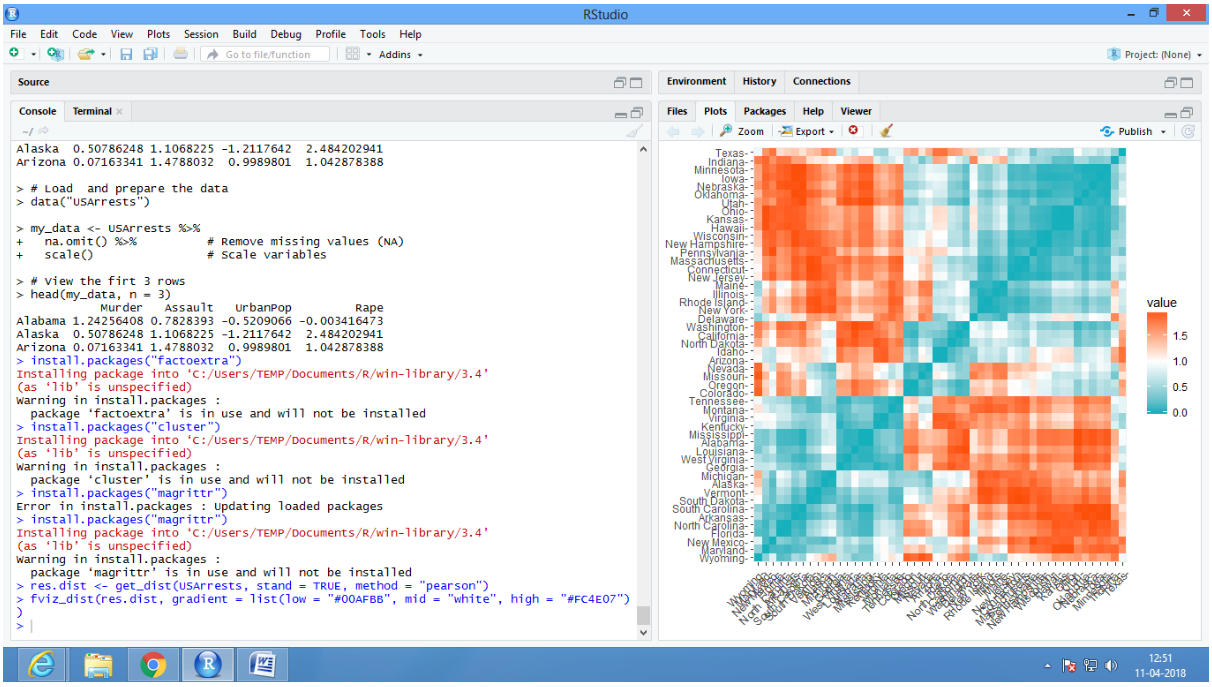
```
Error in install.packages : Updating loaded packages

> install.packages("cluster")
Error in install.packages : Updating loaded packages

> install.packages("magrittr")
Error in install.packages : Updating loaded packages

> library("cluster")

> library("factoextra")

> library("magrittr")

> data("USArrests")

> my_data <- USArrests %>%
+ na.omit() %>%
+ scale()

> head(my_data, n = 3)
            Murder   Assault   UrbanPop         Rape
Alabama 1.24256408 0.7828393 -0.5209066 -0.003416473
Alaska  0.50786248 1.1068225 -1.2117642  2.484202941
Arizona 0.07163341 1.4788032  0.9989801  1.042878388

> # Load  and prepare the data
> data("USArrests")

> my_data <- USArrests %>%
+   na.omit() %>%          # Remove missing values (NA)
+   scale()                # Scale variables

> # View the firt 3 rows
> head(my_data, n = 3)
            Murder   Assault   UrbanPop         Rape
Alabama 1.24256408 0.7828393 -0.5209066 -0.003416473
Alaska  0.50786248 1.1068225 -1.2117642  2.484202941
```

```
linearMod      List of 12
modelSummary   List of 11
my_data        num [1:50, 1:4] 1.2426 0.5079 0.0716 0.2323 0.278…
output.tree    Formal class BinaryTree
USArrests      50 obs. of 4 variables
Values
GCtorture      FALSE
```

	Name	Size	Modified
□	R		
□	decision_tree.png	6.1 KB	Apr 11, 2018, 12:10 PM

Distance measures

res.dist <- get_dist(USArrests, stand = TRUE, method = "pearson") fviz_dist(res.dist, gradient = list(low = "#00AFBB", mid = "white", high = "#FC4E07"))



```
Alaska  0.50786248 1.1068225 -1.2117642  2.484202941
Arizona 0.07163341 1.4788032  0.9989801  1.042878388

> # Load  and prepare the data
> data("USArrests")

> my_data <- USArrests %>%
+   na.omit() %>%          # Remove missing values (NA)
+   scale()                # Scale variables

> # View the firt 3 rows
> head(my_data, n = 3)
            Murder   Assault   UrbanPop         Rape
Alabama 1.24256408 0.7828393 -0.5209066 -0.003416473
Alaska  0.50786248 1.1068225 -1.2117642  2.484202941
Arizona 0.07163341 1.4788032  0.9989801  1.042878388
> install.packages("factoextra")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
Warning in install.packages :
  package ‘factoextra’ is in use and will not be installed
> install.packages("cluster")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
Warning in install.packages :
  package ‘cluster’ is in use and will not be installed
> install.packages("magrittr")
Error in install.packages : Updating loaded packages
> install.packages("magrittr")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
Warning in install.packages :
  package ‘magrittr’ is in use and will not be installed
> res.dist <- get_dist(USArrests, stand = TRUE, method = "pearson")
> fviz_dist(res.dist, gradient = list(low = "#00AFBB", mid = "white", high = "#FC4E07")
)
> |
```

VIVA QUESTIONS

1. **What is Hierarchical Clustering algorithm?**
   HCA is an unsupervised clustering algorithm which involves creating clusters that have predominant ordering from top to bottom.

2. **What are the various types of Hierarchical clustering algorithm?**
   Divisive and Agglomerative.

3. **What do you mean by clustering?**
   Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups.

4. **What is the minimum number of variables/features required to perform clustering?**
   At least a single variable is required to perform clustering analysis. Clustering analysis with a single variable can be visualized with the help of a histogram.

5. **What is the strength and weaknesses of Hierarchical Clustering?**

   Strength: sums up the data, good for small data sets.

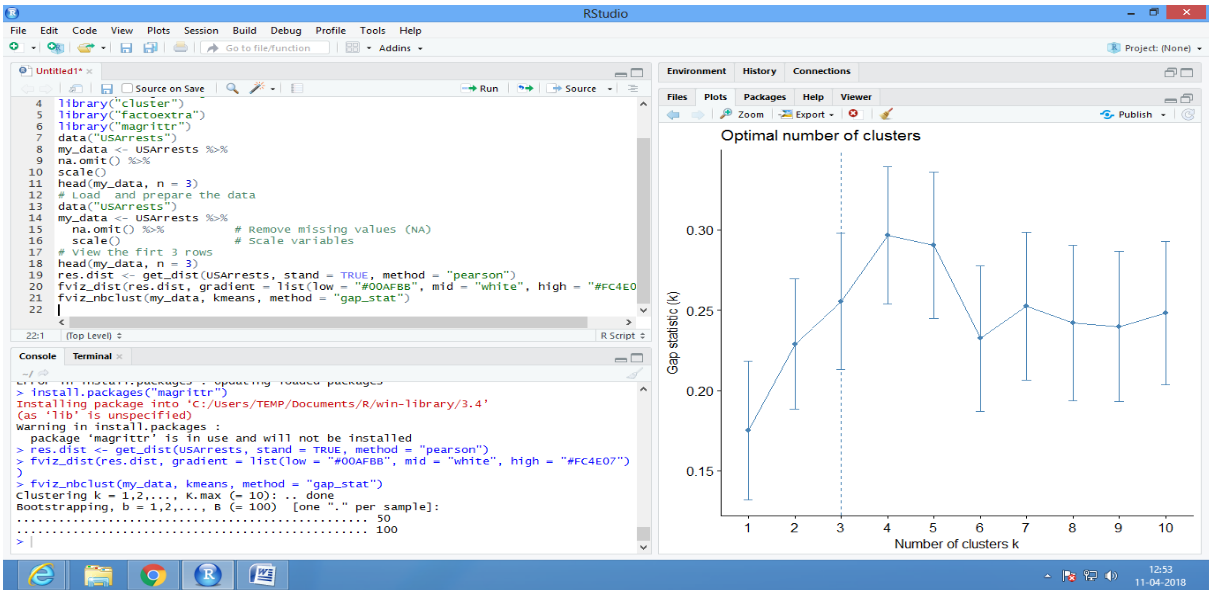
   Weakness: computationally demanding, fails on larger sets.

WEEK 8

IMPLEMENTATION OF PARTITIONING CLUSTERING

Determining the optimal number of clusters: use factoextra::fviz_nbclust()

library("factoextra")
fviz_nbclust(my_data, kmeans, method = "gap_stat")

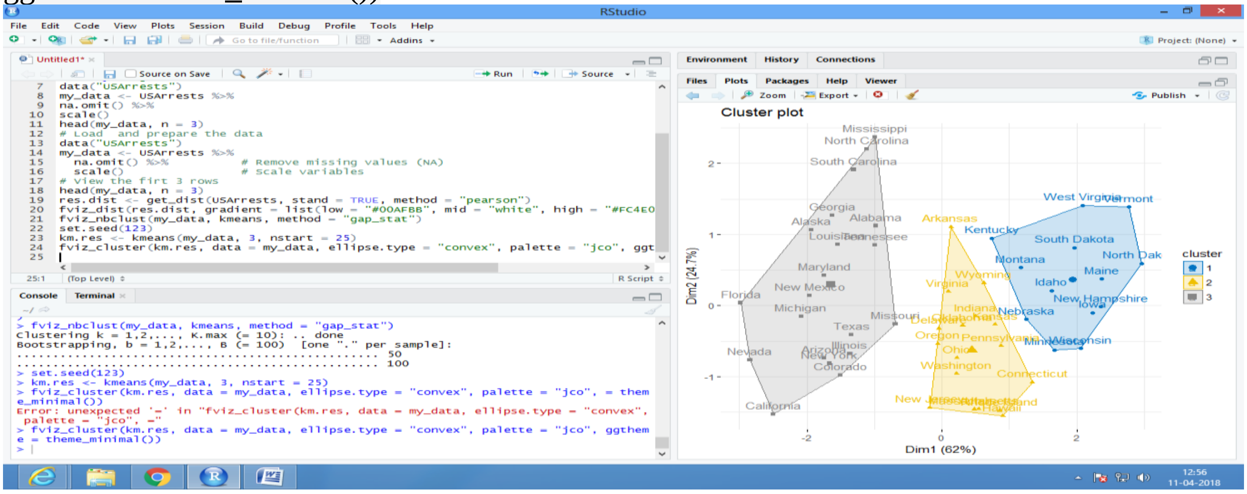


```
library("cluster")
library("factoextra")
library("magrittr")
data("USArrests")
my_data <- USArrests %>%
na.omit() %>%
scale()
head(my_data, n = 3)
# Load  and prepare the data
data("USArrests")
my_data <- USArrests %>%
  na.omit() %>%               # Remove missing values (NA)
  scale()                     # Scale variables
# View the firt 3 rows
head(my_data, n = 3)
res.dist <- get_dist(USArrests, stand = TRUE, method = "pearson")
fviz_dist(res.dist, gradient = list(low = "#00AFBB", mid = "white", high = "#FC4E0
fviz_nbclust(my_data, kmeans, method = "gap_stat")

```

```
Error in install.packages : updating loaded packages
> install.packages("magrittr")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
Warning in install.packages :
  package ‘magrittr’ is in use and will not be installed
> res.dist <- get_dist(USArrests, stand = TRUE, method = "pearson")
> fviz_dist(res.dist, gradient = list(low = "#00AFBB", mid = "white", high = "#FC4E07")
)
> fviz_nbclust(my_data, kmeans, method = "gap_stat")
Clustering k = 1,2,..., K.max (= 10): .. done
Bootstrapping, b = 1,2,..., B (= 100)  [one "." per sample]:
.................................................. 50
.................................................. 100
> 
```

Compute and visualize k-means clustering

set.seed(123)
km.res <- kmeans(my_data, 3, nstart = 25)
# Visualize
library("factoextra")
fviz_cluster(km.res, data = my_data,ellipse.type = "convex", palette = "jc,
ggtheme = theme_minimal())



```
data("USArrests")
my_data <- USArrests %>%
na.omit() %>%
scale()
head(my_data, n = 3)
# Load  and prepare the data
data("USArrests")
my_data <- USArrests %>%
  na.omit() %>%               # Remove missing values (NA)
  scale()                     # Scale variables
# View the firt 3 rows
head(my_data, n = 3)
res.dist <- get_dist(USArrests, stand = TRUE, method = "pearson")
fviz_dist(res.dist, gradient = list(low = "#00AFBB", mid = "white", high = "#FC4E0
fviz_nbclust(my_data, kmeans, method = "gap_stat")
set.seed(123)
km.res <- kmeans(my_data, 3, nstart = 25)
fviz_cluster(km.res, data = my_data, ellipse.type = "convex", palette = "jco", ggt

```

```
> fviz_nbclust(my_data, kmeans, method = "gap_stat")
Clustering k = 1,2,..., K.max (= 10): .. done
Bootstrapping, b = 1,2,..., B (= 100)  [one "." per sample]:
.................................................. 50
.................................................. 100
> set.seed(123)
> km.res <- kmeans(my_data, 3, nstart = 25)
> fviz_cluster(km.res, data = my_data, ellipse.type = "convex", palette = "jco", = them
e_minimal())
Error: unexpected '=' in "fviz_cluster(km.res, data = my_data, ellipse.type = "convex",
 palette = "jco", ="
> fviz_cluster(km.res, data = my_data, ellipse.type = "convex", palette = "jco", ggthem
e = theme_minimal())
> 
```

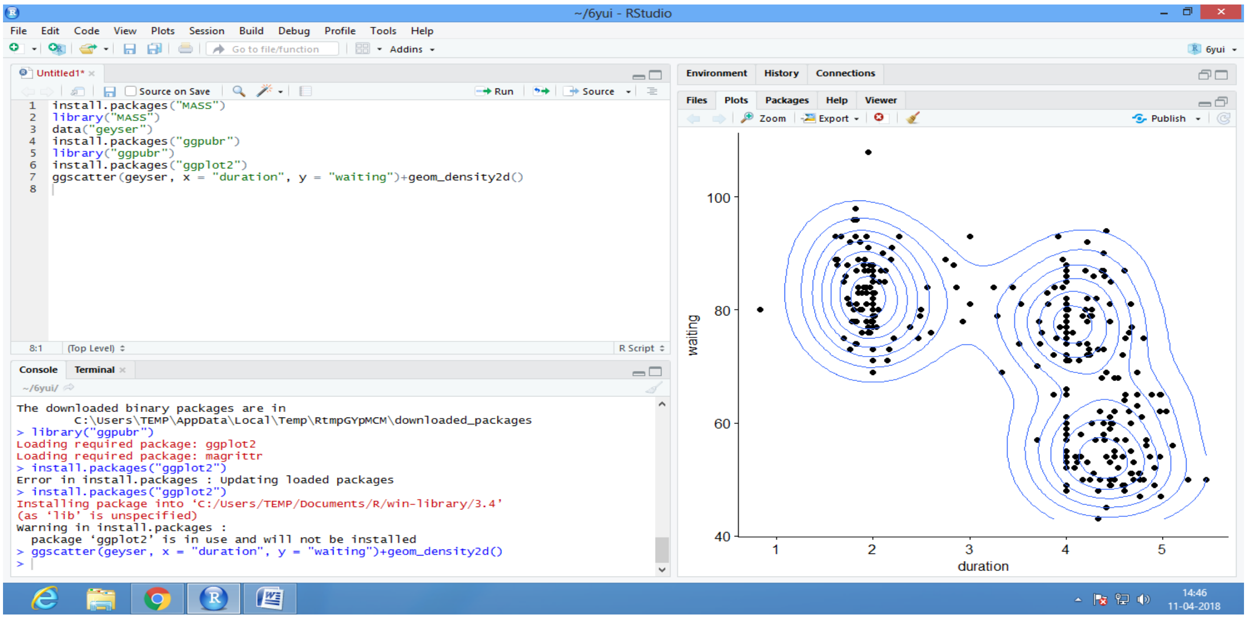
MODEL BASED CLUSTERING:

# Load the data

library("MASS") data("geyser")

# Scatter plot **library**("ggpubr")

ggscatter(geyser, x = "duration", y = "waiting")+ geom_density2d() # Add 2D density



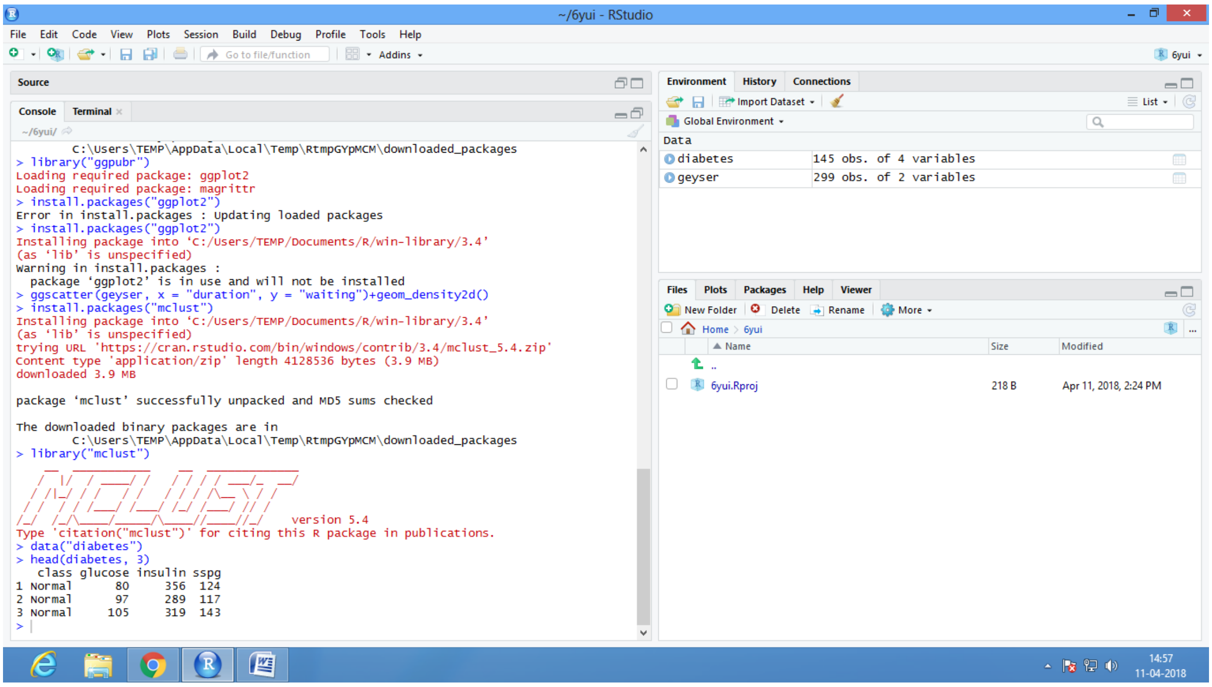
```
install.packages("MASS")
library("MASS")
data("geyser")
install.packages("ggpubr")
library("ggpubr")
install.packages("ggplot2")
ggscatter(geyser, x = "duration", y = "waiting")+geom_density2d()
```

```
The downloaded binary packages are in
        C:\Users\TEMP\AppData\Local\Temp\RtmpGYpMCM\downloaded_packages
> library("ggpubr")
Loading required package: ggplot2
Loading required package: magrittr
> install.packages("ggplot2")
Error in install.packages : Updating loaded packages
> install.packages("ggplot2")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
Warning in install.packages :
  package ‘ggplot2’ is in use and will not be installed
> ggscatter(geyser, x = "duration", y = "waiting")+geom_density2d()
>
```

library("mclust")

data("diabetes")

head(diabetes, 3)



~/6yui - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
6yui
Source
Console
Terminal
~/6yui/
Environment History Connections
Import Dataset
List
Global Environment
Files Plots Packages Help Viewer
New Folder Delete Rename More
Home > 6yui
14:57
11-04-2018

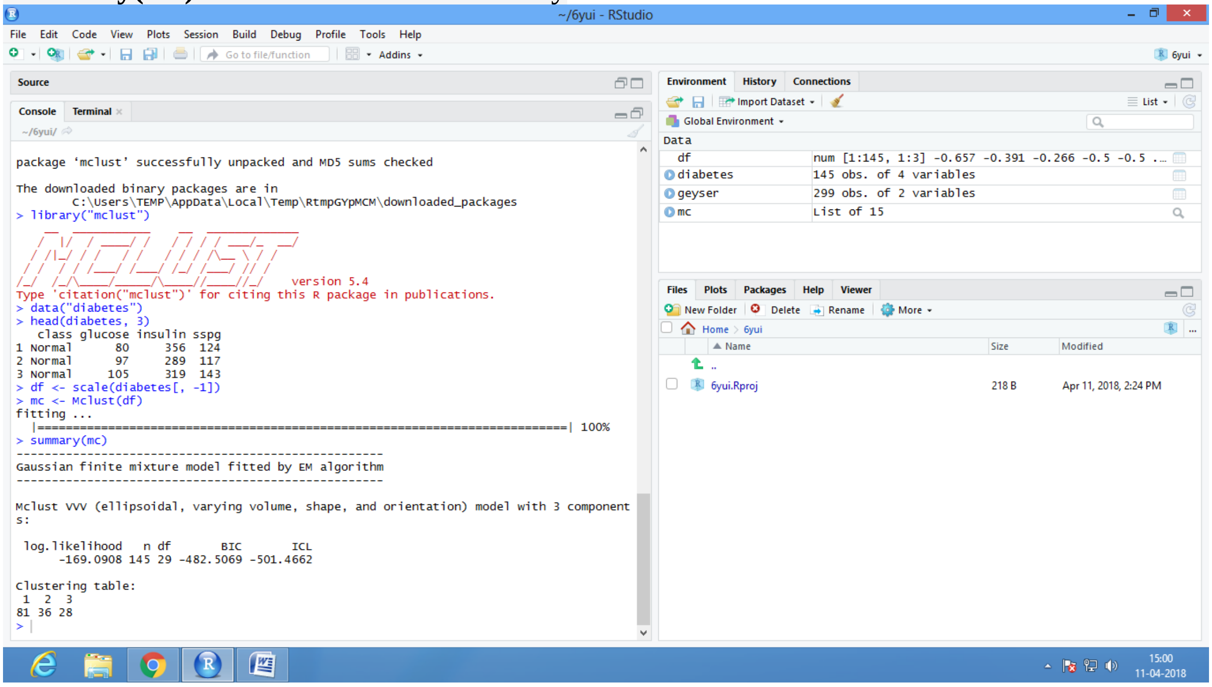
```
        C:\Users\TEMP\AppData\Local\Temp\RtmpGYpMCM\downloaded_packages
> library("ggpubr")
Loading required package: ggplot2
Loading required package: magrittr
> install.packages("ggplot2")
Error in install.packages : Updating loaded packages
> install.packages("ggplot2")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
Warning in install.packages :
  package ‘ggplot2’ is in use and will not be installed
> ggscatter(geyser, x = "duration", y = "waiting")+geom_density2d()
> install.packages("mclust")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/mclust_5.4.zip'
Content type 'application/zip' length 4128536 bytes (3.9 MB)
downloaded 3.9 MB

package ‘mclust’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\TEMP\AppData\Local\Temp\RtmpGYpMCM\downloaded_packages
> library("mclust")
    __  ___________    __  _____________
   /  |/  / ____/ /   / / / / ___/_  __/
  / /|_/ / /   / /   / / / /\__ \ / /
 / /  / / /___/ /___/ /_/ /___/ // /
/_/  /_/\____/_____/\____//____//_/    version 5.4
Type 'citation("mclust")' for citing this R package in publications.
> data("diabetes")
> head(diabetes, 3)
   class glucose insulin sspg
1 Normal      80     356  124
2 Normal      97     289  117
3 Normal     105     319  143
> 
```

```
Data
diabetes    145 obs. of 4 variables
geyser      299 obs. of 2 variables
```

```
Name          Size    Modified
..
6yui.Rproj    218 B   Apr 11, 2018, 2:24 PM
```

Model-based clustering can be computed using the function Mclust() as follow:

```
library(mclust)
df <- scale(diabetes[, -1]) # Standardize the data
mc <- Mclust(df)            # Model-based-clustering
summary(mc)                 # Print a summary
```



~/6yui - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
6yui
Source
Console
Terminal
~/6yui/
Environment History Connections
Import Dataset
List
Global Environment
Files Plots Packages Help Viewer
New Folder Delete Rename More
Home > 6yui
15:00
11-04-2018

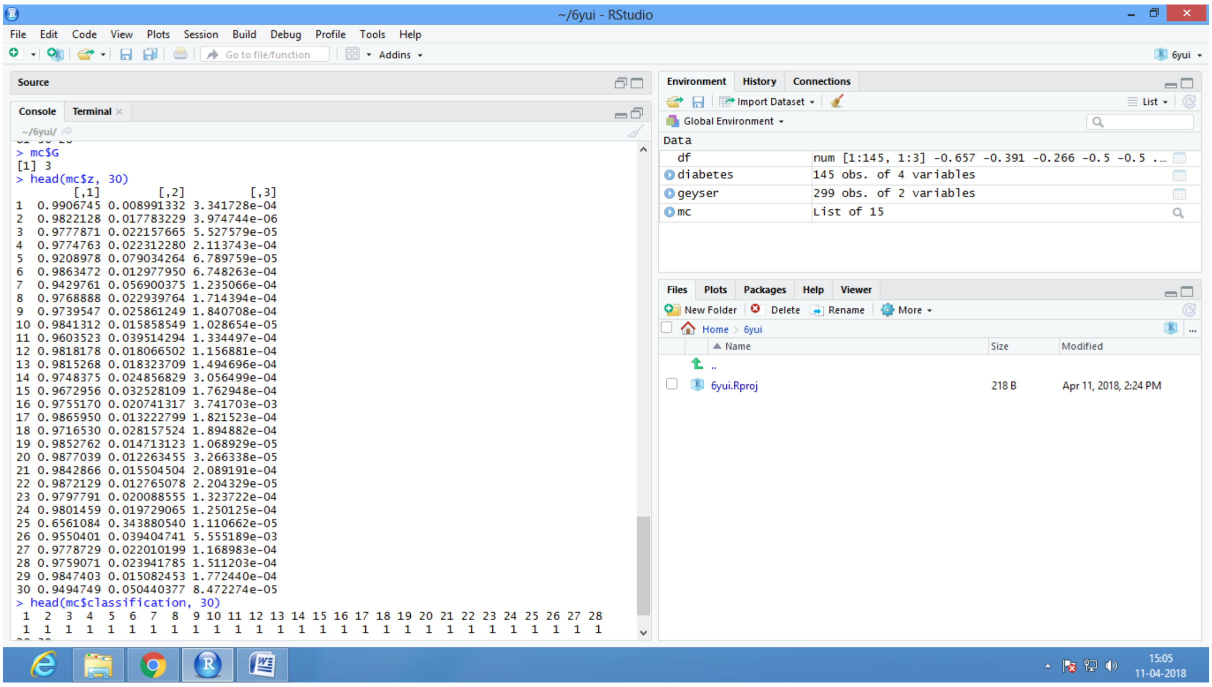
```
package ‘mclust’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\TEMP\AppData\Local\Temp\RtmpGYpMCM\downloaded_packages
> library("mclust")
    __  ___________    __  _____________
   /  |/  / ____/ /   / / / / ___/_  __/
  / /|_/ / /   / /   / / / /\__ \ / /
 / /  / / /___/ /___/ /_/ /___/ // /
/_/  /_/\____/_____/\____//____//_/    version 5.4
Type 'citation("mclust")' for citing this R package in publications.
> data("diabetes")
> head(diabetes, 3)
   class glucose insulin sspg
1 Normal      80     356  124
2 Normal      97     289  117
3 Normal     105     319  143
> df <- scale(diabetes[, -1])
> mc <- Mclust(df)
fitting ...
  |=====================================================================| 100%
> summary(mc)
----------------------------------------------------
Gaussian finite mixture model fitted by EM algorithm
----------------------------------------------------

Mclust VVV (ellipsoidal, varying volume, shape, and orientation) model with 3 component
s:

 log.likelihood   n df       BIC       ICL
      -169.0908 145 29 -482.5069 -501.4662

Clustering table:
 1  2  3
81 36 28
> 
```

```
Data
  df        num [1:145, 1:3] -0.657 -0.391 -0.266 -0.5 -0.5 ...
diabetes    145 obs. of 4 variables
geyser      299 obs. of 2 variables
mc          List of 15
```

```
Name          Size    Modified
..
6yui.Rproj    218 B   Apr 11, 2018, 2:24 PM
```

mc$modelName # Optimal selected model ==> "VVV"

mc$G # Optimal number of cluster => 3 **head**(mc$z, 30) # Probality to belong to a given cluster **head(mc$classification, 30)** # Cluster assignement of each observation



~/6yui - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
6yui
Source
Console
Terminal
~/6yui/
Environment History Connections
Import Dataset
List
Global Environment
Files Plots Packages Help Viewer
New Folder Delete Rename More
Home > 6yui
Name Size Modified

```
> mc$G
[1] 3
> head(mc$z, 30)
        [,1]        [,2]         [,3]
1  0.9906745 0.008991332 3.341728e-04
2  0.9822128 0.017783229 3.974744e-06
3  0.9777871 0.022157665 5.527579e-05
4  0.9774763 0.022312280 2.113743e-04
5  0.9208978 0.079034264 6.789759e-05
6  0.9863472 0.012977950 6.748263e-04
7  0.9429761 0.056900375 1.235066e-04
8  0.9768888 0.022939764 1.714394e-04
9  0.9739547 0.025861249 1.840708e-04
10 0.9841312 0.015858549 1.028654e-05
11 0.9603523 0.039514294 1.334497e-04
12 0.9818178 0.018066502 1.156881e-04
13 0.9815268 0.018323709 1.494696e-04
14 0.9748375 0.024856829 3.056499e-04
15 0.9672956 0.032528109 1.762948e-04
16 0.9755170 0.020741317 3.741703e-03
17 0.9865950 0.013222799 1.821523e-04
18 0.9716530 0.028157524 1.894882e-04
19 0.9852762 0.014713123 1.068929e-05
20 0.9877039 0.012263455 3.266338e-05
21 0.9842866 0.015504504 2.089191e-04
22 0.9872129 0.012765078 2.204329e-05
23 0.9797791 0.020088555 1.323722e-04
24 0.9801459 0.019729065 1.250125e-04
25 0.6561084 0.343880540 1.110662e-05
26 0.9550401 0.039404741 5.555189e-03
27 0.9778729 0.022010199 1.168983e-04
28 0.9759071 0.023941785 1.511203e-04
29 0.9847403 0.015082453 1.772440e-04
30 0.9494749 0.050440377 8.472274e-05
> head(mc$classification, 30)
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
 1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

```
Data
  df          num [1:145, 1:3] -0.657 -0.391 -0.266 -0.5 -0.5 ...
  diabetes    145 obs. of 4 variables
  geyser      299 obs. of 2 variables
  mc          List of 15
```

```
..
6yui.Rproj    218 B    Apr 11, 2018, 2:24 PM
```

VISUALIZING MODEL-BASED CLUSTERING

library(factoextra)

# BIC values used for choosing the number of clusters

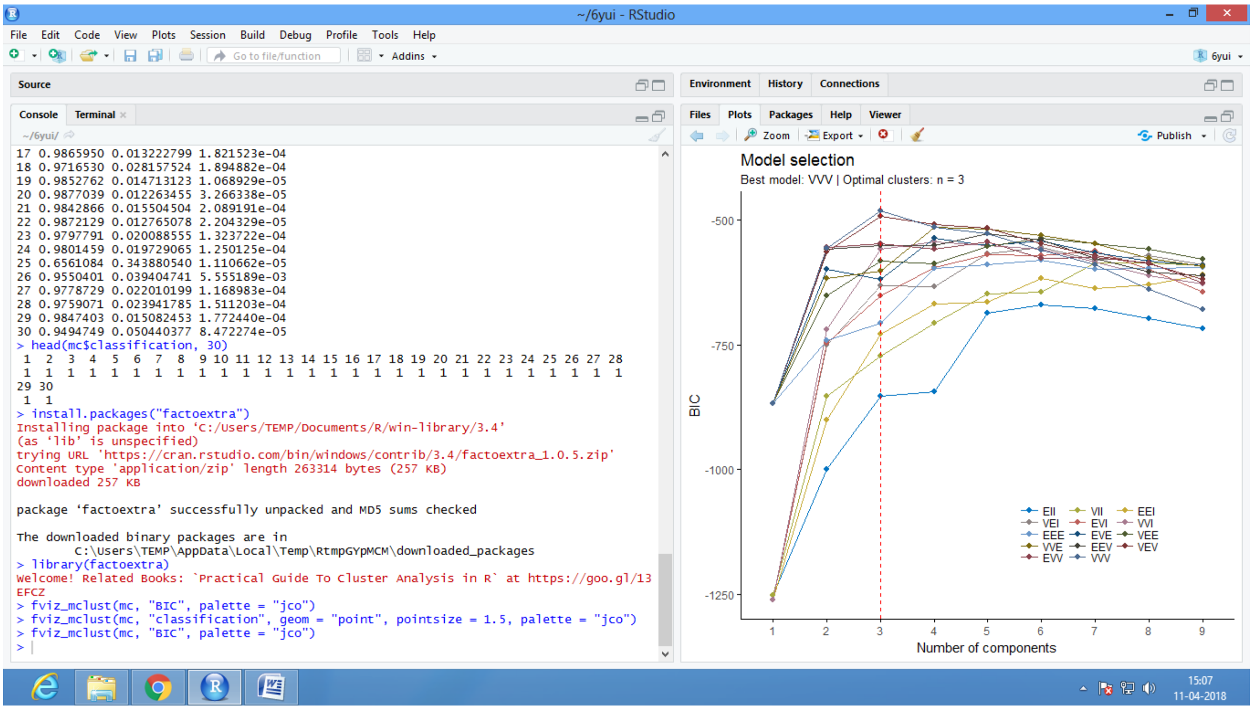
fviz_mclust(mc, "BIC", palette = "jco")

# Classification: plot showing the clustering

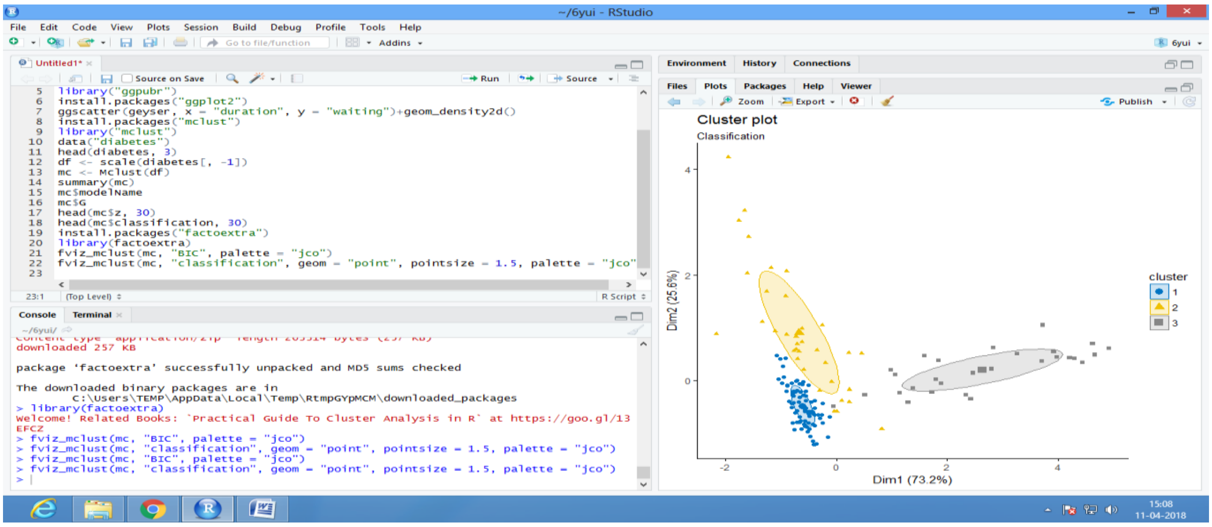
fviz_mclust(mc, "classification", geom = "point", pointsize = 1.5, palette = "jco")

# Classification uncertainty

fviz_mclust(mc, "uncertainty", palette = "jco")

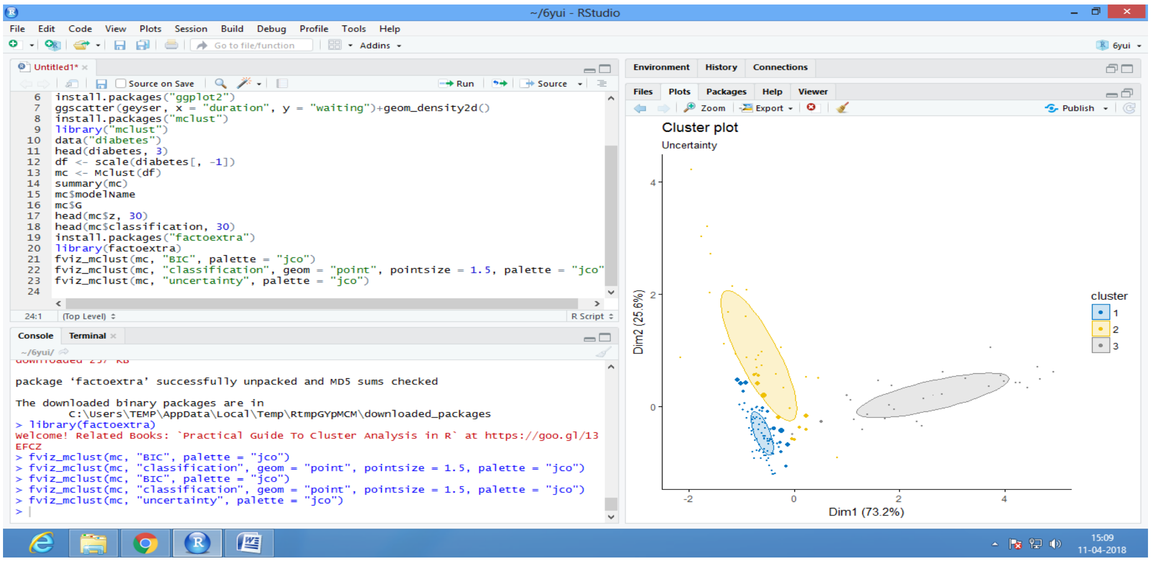


```
17 0.9865950 0.013222799 1.821523e-04
18 0.9716530 0.028157524 1.894882e-04
19 0.9852762 0.014713123 1.068929e-05
20 0.9877039 0.012263455 3.266338e-05
21 0.9842866 0.015504504 2.089191e-04
22 0.9872129 0.012765078 2.204329e-05
23 0.9797791 0.020088555 1.323722e-04
24 0.9801459 0.019729065 1.250125e-04
25 0.6561084 0.343880540 1.110662e-05
26 0.9550401 0.039404741 5.555189e-03
27 0.9778729 0.022010199 1.168983e-04
28 0.9759071 0.023941785 1.511203e-04
29 0.9847403 0.015082453 1.772440e-04
30 0.9494749 0.050440377 8.472274e-05
> head(mc$classification, 30)
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
 1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
29 30
 1  1
> install.packages("factoextra")
Installing package into ‘C:/Users/TEMP/Documents/R/win-library/3.4’
(as ‘lib’ is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/factoextra_1.0.5.zip'
Content type 'application/zip' length 263314 bytes (257 KB)
downloaded 257 KB

package ‘factoextra’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\TEMP\AppData\Local\Temp\RtmpGYpMCM\downloaded_packages
> library(factoextra)
Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13
EFCZ
> fviz_mclust(mc, "BIC", palette = "jco")
> fviz_mclust(mc, "classification", geom = "point", pointsize = 1.5, palette = "jco")
> fviz_mclust(mc, "BIC", palette = "jco")
> 
```



```
library("ggpubr")
install.packages("ggplot2")
ggscatter(geyser, x = "duration", y = "waiting")+geom_density2d()
install.packages("mclust")
library("mclust")
data("diabetes")
head(diabetes, 3)
df <- scale(diabetes[, -1])
mc <- Mclust(df)
summary(mc)
mc$modelName
mc$G
head(mc$z, 30)
head(mc$classification, 30)
install.packages("factoextra")
library(factoextra)
fviz_mclust(mc, "BIC", palette = "jco")
fviz_mclust(mc, "classification", geom = "point", pointsize = 1.5, palette = "jco"
```

```
downloaded 257 KB

package ‘factoextra’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\TEMP\AppData\Local\Temp\RtmpGYpMCM\downloaded_packages
> library(factoextra)
Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13
EFCZ
> fviz_mclust(mc, "BIC", palette = "jco")
> fviz_mclust(mc, "classification", geom = "point", pointsize = 1.5, palette = "jco")
> fviz_mclust(mc, "BIC", palette = "jco")
> fviz_mclust(mc, "classification", geom = "point", pointsize = 1.5, palette = "jco")
> |
```



```
install.packages("ggplot2")
ggscatter(geyser, x = "duration", y = "waiting")+geom_density2d()
install.packages("mclust")
library("mclust")
data("diabetes")
head(diabetes, 3)
df <- scale(diabetes[, -1])
mc <- Mclust(df)
summary(mc)
mc$modelName
mc$G
head(mc$z, 30)
head(mc$classification, 30)
install.packages("factoextra")
library(factoextra)
fviz_mclust(mc, "BIC", palette = "jco")
fviz_mclust(mc, "classification", geom = "point", pointsize = 1.5, palette = "jco"
fviz_mclust(mc, "uncertainty", palette = "jco")
```

```
package ‘factoextra’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\TEMP\AppData\Local\Temp\RtmpGYpMCM\downloaded_packages
> library(factoextra)
Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13
EFCZ
> fviz_mclust(mc, "BIC", palette = "jco")
> fviz_mclust(mc, "classification", geom = "point", pointsize = 1.5, palette = "jco")
> fviz_mclust(mc, "BIC", palette = "jco")
> fviz_mclust(mc, "classification", geom = "point", pointsize = 1.5, palette = "jco")
> fviz_mclust(mc, "uncertainty", palette = "jco")
> |
```

VIVA QUESTIONS

1. What is partitioning in clustering?

Partitional clustering (or partitioning clustering) are clustering methods used to classify observations, within a data set, into multiple groups based on their similarity. The algorithms require the analyst to specify the number of clusters to be generated.

2. What is the purpose of K-Mean Clustering?

The K-means clustering algorithm is used to find groups which have not been explicitly labeled in the data. This can be used to confirm business assumptions about what types of groups exist or to identify unknown groups in complex data sets.

3. What is clustering used for?

Clustering is an unsupervised machine learning method of identifying and grouping similar data points in larger datasets without concern for the specific outcome.

4. How does K-mean Clustering work?

K-means clustering uses “centroids”, K different randomly-initiated points in the data, and assigns every data point to the nearest centroid.

5. What are the advantages and disadvantages of k-means algorithm?

Advantages

1) If variables are huge, then K-Means most of the times computationally faster than hierarchical clustering, if we keep k smalls.
2) K-Means produce tighter clusters than hierarchical clustering, especially if the clusters are globular.

Disadvantages

1) Difficult to predict K-Value.
2) With global cluster, it didn't work well.
3) Different initial partitions can result in different final clusters.
4) It does not work well with clusters (in the original data) of Different size and Different density

WEEK9

DATA VISUALIZATION USING PIE, BAR, BOXPLOT CHART PLOTTING FRAMEWORK

IMPLEMENTATION:

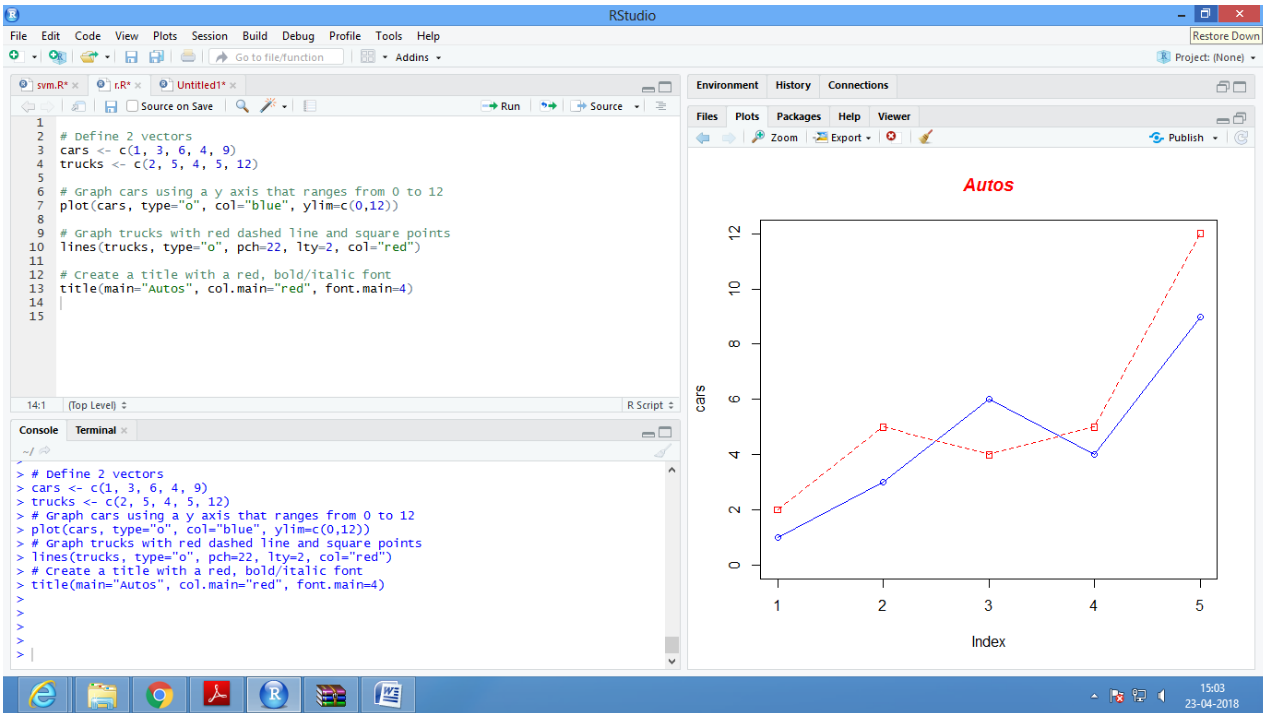
Step 1: Define two vectors as truck and car

Step 2: Using plot function specify the y-axis range directly so it will be large enough to fit the truck data

Step 3: plot cars using a y axis that ranges from 0 to 12

Step 4: plot trucks with red dashed line and square point strucks with red dashed line and square points

Step 5: Create a title with a red, bold/italic font



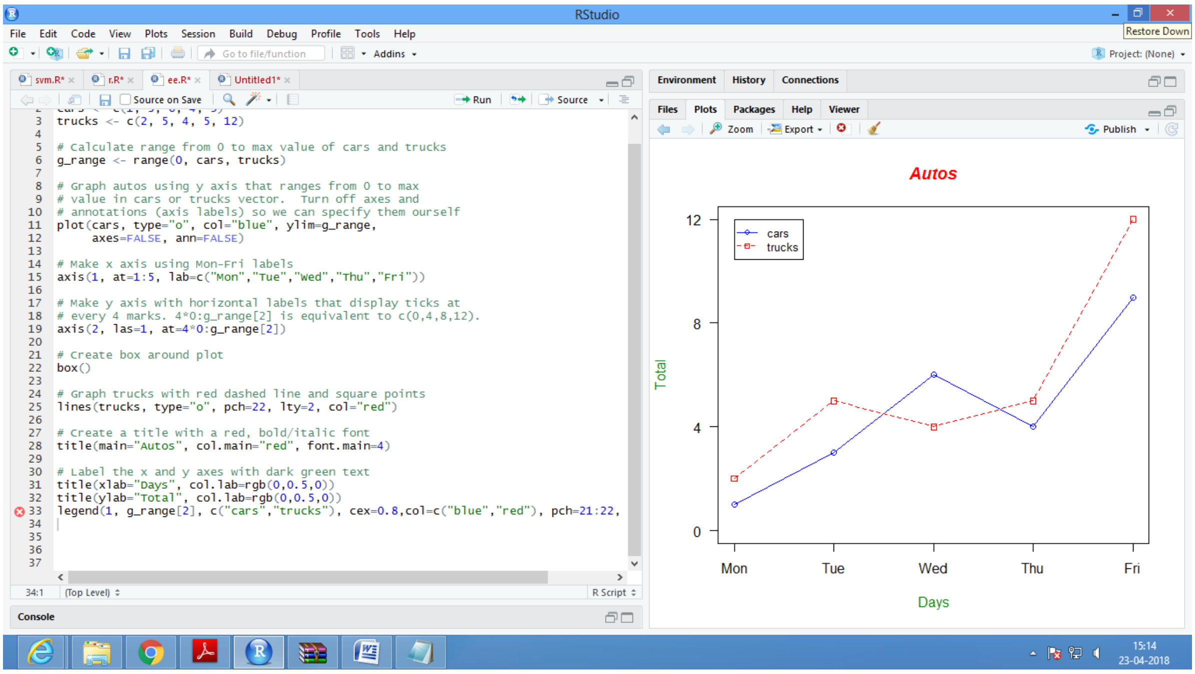
```
# Define 2 vectors
cars <- c(1, 3, 6, 4, 9)
trucks <- c(2, 5, 4, 5, 12)

# Graph cars using a y axis that ranges from 0 to 12
plot(cars, type="o", col="blue", ylim=c(0,12))

# Graph trucks with red dashed line and square points
lines(trucks, type="o", pch=22, lty=2, col="red")

# Create a title with a red, bold/italic font
title(main="Autos", col.main="red", font.main=4)
```

```
> # Define 2 vectors
> cars <- c(1, 3, 6, 4, 9)
> trucks <- c(2, 5, 4, 5, 12)
> # Graph cars using a y axis that ranges from 0 to 12
> plot(cars, type="o", col="blue", ylim=c(0,12))
> # Graph trucks with red dashed line and square points
> lines(trucks, type="o", pch=22, lty=2, col="red")
> # Create a title with a red, bold/italic font
> title(main="Autos", col.main="red", font.main=4)
>
>
>
>
> |
```

Step 6: Create a legend at (1, g_range[2]) that is slightly smaller (cex) and uses the same line colors and points used by the actual plots.

legend(1, g_range[2], c("cars","trucks"), cex=0.8,col=c("blue","red"), pch=21:22, lty=1:2);

x`



```
trucks <- c(2, 5, 4, 5, 12)

# Calculate range from 0 to max value of cars and trucks
g_range <- range(0, cars, trucks)

# Graph autos using y axis that ranges from 0 to max
# value in cars or trucks vector.  Turn off axes and
# annotations (axis labels) so we can specify them ourself
plot(cars, type="o", col="blue", ylim=g_range,
     axes=FALSE, ann=FALSE)

# Make x axis using Mon-Fri labels
axis(1, at=1:5, lab=c("Mon","Tue","Wed","Thu","Fri"))

# Make y axis with horizontal labels that display ticks at
# every 4 marks. 4*0:g_range[2] is equivalent to c(0,4,8,12).
axis(2, las=1, at=4*0:g_range[2])

# Create box around plot
box()

# Graph trucks with red dashed line and square points
lines(trucks, type="o", pch=22, lty=2, col="red")

# Create a title with a red, bold/italic font
title(main="Autos", col.main="red", font.main=4)

# Label the x and y axes with dark green text
title(xlab="Days", col.lab=rgb(0,0.5,0))
title(ylab="Total", col.lab=rgb(0,0.5,0))
legend(1, g_range[2], c("cars","trucks"), cex=0.8,col=c("blue","red"), pch=21:22,
```

VIVA QUESTIONS

1. What is data visualization?

The representation of information in the form of a chart, diagram, picture, etc.

2. What are some important features of good data visualization?

- It's visually appealing. ...
- It's scalable. ...
- It gives the user the right information. ...
- It's accessible. ...
- It allows rapid development and deployment.

3. What is a bar plot? For what type of data is bar plot usually used for?

Bar charts are one of the many techniques used to present data in a visual form so that the reader may readily recognize patterns or trends. Bar charts usually present categorical variables, discrete variables or continuous variables grouped in class intervals.

4. What is an outlier?

Outlier is an observation of data that does not fit the rest of the data. It is sometimes called an extreme value. When you graph an outlier, it will appear not to fit the pattern of the graph.

5. What type of data is box-plots usually used for? Why?

Box plots are used to show distributions of numeric data values, especially when you want to compare them between multiple groups. They are built to provide high-level information at a glance, offering general information about a group of data's symmetry, skew, variance, and outliers.

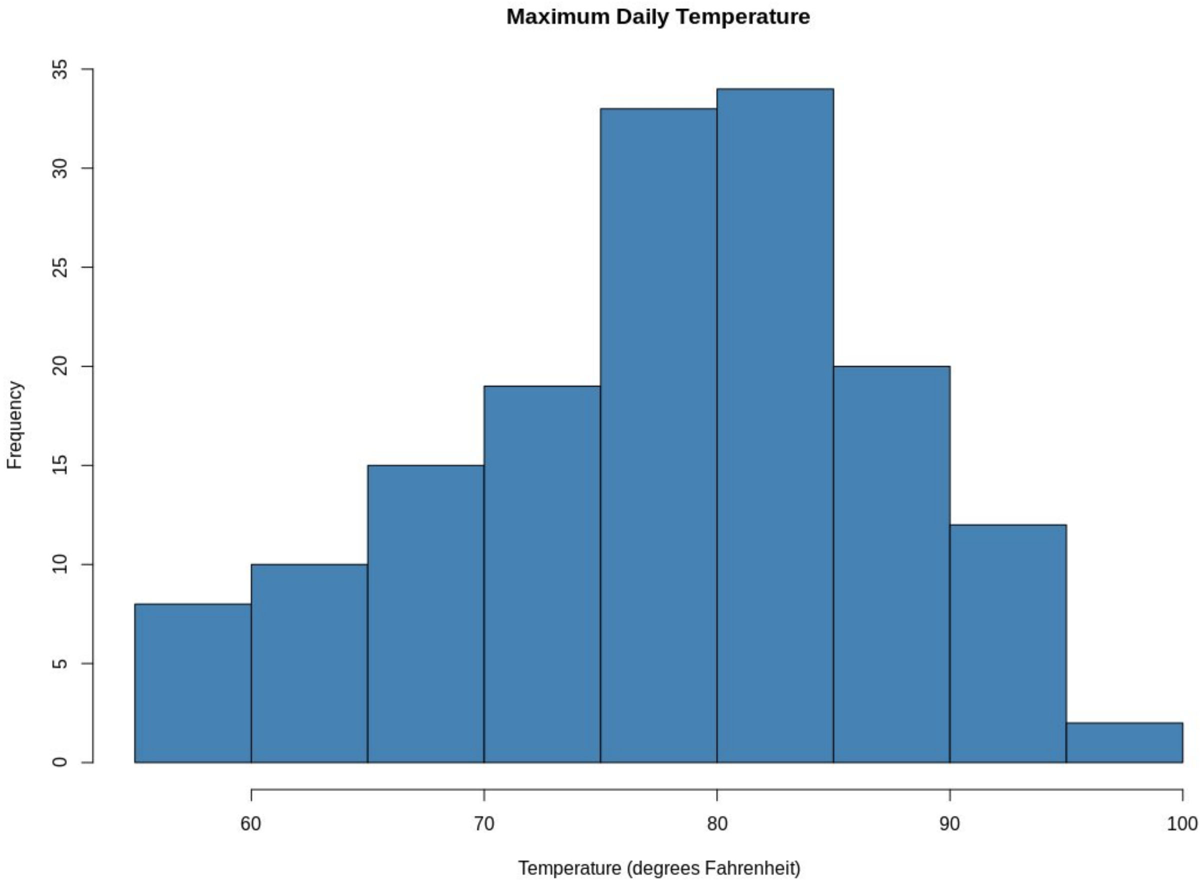
WEEK10

DATA VISUALIZATION USING HISTOGRAM PLOTTING FRAMEWORK

Histograms are standard in some academic fields, but they're usually reserved for the senior-most levels. These charts are best with highly precise or accurate numbers in R.

It ultimately provides a probability estimate of a variable — the period of time before a project's completion, for example. R provides a simple function for this as well.

```
# histogram of frequency of ozone values in 'airquality' dataset
hist(airquality$Temp,col='steelblue',main='Maximum Daily Temperature',
    xlab='Temperature (degrees Fahrenheit)')
```



VIVA QUESTIONS

1. When analyzing a histogram, what are some of the features to look for?

A histogram has an appearance similar to a vertical bar chart, but there are no gaps between the bars. Generally, a histogram will have bars of equal width.

2. What type of data is histogram usually used for?

The histogram is a popular graphing tool. It is used to summarize discrete or continuous data that are measured on an interval scale.

3. When will you use a histogram and when will you use a bar chart? Explain with an example.

The histogram is used to showcase a graphical presentation that represents the data in the form of frequency; whereas a bar chart is also a graphical representation of data and the information that is used for the comparison of two categories.

4. How do you make multiple plots to a single page layout in R?

To put multiple plots on the same graphics pages in R, you can use the graphics parameter mfrow or mfcol.

5. How missing values and impossible values are represented in R?

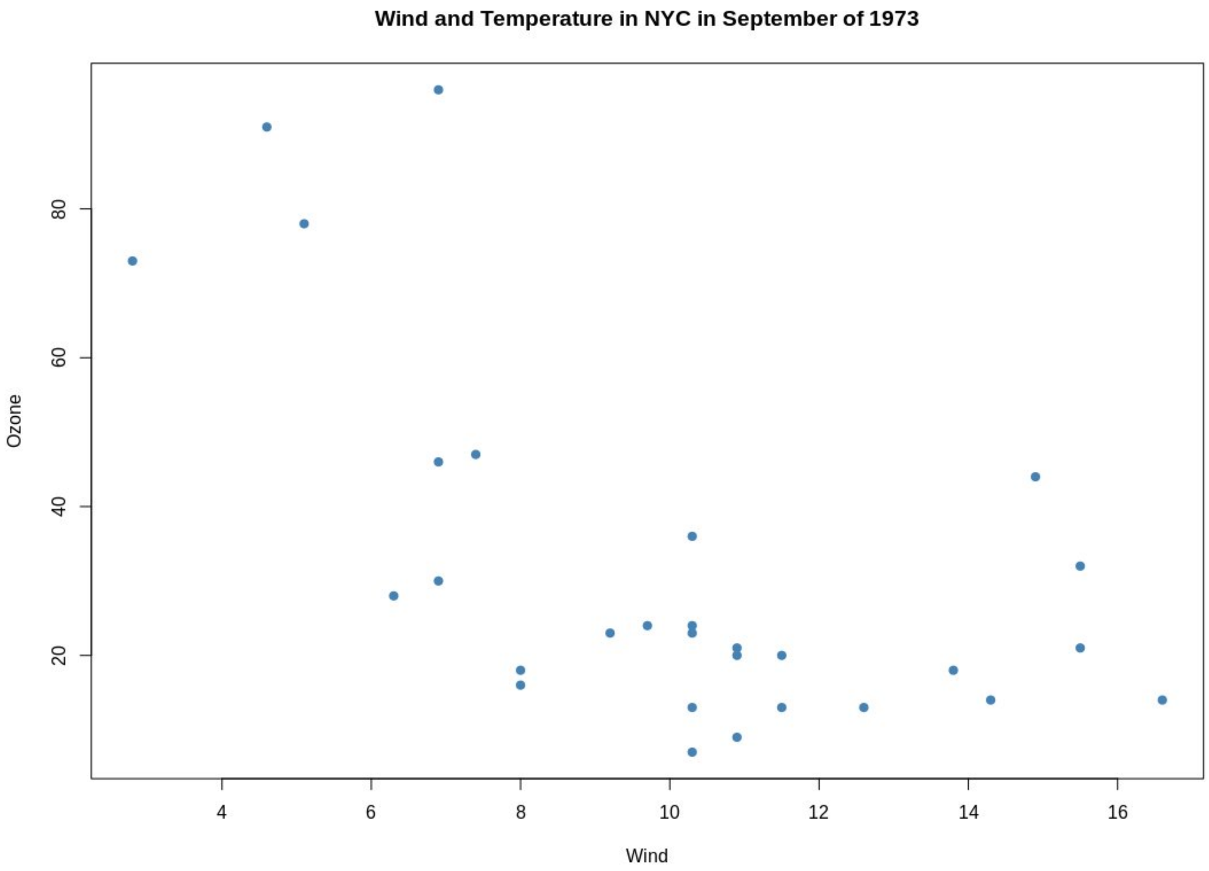
In R, missing values are represented by the symbol NA (not available). Impossible values (domain errors like division by 0 et logs of negative numbers are represented by the symbol NaN (Not-A-Number). NA is used for both numeric and string data.

WEEK11

DATA VISUALIZATION USING LINE GRAPH PLOTTING, SCATTERPLOT PLOTTING FRAMEWORK

Plotting is a popular alternative to charting or graphing. It provides a unique visualization involving various dots. The most standard iteration — the scatter plot — tracks two continuous variables over the course of time. A basic application of the scatter plot involves tracking the height and weight of children throughout the years.
Scatter plots are useful when trying to avoid misinformation in the visualization. Only use a plot if you're sure the audience is familiar with that type of chart, and always use it sparingly. When in doubt, go with one of your other options.

```
# Plot Ozone and Temperature measurements for only the month of September
with(subset(airquality,Month==9),plot(Wind,Ozone,col='steelblue',pch=20,cex=1.5))
title('Wind and Temperature in NYC in September of 1973')
```



VIVA QUESTIONS

1. What is a scatter plot?
A scatterplot is a type of data display that shows the relationship between two numerical variables.

2. What features might be visible in scatterplots?
Visually show the strength of the relationship between the variables.

3. Explain what should be done with suspected or missing data?

- Use deletion methods to eliminate missing data. The deletion methods only work for certain datasets where participants have missing fields.
- Use regression analysis to systematically eliminate data.
- Data scientists can use data imputation techniques.

4. For what type of data is scatter plot usually used for?

Scatter plots' primary uses are to observe and show relationships between two numeric variables. Relationships between variables can be described in many ways: positive or negative, strong or weak, linear or nonlinear. A scatter plot can also be useful for identifying other patterns in data.

WEEK12

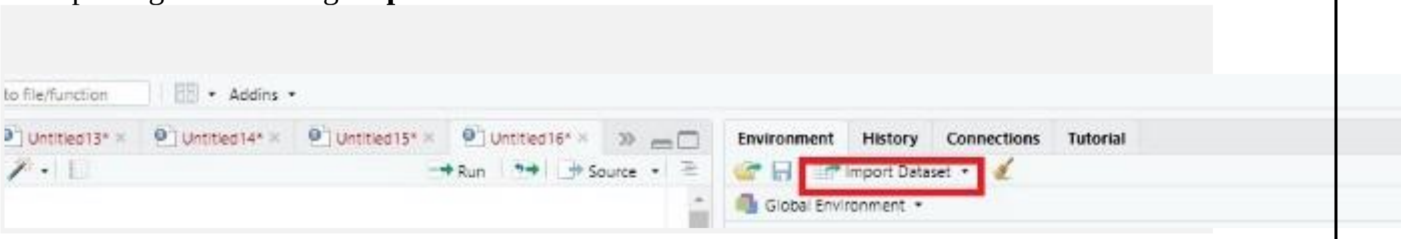
APPLICATION TO ANALYZE STOCK MARKET USING R LANGUAGE.

The stock market of India fell by 41.2% to be precise during the period between January to April. At the starting of this year, most of the companies were at their peak and turned down to be 52 weeks low (considered as lowest ever in the year). Now, the stock market started recovering and the gain percentage in an average is 43% but still, there are pending recovery values that need to be retrieved. This analysis is taken sector-wise and is compared with the NSE (National Stock Exchange) indices movement. All these data and scrutiny (observation) is between Jan-01–2020 to Jun-09–2020. This analysis also reveals how the stocks and index performed during this period and which are the best and worst recovered companies.

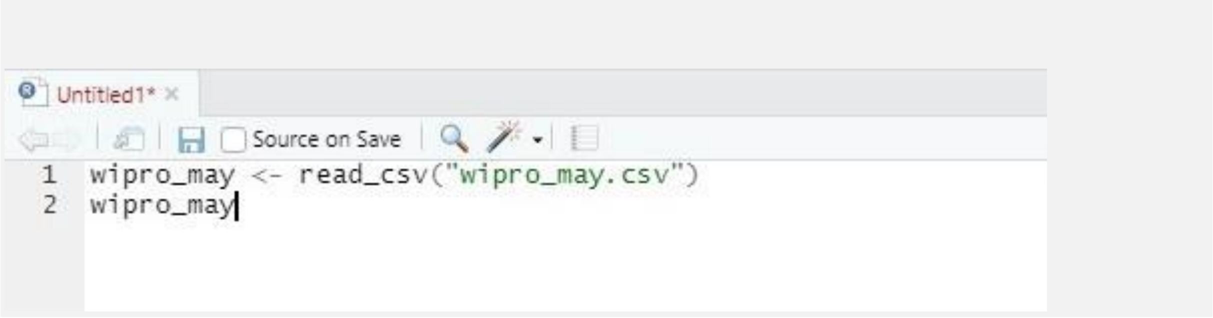
R as an Analysis Tool

R programming language is considered one of the powerful languages for Data Science and it is also very accessible for users. The software used for coding R is RStudio and there are also other platforms to code R which you can prefer. This language is widely used by Statisticians and Data miners for data analysis and for developing statistical software. Let’s see how to do Stock Analysis with R :

1. Importing dataset using **Import Dataset** :



2. Read the dataset file which you have imported :



```
wipro_may <- read_csv("wipro_may.csv")
wipro_may
```

3. Coding for our required Analysis (I have extracted the data from NSE Website, the above one is just an example, now I’m coding with a different dataset)

This is an example with Wipro :

```
library(dplyr)
#Examining the lowest value of wipro
> min(wipro_index$Close)
162.35
#Examining the highest value of wipro
> max(wipro_index$Close)
257.2
#Calculating fall value of wipro
> max(wipro_index$Close) - min(wipro_index$Close)
94.85
#Calculating fall percentage of wipro
> 94.85 / max(wipro_index$Close) * 100
36.88
#Examining the recovery of wipro
> max(wipro_may$'Close Price')
226.45
#Calculating recovery value of wipro
> max(wipro_may$'Close Price') - min(wipro_index$Close)
64.1
#Calculating recovery percentage of wipro
> 64.1 / min(wipro_index$Close) * 100
39.48
#Calculating pending recovery value of wipro
> max(wipro_index$Close) - max(wipro_may$'Close Price')
30.75
#Calculating pending recovery percentage of wipro
> 30.75 / max(wipro_index$Close) * 100
11.96
#Summarizing our analysis of wipro
> wipro_index  %>%
   summarize(high = max(wipro_index$Close) ,
   low = min(wipro_index$Close) ,
   fall_value = max(wipro_index$Close) - min(wipro_index$Close) ,
   fall_percentage = 94.85 / max(wipro_index$Close) * 100 ,
   recovery = max(wipro_may$`Close Price`) ,
   recovery_value  = max(wipro_may$`Close Price`) -        min(wipro_index$Close) ,
   recovery_percentage = 64.10 / min(wipro_index$Close) * 100 ,
   pending_recovery = max(wipro_index$Close) - max(wipro_may$`Close Price`) ,
   pending_percentage = 30.75 / max(wipro_index$Close) * 100 )
#Output
high    low     fall_value  fall_percentage  recovery  recovery_value
257.2   162.35  94.85       36.88            226.45    64.1
recovery_percentage  pending_recovery  pending_percentage           39.4826            30.75
11.96
```

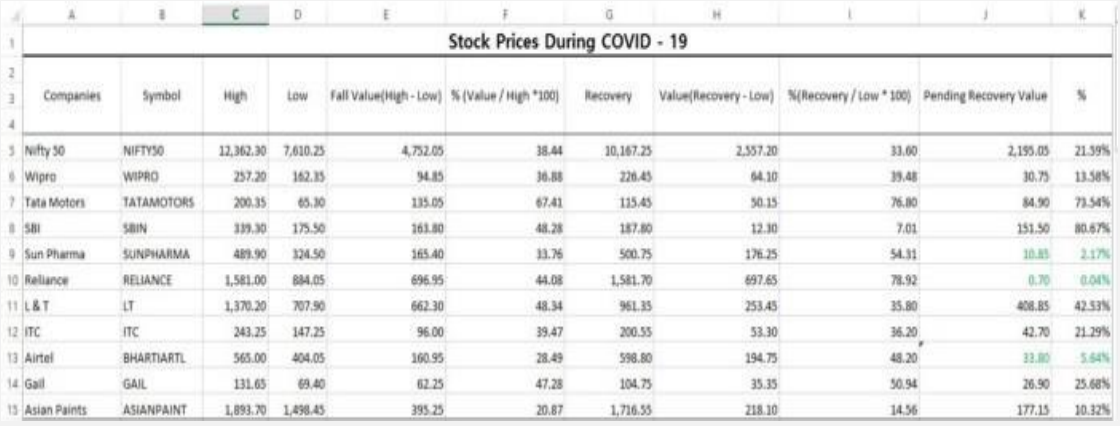
From this coding, we can generate our specified result which we want to form the dataset. After getting our data from R we can use Excel for a neat tabulation for our Analysis. In this program, I have used functions including “min,” ”max,” “summarize.”

min — “min” function returns the minimum value of a vector or column.

max — It returns the highest or the maximum value of a vector or column.

summarize — The “summarize” function reduces a data frame into a summary of just one vector or value.

Tabulation of our Data



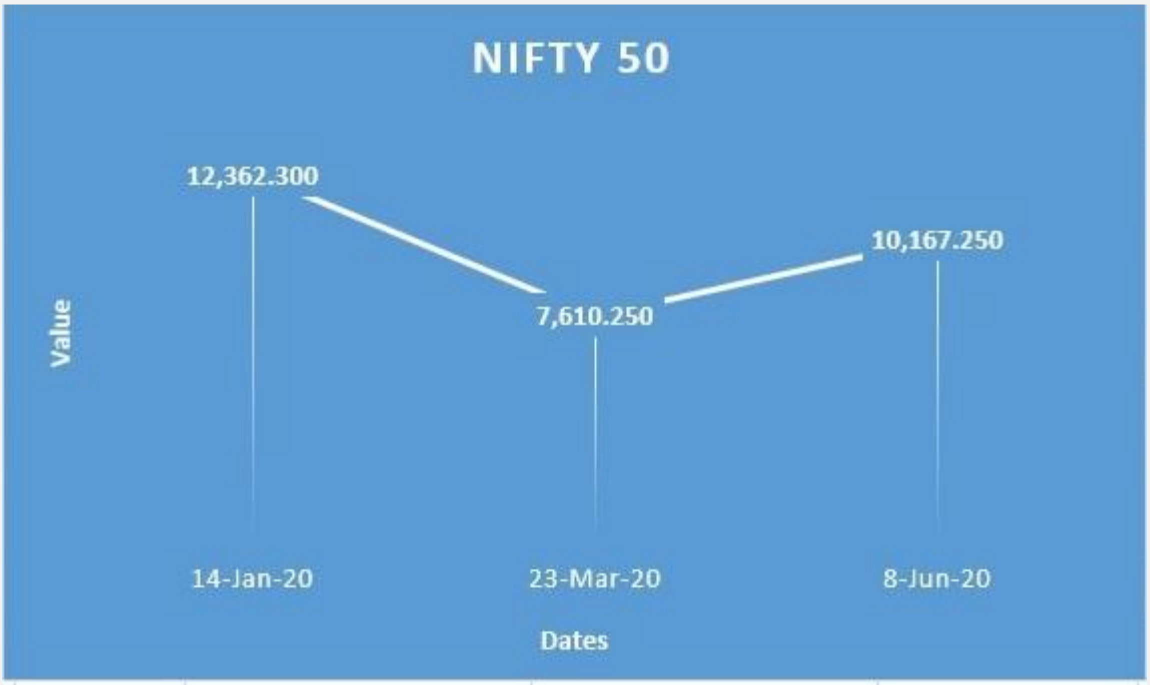
Stock Prices During COVID - 19

Companies	Symbol	High	Low	Fall Value(High - Low)	% (Value / High *100)	Recovery	Value(Recovery - Low)	%(Recovery / Low * 100)	Pending Recovery Value	%
Nifty 50	NIFTY50	12,362.30	7,610.25	4,752.05	38.44	10,167.25	2,557.20	33.60	2,195.05	21.59%
Wipro	WIPRO	257.20	162.35	94.85	36.88	226.45	64.10	39.48	30.75	13.58%
Tata Motors	TATAMOTORS	200.35	65.30	135.05	67.41	115.45	50.15	76.80	84.90	73.54%
SBI	SBIN	339.30	175.50	163.80	48.28	187.80	12.30	7.01	151.50	80.67%
Sun Pharma	SUNPHARMA	489.90	324.50	165.40	33.76	500.75	176.25	54.31	10.85	2.17%
Reliance	RELIANCE	1,581.00	884.05	696.95	44.08	1,581.70	697.65	78.92	0.70	0.04%
L & T	LT	1,370.20	707.90	662.30	48.34	961.35	253.45	35.80	408.85	42.53%
ITC	ITC	243.25	147.25	96.00	39.47	200.55	53.30	36.20	42.70	21.29%
Airtel	BHARTIARTL	565.00	404.05	160.95	28.49	598.80	194.75	48.20	33.80	5.64%
Gail	GAIL	131.65	69.40	62.25	47.28	104.75	35.35	50.94	26.90	25.68%
Asian Paints	ASIANPAINT	1,893.70	1,498.45	395.25	20.87	1,716.55	218.10	14.56	177.15	10.32%

Graph comparison of Values

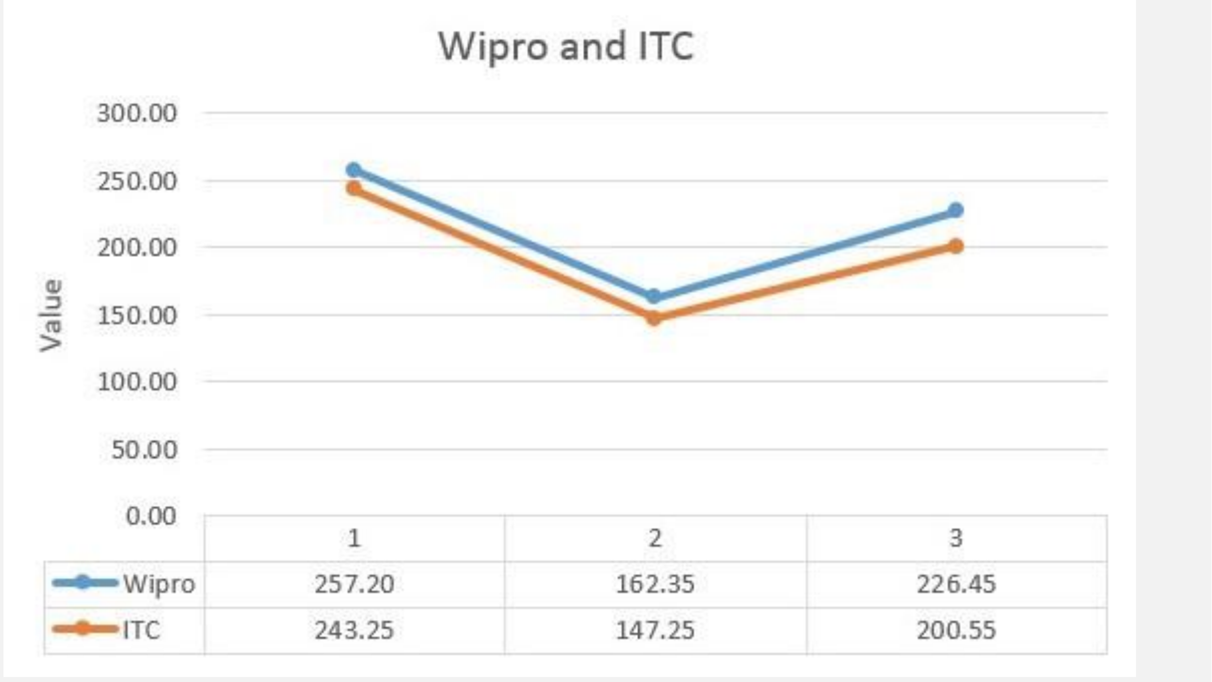
Data visualization is very important in the financial world, especially for Analysis. Instead of creating a tabulation, a simple graph or any way of representation is easy to understand and it is very useful for comparison too. In this blog, I’ve compared the values of companies using line graphs for easy understanding.

1. Nifty 50



From this graph, we can understand that Nifty 50 index was at its peak on 14-Jan-2020 and had a fall price of 7,610.250 on 23-Mar-2020. We can also observe that the price of 12,362.30 came down to 7,610.250 in just 68 days which is 38.44% from its peak value. The next thing we can observe from this graph is the recovery value of Nifty 50. The stock values of Nifty 50 zoomed from 7,610 to 10,167 in only 78 days and had a jump percentage of 33.6 %. The last thing about this graph is that even though the prices spiked in 78 days it couldn't make up to its peak price of 12,362. So, we can conclude that the pending recovery value of Nifty 50 is still 2,195 and needs to shine up for a percentage of 21.59%.

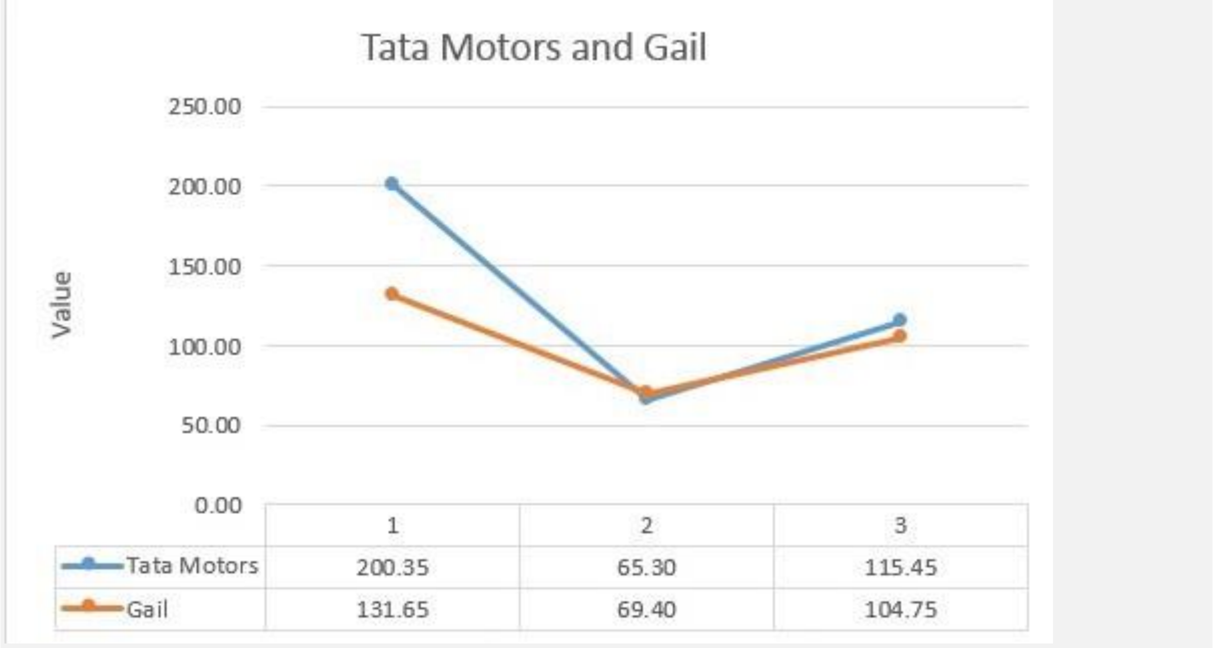
2. Wipro and ITC (Indian Tobacco Company)



	1	2	3
Wipro	257.20	162.35	226.45
ITC	243.25	147.25	200.55

This is a graph comparing stock prices of two companies, Wipro and ITC. We can observe that the movement of both companies is relatively similar during this period. The stock price of Wipro in January was 257.2, had a fall of 94.85, and reached a stock price of 162. The percentage of fall value is 36.88%. Soon it recovered in June and zoomed to a price of 226 and had a recovery rate of 39.4%. When we compare the peak price of Wipro with the recovery price, the pending recovery value is 30.75 and has to rise by 13.58%. Coming to ITC, its peak price was 243.25 and declined to a price of 147.25 during this period. This company had a fall value of 96 and declined by 39.47%. The stock prices are appreciated to 200.55 between March and June, the recovery rate was 36.20%. In case of pending recovery value, Wipro has a less pending recovery rate when compared to ITC which is at 21.29%. We can say that Wipro performed well during this period as it has less pending recovery rate when compared to ITC.

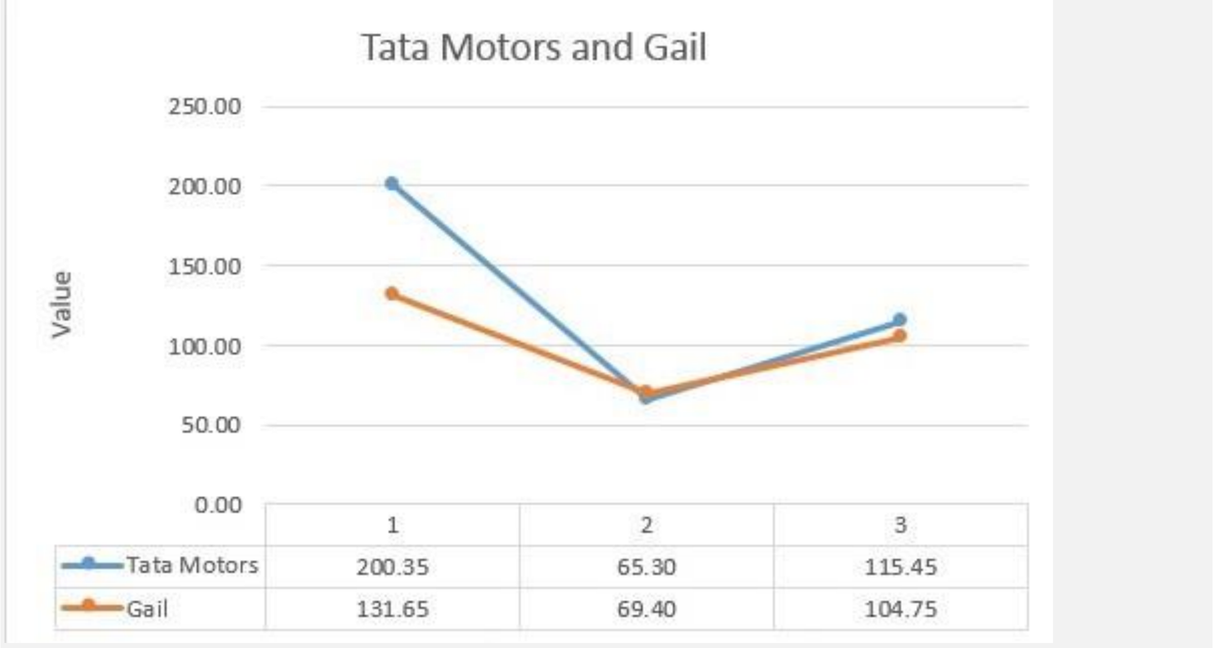
3. Tata Motors and Gail



	1	2	3
Tata Motors	200.35	65.30	115.45
Gail	131.65	69.40	104.75

This graph compares the stock prices of two companies Tata Motors and Gail. We can notice that the peak price of these two companies differs but in case of declined price, they are similar. The peak price of Tata Motors was 200.35 and the declined price is 65.30. From these prices, we can observe that Tata Motors has phenomenal fall value and rate. The fall value of Tata motors is 135.05 and declined percentage is 64.41%. In Spite of having drastic depreciation in prices Tata Motors recovered in a very short span of time. It zoomed to a price of 115.45 from a declined price of 65.30. The recovery rate is 76.80%. But still, the pending recovery value is 84.90 with a rate of 73.54%. Coming to Gail the price movements are not so phenomenal when compared to Tata Motors. The peak price was 131.65 and the declined price is 69.40 which is more similar to Tata Motors. The fall value and rate of Gail are 62.25 and 42.28% respectively. The recovery price is 104.75 which can be considered as a good comeback when compared to Tata Motors. The pending recovery value of Gail is 26.90 with a percentage of 25.68%. This graph comparison clearly reveals that the price movements of Tata Motors during this period were very volatile.

4. Reliance, L & T and Asian Paints



This graph comprises the stock prices of three companies namely Reliance, L & T, and Asian Paints. The stock price of Reliance was it's peak value of 1,581 and had a price depreciation to 884.05. But the recovery rate was phenomenally high which is higher than it's peak price. The recovery price is 1,581.70, an increased rate of 0.7% from its peak price. Obviously there is no pending recovery rate as the price exceeds its peak value. The highest stock price of L & T was 1,370.20. The fall price and value of L & T are 707.90 and 662.30 respectively. The down rate of L & T is 48.34%. The recovery price of L & T is 961.35 and the rate is 35.80. The pending recovery value of L & T is 408.85. From this we can say that the performance of L & T is poor when compared to L & T. Coming to Asian Paints, considering all the peak prices Asian Paints is top among that. The peak price of Asian Paints was 1,893.70 and the declined price is 1,498.45. The recovery price is 1,716.55, from this we can note that the pending value and rate of Asian Paints would be lower than L & T. The pending recovery value and rate of Asian Paints are 177.15 and 10.32%. Like this, we can predict the values easily in charts or any type of representation without doing any calculations.

VIVA QUESTIONS

1. Is R good for Stock analysis?

Yes, R has excellent packages for analyzing stock data, so I feel there should be a "translation" of the post for using R for stock data analysis.

2. How do I get Stock data in R?

By using quantmod package. You can install it by typing the command "install packages("quantmod")" in your R console.

3. How do you analyze data from the stock market?

The quantmod package for R is designed to assist the quantitative trader in the development, testing, and deployment of statistically based trading models.

4. Which app is best for stock market analysis?

MoneyControl, Stock Edge, Economic Times (ET) Markets

5. List out some of the functions that R provides?

Abline, abs, addmargins, aggregare and etc..

ADDITIONAL PROGRAMS

IMPLEMENT BASIC PROGRAMS IN R BY USING DATA STRUCTURES

```
> myString <- "Hello, World!"
> print ( myString)
[1] "Hello, World!"
```

Vectors:

```
apple <- c('red','green',"yellow")
print(apple)
print(class(apple))
[1] "red"    "green"  "yellow"
[1] "character"
```

Lists:

```
list1 <- list(c(2,5,3),21.3,sin)
print(list1)
[[1]]
[1] 2 5 3
[[2]]
[1] 21.3
[[3]]
function (x)  .Primitive("sin")
```

Matrices:

```
M=matrix(c('a','a','b','c','b','a'),
  nrow=2,ncol=3,byrow=TRUE)
print(M)
    [,1] [,2] [,3]
[1,] "a"  "a"  "b"
[2,] "c"  "b"  "a"
```

Arrays:

```
a <- array(c('green','yellow'),dim = c(3,3,1))
print(a)
, , 1
     [,1]     [,2]     [,3]
[1,] "green"  "yellow" "green"
[2,] "yellow" "green"  "yellow"
[3,] "green"  "yellow" "green"
```

Data Frames:

```
BMI <- data.frame(
   gender = c("Male", "Male","Female"),
   height = c(152, 171.5, 165),
   weight = c(81,93, 78),
   Age = c(42,38,26)
)
print(BMI)
  gender height weight Age
1   Male  152.0     81  42
2   Male  171.5     93  38
3 Female  165.0     78  26
```

1. **Write a R program to take input from the user (name and age) and display the values. Also print the version of R installation.**

```
name = readline(prompt="Input your name: ")
age =  readline(prompt="Input your age: ")
print(paste("My name is",name, "and I am",age ,"years old."))
print(R.version.string)
```

Output

```
Input your name: abc
Input your age: 25
[1] "My name is abc and I am  25 years old."
[1] "R version 3.4.4 (2018-03-15)"
```

2. Write a R program to get the details of the objects in memory.

```
name = "Python";
n1 =  10;
n2 =  0.5
nums = c(10, 20, 30, 40, 50, 60)
print(ls())
print("Details of the objects in memory:")
print(ls.str())
```

Output

```
[1] "n1"   "n2"   "name" "nums"
[1] "Details of the objects in memory:"
n1 :  num 10
n2 :  num 0.5
name :  chr "Python"
nums :  num [1:6] 10 20 30 40 50 60
```

3. Write a R program to create a sequence of numbers from 20 to 50 and find the mean of numbers from 20 to 60 and sum of numbers from 51 to 91.

```
print("Sequence of numbers from 20 to 50:")
print(seq(20,50))
print("Mean of numbers from 20 to 60:")
print(mean(20:60))
print("Sum of numbers from 51 to 91:")
print(sum(51:91))
```

Output

```
[1] "Sequence of numbers from 20 to 50:"
 [1] 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44
[26] 45 46 47 48 49 50
[1] "Mean of numbers from 20 to 60:"
[1] 40
[1] "Sum of numbers from 51 to 91:"
```

```
[1] 2911
```

4. Write a R program to create a vector which contains 10 random integer values between -50 and +50.

```
v = sample(-50:50, 10, replace=TRUE)
print("Content of the vector:")
print("10 random integer values between -50 and +50:")
print(v)
```

Output

```
[1] "Content of the vector:"
[1] "10 random integer values between -50 and +50:"
 [1]  31 -13 -21  42  49 -39  20  12  39  -2
```

5. Write a R program to get the first 10 Fibonacci numbers.

```
Fibonacci <- numeric(10)
Fibonacci[1] <- Fibonacci[2] <- 1
for (i in 3:10) Fibonacci[i] <- Fibonacci[i - 2] + Fibonacci[i - 1]
print("First 10 Fibonacci numbers:")
print(Fibonacci)
```

Output

```
[1] "First 10 Fibonacci numbers:"
 [1]  1  1  2  3  5  8 13 21 34 55
```

6.Write a R program to get all prime numbers up to a given number (based on the sieve of Eratosthenes).

```
prime_numbers <- function(n) {
if (n >= 2) {
 x = seq(2, n)
 prime_nums = c()
 for (i in seq(2, n)) {
 if (any(x == i)) {
 prime_nums = c(prime_nums, i)
 x = c(x[(x %% i) != 0], i)
 }
 }
 return(prime_nums)
 }
 else
 {
 stop("Input number should be at least 2.")
 }
 }
prime_numbers(12)
```

Output

```
[1]  2  3  5  7 11
```

7.Write a R program to print the numbers from 1 to 100 and print "Fizz" for multiples of 3, print "Buzz" for multiples of 5, and print "FizzBuzz" for multiples of both

```
for (n in 1:100) {

 if (n %% 3 == 0 & n %% 5 == 0) {print("FizzBuzz")}

 else if (n %% 3 == 0) {print("Fizz")}

 else if (n %% 5 == 0) {print("Buzz")}

 else print(n)

}
```

Ouput

```
[1] 1
[1] 2
[1] "Fizz"
[1] 4
[1] "Buzz"
[1] "Fizz"
[1] 7
[1] 8
[1] "Fizz"
[1] "Buzz"
[1] 11
[1] "Fizz"
[1] 13
[1] 14
[1] "FizzBuzz"
[1] 16
[1] 17
[1] "Fizz"
[1] 19
[1] "Buzz"
[1] "Fizz"
[1] 22
[1] 23
[1] "Fizz"
[1] "Buzz"
[1] 26
[1] "Fizz"
[1] 28
[1] 29
[1] "FizzBuzz"
[1] 31
[1] 32
[1] "Fizz"
[1] 34
[1] "Buzz"
```

```
[1] "Fizz"
[1] 37
[1] 38
[1] "Fizz"
[1] "Buzz"
[1] 41
[1] "Fizz"
[1] 43
[1] 44
[1] "FizzBuzz"
[1] 46
[1] 47
[1] "Fizz"
[1] 49
[1] "Buzz"
[1] "Fizz"
[1] 52
[1] 53
[1] "Fizz"
[1] "Buzz"
[1] 56
[1] "Fizz"
[1] 58
[1] 59
[1] "FizzBuzz"
[1] 61
[1] 62
[1] "Fizz"
[1] 64
[1] "Buzz"
[1] "Fizz"
[1] 67
[1] 68
[1] "Fizz"
[1] "Buzz"
[1] 71
[1] "Fizz"
[1] 73
[1] 74
[1] "FizzBuzz"
[1] 76
[1] 77
[1] "Fizz"
[1] 79
[1] "Buzz"
[1] "Fizz"
[1] 82
[1] 83
[1] "Fizz"
[1] "Buzz"
```

```
[1] 86
[1] "Fizz"
[1] 88
[1] 89
[1] "FizzBuzz"
[1] 91
[1] 92
[1] "Fizz"
[1] 94
[1] "Buzz"
[1] "Fizz"
[1] 97
[1] 98
[1] "Fizz"
[1] "Buzz"
```

8.Write a R program to extract first 10 english letter in lower case and last 10 letters in upper case and extract letters between 22nd to 24th letters in upper case.

```
print("First 10 letters in lower case:")
t = head(letters, 10)
print(t)
print("Last 10 letters in upper case:")
t = tail(LETTERS, 10)
print(t)
print("Letters between 22nd to 24th letters in upper case:")
e = tail(LETTERS[22:24])
print(e)
```

Output

```
[1] "First 10 letters in lower case:"
 [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j"
[1] "Last 10 letters in upper case:"
 [1] "Q" "R" "S" "T" "U" "V" "W" "X" "Y" "Z"
[1] "Letters between 22nd to 24th letters in upper case:"
[1] "V" "W" "X"
```

9.Write a R program to find the factors of a given number.

```
print_factors = function(n) {
print(paste("The factors of",n,"are:"))
for(i in 1:n) {
if((n %% i) == 0) {
print(i)
}
}
}
print_factors(4)
print_factors(7)
print_factors(12)
```

Output

```
[1] "The factors of 4 are:"
[1] 1
[1] 2
[1] 4
[1] "The factors of 7 are:"
[1] 1
[1] 7
[1] "The factors of 12 are:"
[1] 1
[1] 2
[1] 3
[1] 4
[1] 6
[1] 12
```

10.Write a R program to find the maximum and the minimum value of a given vector.

```
nums = c(10, 20, 30, 40, 50, 60)
print('Original vector:')
print(nums)
print(paste("Maximum value of the said vector:",max(nums)))
print(paste("Minimum value of the said vector:",min(nums)))
```

Output

```
[1] "Original vector:"
[1] 10 20 30 40 50 60
[1] "Maximum value of the said vector: 60"
[1] "Minimum value of the said vector: 10"
```

11. Write a R program to get the unique elements of a given string and unique numbers of vector

str1 = "The quick brown fox jumps over the lazy dog."

```
print("Original vector(string)")
print(str1)
print("Unique elements of the said vector:")
print(unique(tolower(str1)))
nums = c(1, 2, 2, 3, 4, 4, 5, 6)
print("Original vector(number)")
print(nums)
print("Unique elements of the said vector:")
print(unique(nums))
```

Output

```
[1] "Original vector(string)"
[1] "The quick brown fox jumps over the lazy dog."
[1] "Unique elements of the said vector:"
[1] "the quick brown fox jumps over the lazy dog."
[1] "Original vector(number)"
[1] 1 2 2 3 4 4 5 6
[1] "Unique elements of the said vector:"
[1] 1 2 3 4 5 6
```

12. Write a R program to create three vectors a,b,c with 3 integers. Combine the three vectors to become a 3×3 matrix where each column represents a vector. Print the content of the matrix

```
a<-c(1,2,3)
b<-c(4,5,6)
c<-c(7,8,9)
m<-cbind(a,b,c)
print("Content of the said matrix:")
print(m)
```

Output

```
[1] "Content of the said matrix:"
     a b c
[1,] 1 4 7
[2,] 2 5 8
[3,] 3 6 9
```

SUMMARIZING THE STATISTICS

1. **summary(iris)**

Output:

```
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width      Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900       Max.   :4.400       Max.   :6.900       Max.   :2.500
```

2. **str(mtcars)**

Output:

```
'data.frame':    32 obs. of  11 variables:
 $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
 $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
 $ disp: num  160 160 108 258 360 ...
 $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
 $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
 $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
 $ qsec: num  16.5 17 18.6 19.4 17 ...
 $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
 $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
 $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
 $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

3. **head(mtcars)**

Output:

```
                   mpg   cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4          21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag      21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710         22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive     21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout  18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant            18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

4. **tail(mtcars)**

Output:

```
               mpg cyl  disp  hp drat    wt qsec vs am gear carb
Porsche 914-2  26.0   4 120.3  91 4.43 2.140 16.7  0  1    5    2
```

```
Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.9  1  1    5    2
Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.5  0  1    5    4
Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.5  0  1    5    6
Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.6  0  1    5    8
Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.6  1  1    4    2
```

5. **names(mtcars)**

Output:

```
[1] "mpg"  "cyl"  "disp" "hp"   "drat" "wt"   "qsec" "vs"   "am"   "gear" "carb"
```

6. **nrow(mtcars)**

Output:

```
[1] 32
```

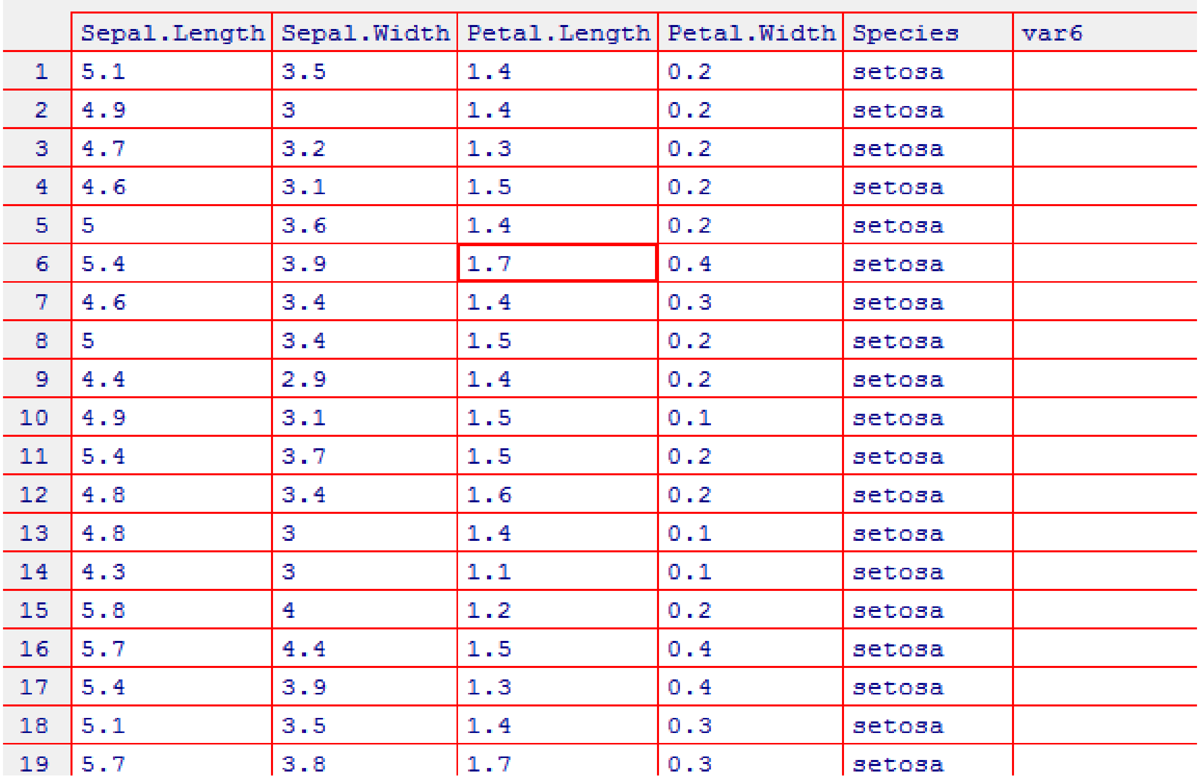
7. **aggregate(Sepal.Length~Species,iris,mean)**

Output:

```
     Species      Sepal.Length
1    setosa           5.006
2    versicolor       5.936
3    virginica        6.588
```

8. **fix(iris)**

Output:



	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species	var6
1	5.1	3.5	1.4	0.2	setosa	
2	4.9	3	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	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	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	
11	5.4	3.7	1.5	0.2	setosa	
12	4.8	3.4	1.6	0.2	setosa	
13	4.8	3	1.4	0.1	setosa	
14	4.3	3	1.1	0.1	setosa	
15	5.8	4	1.2	0.2	setosa	
16	5.7	4.4	1.5	0.4	setosa	
17	5.4	3.9	1.3	0.4	setosa	
18	5.1	3.5	1.4	0.3	setosa	
19	5.7	3.8	1.7	0.3	setosa	

9. **sepalsub<- subset(iris,Sepal.Length>7)**

sepalsub

Output:

```
            Sepal.Length   Sepal.Width   Petal.Length   Petal.Width   Species     var6
103                 7.1           3.0            5.9           2.1     virginica   <N
A>
106                 7.6           3.0            6.6           2.1     virginica   <N
A>
108                 7.3           2.9            6.3           1.8     virginica   <N
A>
110                 7.2           3.6            6.1           2.5     virginica   <N
A>
118                 7.7           3.8            6.7           2.2     virginica   <N
A>
119                 7.7           2.6            6.9           2.3     virginica   <N
A>
123                 7.7           2.8            6.7           2.0     virginica   <N
A>
126                 7.2           3.2            6.0           1.8     virginica   <N
A>
```

130	7.2	3.0	5.8	1.6	virginica	<NA>
131	7.4	2.8	6.1	1.9	virginica	<NA>
132	7.9	3.8	6.4	2.0	virginica	<NA>
136	7.7	3.0	6.1	2.3	virginica	<NA>

BINOMIAL DISTRIBUTION

1. **choose(10,3)*((1/6)^3*(5/6)^7)**

<u>Output:</u>

```
[1] 0.1550454
```

2. **dbinom(3,size=10,prob=(1/6))**

<u>Output:</u>

```
[1] 0.1550454
```

3. **choose(10,0)*((1/6)^0*(5/6)^10) +**

choose(10,1)*((1/6)^1*(5/6)^9)+

choose(10,2)*((1/6)^2*(5/6)^8)+

choose(10,3)*((1/6)^3*(5/6)^7)

<u>Output:</u>

```
[1] 0.9302722
```

4. **pbinom(3,size=10,prob=(1/6),lower=T)**

<u>Output:</u>

```
[1] 0.9302722
```

5. **pbinom(3,size=10,prob=(1/6),lower=F)**

<u>Output:</u>

```
[1] 0.06972784
```

6.
```
dbinom(4,size=12,prob=0.2)
```

Output:

```
[1] 0.1328756
```

7.
```
dbinom(0,size=12,prob=0.2)+
   dbinom(1,size=12,prob=0.2)+
   dbinom(2,size=12,prob=0.2)+
   dbinom(3,size=12,prob=0.2)+
   dbinom(4,size=12,prob=0.2)
```

Output:

```
[1] 0.9274445
```

8.
```
pbinom(4,size=12,prob=0.2)
```

Output:

```
[1] 0.9274445
```

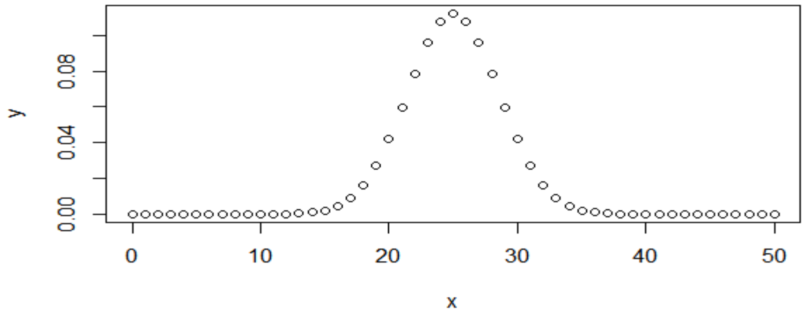
9.
```
x <- pbinom(26,51,0.5)
     print(x)
```

Output:

```
[1] 0.610116
```

10.
```
x <- seq(0,50,by = 1)
    y <- dbinom(x,50,0.5)
    plot(x,y)
```

Output:



11. **x <- qbinom(0.25,51,1/2)**

print(x)

<u>Output:</u>

```
[1] 23
```

12. **x <- rbinom(8,150,.4)**

print(x)

<u>Output:</u>

```
[1] 60 71 57 60 62 62 50 59
```

<u>POISSON DISTRIBUTION</u>

1. **ppois(16,lambda = 12)**

<u>Output:</u>

```
[1] 0.898709
```

2. **ppois(16,lambda = 12,lower=F)**

<u>Output:</u>

```
[1] 0.101291
```

3. **rpois(16,lambda = 12)**

<u>Output:</u>

```
[1] 12  8 12 10 13  8 11 13 12 11 15 12 10 11 14 11
```

4. **dpois(16,lambda = 12)**

<u>Output:</u>

```
[1] 0.05429334
```

NORMAL DISTRIBUTION

1. **pnorm(84,mean=72,sd=15.2,lower.tail=FALSE)**

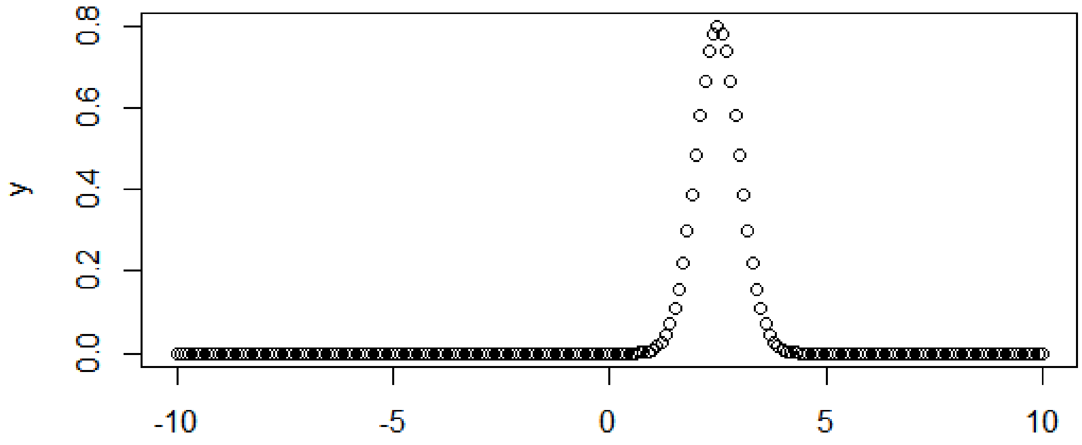
Output:

```
[1] 0.2149176
```

2**. x <- seq(-10, 10, by = .1)**

```
y <- dnorm(x, mean = 2.5, sd = 0.5)
plot(x,y)
```

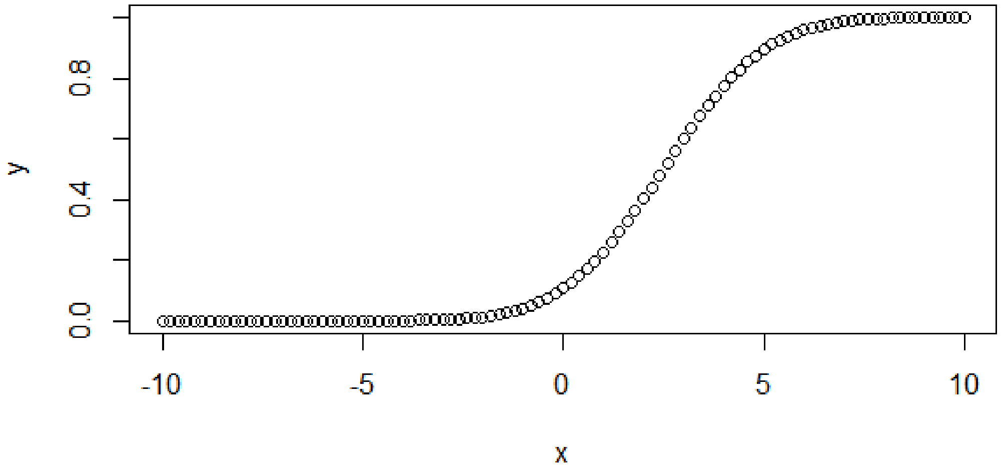
Output:



3. **x <- seq(-10,10,by = .2)**

```
y <- pnorm(x, mean = 2.5, sd = 2)
plot(x,y)
```

Output:

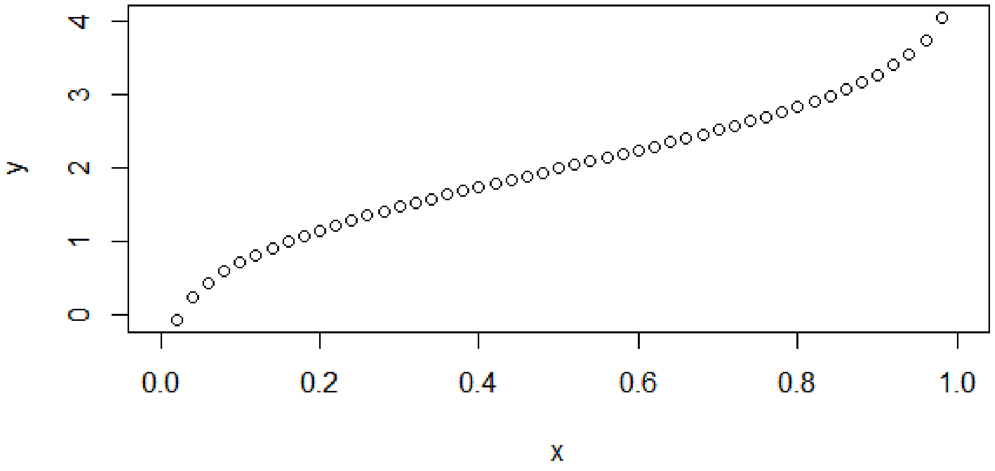


4. **x <- seq(0, 1, by = 0.02)**

y <- qnorm(x, mean = 2, sd = 1)

plot(x,y)

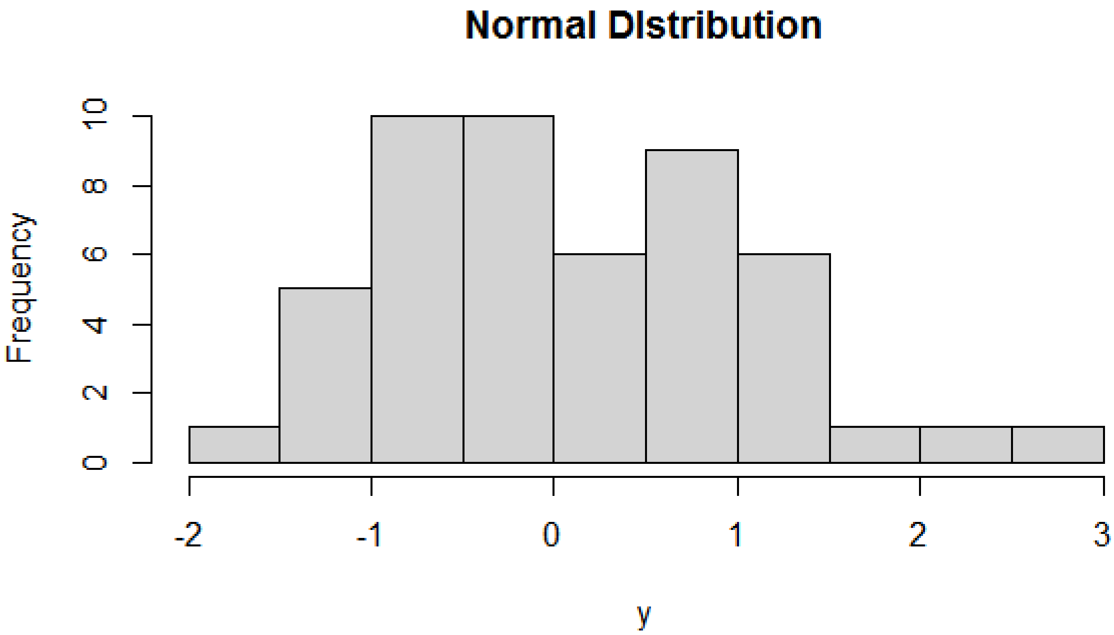
<u>Output:</u>



5. **y <- rnorm(50)**

hist(y, main = "Normal DIstribution")

<u>Output:</u>



LINEAR REGRSSION

Write a program to implement linear and logistic regression

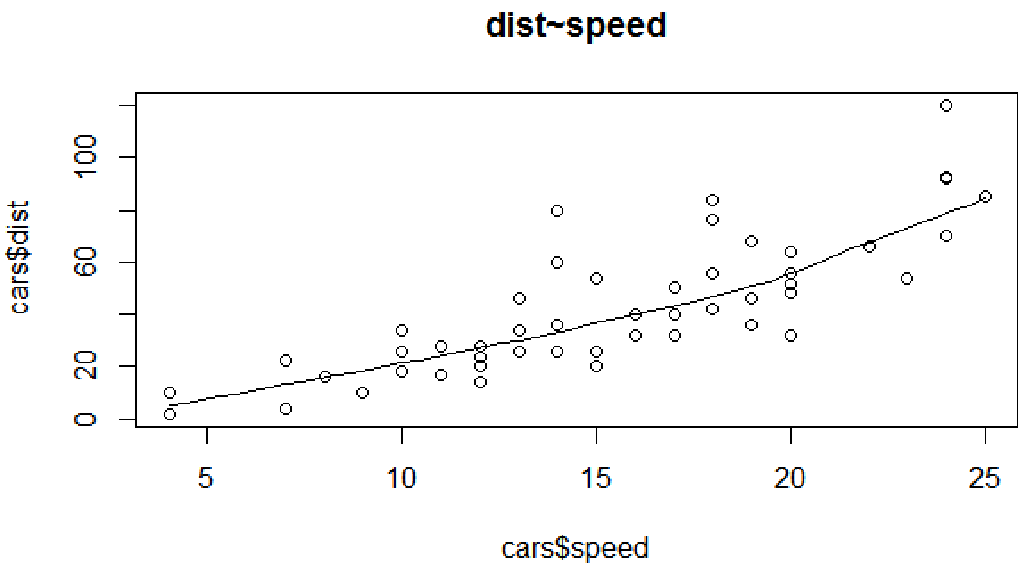
```
head(cars)
```

Output:

```
   speed   dist
1    4       2
2    4      10
3    7       4
4    7      22
5    8      16
6    9      10
```

```
scatter.smooth(x=cars$speed,y=cars$dist,main="dist~speed")
```

Output:



```
linearMod<-lm(dist~speed,data=cars)
   print(linearMod)
```

Output:

```
Coefficients:
(Intercept)       speed
   -17.579       3.932
```

```
summary(linearMod)
```

Output:

```
Residuals:
   Min       1Q      Median     3Q      Max
-29.069   -9.525    -2.272     9.215   43.201
Coefficients:
                      Estimate      Std. Error      t value      Pr(>|t|)
(Intercept)       -17.5791       6.7584        -2.601        0.0123 *
speed                3.9324         0.4155         9.464         1.49e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Residual standard error: 15.38 on 48 degrees of freedom
Multiple R-squared:  0.6511,  Adjusted R-squared:  0.6438
F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

LOGISTIC REGRESSION

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

Output:

```
                                  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
Duster 360                     0   8 245 3.570
Merc 240D                      0   4  62 3.190
Merc 230                        0   4  95 3.150
Merc 280                        0   6 123 3.440
Merc 280C           0   6 123 3.440
Merc 450SE          0   8 180 4.070
Merc 450SL          0   8 180 3.730
```

```
 input<-mtcars[,c("am","cyl","hp","wt")]
   am.data=glm(formula=am~cyl+hp+wt,data=input,family=binomial)
   print(summary(am.data))
```

<u>Output:</u>

```
Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.17272  -0.14907  -0.01464   0.14116   1.27641
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
```