# Practical 3

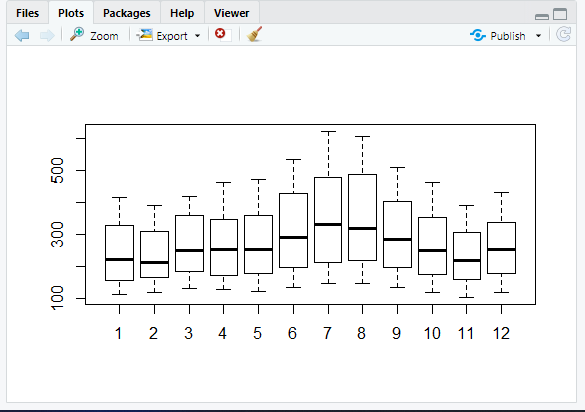
Simple/multiple linear regression

|  |
| --- |
| height <- c(102, 117, 105, 141,135,115, 138, 144, 157, 100, 131,119, 115,121, 113)  weight <- c(61, 46, 62, 54, 60, 69, 51, 50, 46, 64, 48, 56, 64, 48,59)  student <- lm(weight~height)  student  predict(student, data, frame(height=199), interval="confidence")  plot(student) |

# Practical 4

Time-series forecasting

|  |
| --- |
| data('AirPassengers')  class(AirPassengers)  start(AirPassengers)  end(AirPassengers)  frequency(AirPassengers)  summary(AirPassengers)  plot(AirPassengers)  abline(reg=lm(AirPassengers~time(AirPassengers)))  cycle(AirPassengers)  plot(aggregate(AirPassengers, FUN=mean))  boxplot(AirPassengers ~ cycle(AirPassengers)) |



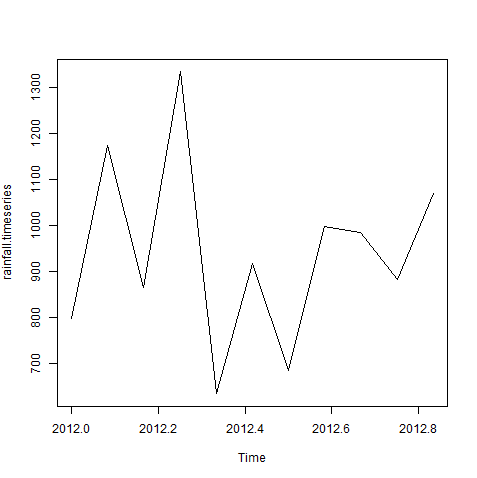
# Practical 5

Time series analysis

|  |
| --- |
| rainfall <- c(799, 1174, 865.1, 1334.6, 635.4, 918.5, 685.5, 998.6, 985, 882.8, 1071)  rainfall.timeseries <- ts(rainfall, start = c(2012, 1), frequency = 12)  print(rainfall.timeseries)  png(file="rainfall.png")  plot(rainfall.timeseries)  dev.off() |

Output:

Files-> rainfall.png

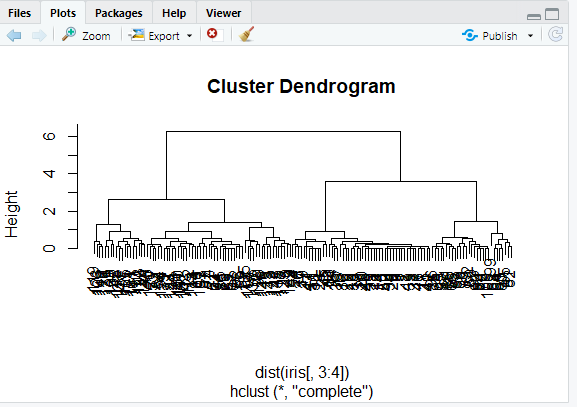


# Practical -6

K-means clustering

|  |
| --- |
| data(iris)  names(iris)  new\_data<-subset(iris,select=c(-Species))  new\_data  c1<-kmeans(new\_data,3)  c1  data<-new\_data  wss<-sapply(1:15,function(k){kmeans(data,k)$tot.withinss})  wss  plot(1:15,wss,type="b",pch=19,frame=FALSE,xlab="Number of clusters K",ylab="Total within-clusters sums of squares")  library(cluster)  clusplot(new\_data,c1$cluster,color=TRUE,shade=TRUE,labels=2,lines=0)  c1$cluster  c1$centers  "agglomarative clustring"  clusters<-hclust(dist(iris[,3:4]))  plot(clusters)  clusterCut<-cutree(cluster,3)  table(clusterCut,iris$Species) |

Ouput:-



# Practical 7

Decision tree

|  |
| --- |
| insatll.packages("party")  library(party)  input.dat<-readingSkills[c(1:15),]  png(file="decision\_tree.png")  output.tree<-ctree(nativeSpeaker~age+shoeSize+score,data=input.dat)  plot(output.tree)  dev.off() |

# Practical 8

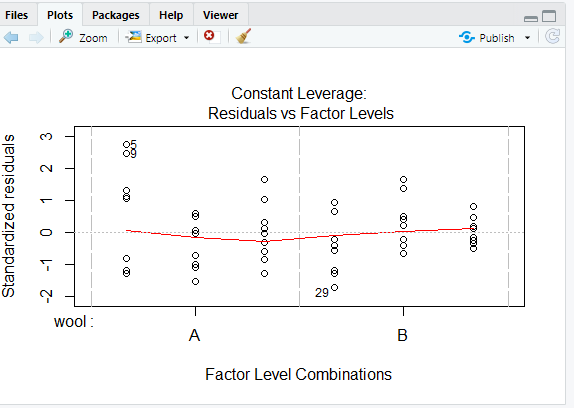
Hypothesis testing

|  |
| --- |
| dataf<-seq(1,20,by=1)  dataf  mean(dataf)  sd(dataf)  a<-t.test(dataf,alternate="two.sided",mu=10,conf.int=0.95)  a  a$p.value  a$statistic  (10.5-10)/(sd(dataf)/sqrt(length(dataf)))  length(dataf)=1  length(dataf)  dataf  dataf<-seq(1,20,by=1)  length(dataf)-1 |

# Practical -9

Analysis of variance

|  |
| --- |
| data("warpbreaks")  head(warpbreaks)  summary(warpbreaks)  Model\_1<-aov(breaks~wool+tension,data=warpbreaks)  summary(Model\_1)  plot(Model\_1)  Model\_2<-aov(breaks~wool+tension+wool:tension,data=warpbreaks)  summary(Model\_2)  plot(Model\_2) |



Wireshark

Wireshark is a network protocol analyzer, or an application that captures packets from a network connection, such as from your computer to your home office or the internet. Packet is the name given to a discrete unit of data in a typical Ethernet network. Wireshark is the most often-used packet sniffer in the world.

Wireshark helps:

Network administrators troubleshoot problems across a network.

Security engineers examine security issues across a network.

QA engineers verify applications.

Developers debug protocol implementations.

Network users learn about a specific protocol.

**Features**

* Deep inspection of hundreds of protocols, with more being added all the time.
* Live capture and offline analysis.
* Standard three-pane packet browser.
* Multi-platform: Runs on Windows, Linux, OS X, FreeBSD, NetBSD, and many others.
* Captured network data can be browsed via a GUI, or via the TTY-mode TShark utility