

VIT-AP UNIVERSITY, ANDHRA PRADESH

**CSE2047 – Data Analytics - Lab Sheet : 9**

**Academic year:** 2020-2021

**Semester:** Fall

**Faculty Name:** Prof. S.Gopikrishnan

**Student name:** Valiveti Manikanta bhuvanesh

**Branch/ Class:** B.Tech/M.Tech

**Date:**

**School:** SCOPE

**Reg. no.:** 19BCD7088

---

**LAB 9**

1. Read the “sample.txt” text file in R. Print the number of characters, number of digits, number of symbols and number of words in the word file.

```
df<-read.delim("sample.txt",header = T, sep = " ")
df
nchar(df,allowNA = FALSE, keepNA = NA)
counti=0
countc=0
sapply(df, class)
for (i in names(df)) {
  for (j in df[,i]) {
    if(is.integer(j)==TRUE){
      counti=counti+1 }
    if(is.character(j)==TRUE){
      countc=countc+1
    }
  }
}
counti
countc
```

```

> df<-read.delim("sample.txt",header = T, sep = " ")
> df
      name age
1 bhuvanesh 21
2      guru 19
3     manas 10
> nchar(df,allowNA = FALSE, keepNA = NA)
name age
  31  13
> counti=0
> countc=0
> sapply(df, class)
      name      age
"character" "integer"
> for (i in names(df)) {
+   for (j in df[,i]) {
+     if(is.integer(j)==TRUE){
+       counti=counti+1 }
+     if(is.character(j)==TRUE){
+       countc=countc+1
+     }
+   }
+ }
> counti
[1] 3
> countc
[1] 3

```

## 2. Append a new row to the “sample.txt” text file.

```

my_data<-"ramesh 22"
write.table(my_data, file = "sample.txt", sep=" ",append = T, quote = F,
col.names= T, row.names= F)
> my_data<-"ramesh 22"
> write.table(my_data, file = "sample.txt", sep= " ",append = T, quote = F, col.names= T, row.names= F)
Warning message:
In write.table(my_data, file = "sample.txt", sep = " ", append = T, :
  appending column names to file

```

## 3. How to read this text file with missing values?

```

df<-read.delim("sample.txt",header = F,na.strings = c("", 'NA', '<NA>'))
df
      V1
1      name age
2 1 bhuvanesh 21
3      2 guru 19
4      3 manas 10
5      x
6      ramesh 22

```

**4. Read the Iris dataset from csv file and write into a xlsx file in R.**

```
write.xlsx(iris,file="iris.xlsx",col.names=T,row.names=T,sheetName="Sheet1")
> write.xlsx(iris,file="iris.xlsx",col.names=T,row.names=T,sheetName="Sheet1")
>
```

**5. Write the covid data set csv file dataset in the second sheet of the xlsx file created for the question2.**

```
df2<-read_csv("COVID_country_wise_latest.csv")
write.xlsx(df2,file="iris.xlsx",col.names=T,row.names=T,sheetName="Sheet2",
append=TRUE)
> df2<-read_csv("COVID_country_wise_latest.csv")
Rows: 187 Columns: 15
-- Column specification -----
Delimiter: ","
chr (2): Country/Region, WHO Region
dbl (13): Confirmed, Deaths, Recovered, Active, New cases, New deaths, New recovere...

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
> write.xlsx(df2,file="iris.xlsx",col.names=T,row.names=T,sheetName="Sheet2",append=TRUE)
>
```

**6. Differentiate scan() and read\_table using Iris data set.**

```
write.table(iris,file = "data.txt",row.names = FALSE)
scan("data.txt",what = "character")
read_table("data.txt")
> write.table(iris,file = "data.txt",row.names = FALSE)
> scan("data.txt",what = "character")
Read 755 items
 [1] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width" "Species"
 [6] "5.1" "3.5" "1.4" "0.2" "setosa"
[11] "4.9" "3" "1.4" "0.2" "setosa"
[16] "4.7" "3.2" "1.3" "0.2" "setosa"
[21] "4.6" "3.1" "1.5" "0.2" "setosa"
[26] "5" "3.6" "1.4" "0.2" "setosa"
[31] "5.4" "3.9" "1.7" "0.4" "setosa"
[36] "4.6" "3.4" "1.4" "0.3" "setosa"
[41] "5" "3.4" "1.5" "0.2" "setosa"
[46] "4.4" "2.9" "1.4" "0.2" "setosa"
[51] "4.9" "3.1" "1.5" "0.1" "setosa"
[56] "5.4" "3.7" "1.5" "0.2" "setosa"
[61] "4.8" "3.4" "1.6" "0.2" "setosa"
[66] "4.8" "3" "1.4" "0.1" "setosa"
[71] "4.3" "3" "1.1" "0.1" "setosa"
[76] "5.8" "4" "1.2" "0.2" "setosa"
[81] "5.7" "4.4" "1.5" "0.4" "setosa"
[86] "5.4" "3.9" "1.3" "0.4" "setosa"
[91] "5.1" "3.5" "1.4" "0.3" "setosa"
[96] "5.7" "3.8" "1.7" "0.3" "setosa"
[101] "5.1" "3.8" "1.5" "0.3" "setosa"
[106] "5.4" "3.4" "1.7" "0.2" "setosa"
[111] "5.1" "3.7" "1.5" "0.4" "setosa"
[116] "4.6" "3.6" "1" "0.2" "setosa"
[121] "5.1" "3.3" "1.7" "0.5" "setosa"
```

```
> read_table("data.txt")

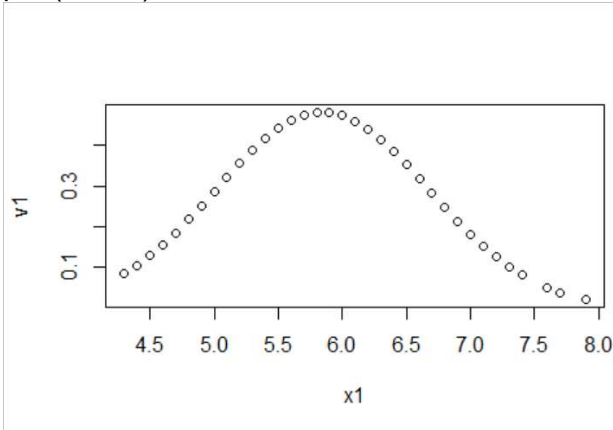
-- Column specification -----
cols(
  `Sepal.Length` = col_double(),
  `Sepal.Width` = col_double(),
  `Petal.Length` = col_double(),
  `Petal.Width` = col_double(),
  `Species` = col_character()
)

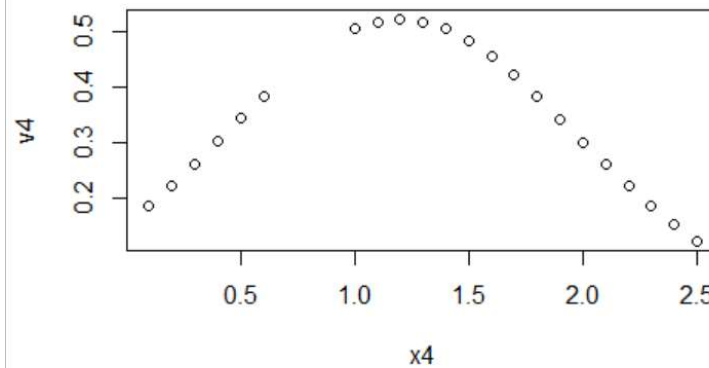
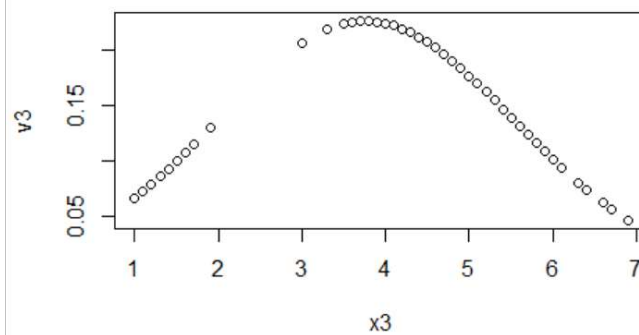
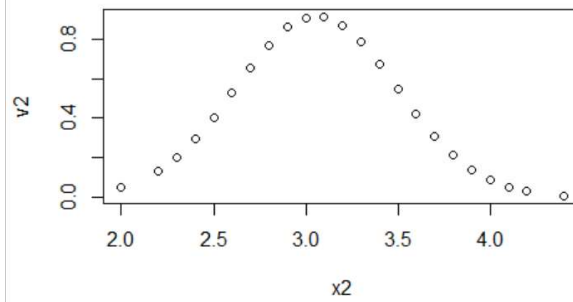
# A tibble: 150 x 5
  `Sepal.Length` `Sepal.Width` `Petal.Length` `Petal.Width` `Species`
      <dbl>         <dbl>         <dbl>         <dbl>      <chr>
1         5.1         3.5         1.4         0.2  "\"setosa\""
2         4.9         3         1.4         0.2  "\"setosa\""
3         4.7         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\""
# ... with 140 more rows
```

## 7. Use iris dataset and plot the normal distribution on all the numerical columns

```
x1 <- iris$Sepal.Length
x2 <- iris$Sepal.Width
x3 <- iris$Petal.Length
x4 <- iris$Petal.Width

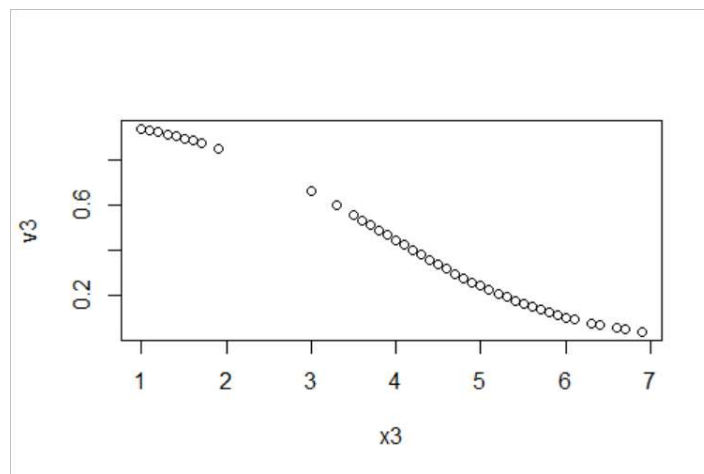
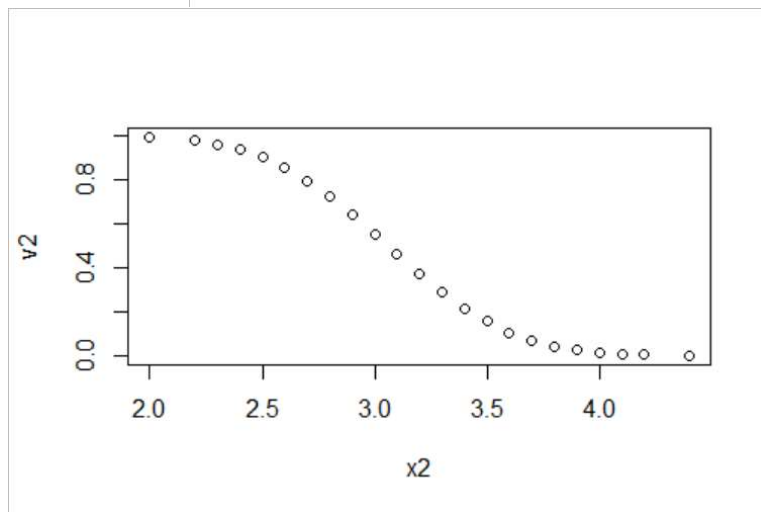
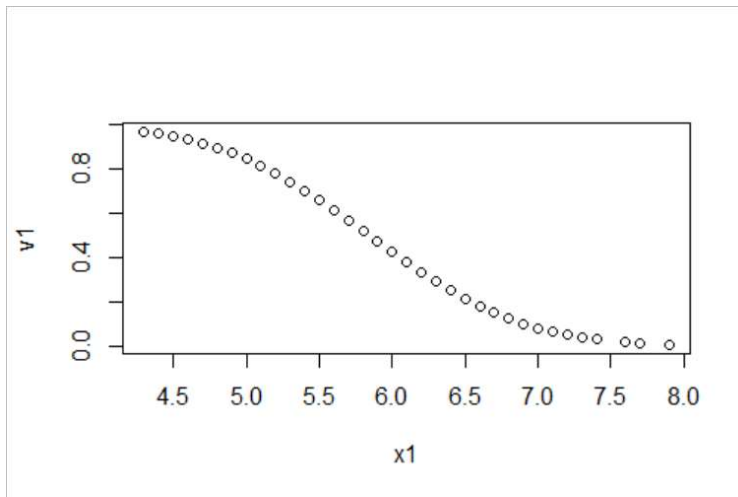
a. dnorm()
v1 <- dnorm(x1, mean = mean(x1), sd = sd(x1))
plot(x1, v1)
v2 <- dnorm(x2, mean = mean(x2), sd = sd(x2))
plot(x2, v2)
v3 <- dnorm(x3, mean = mean(x3), sd = sd(x3))
plot(x3, v3)
v4 <- dnorm(x4, mean = mean(x4), sd = sd(x4))
plot(x4, v4)
```

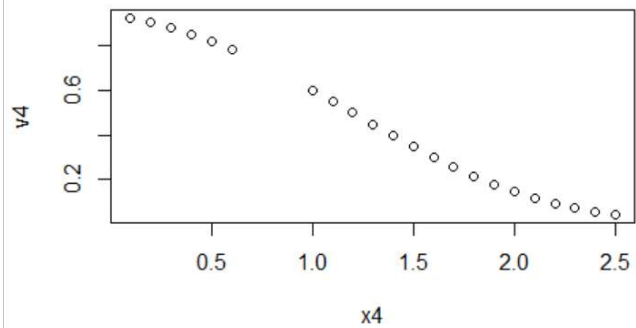




**b. pnorm()**

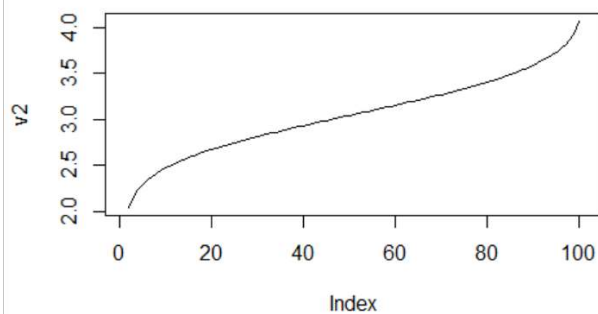
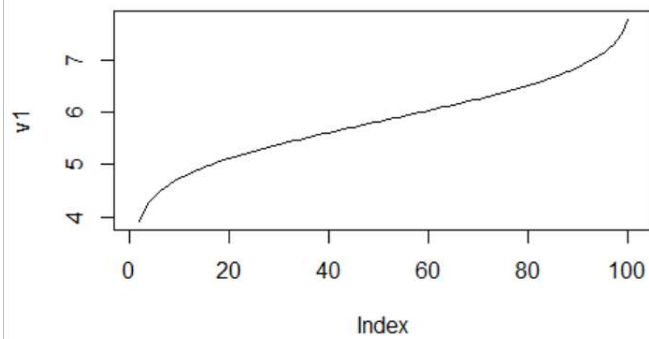
```
v1 <- pnorm(x1, mean = mean(x1), sd = sd(x1), lower.tail = FALSE)
plot(x1, v1)
v2 <- pnorm(x2, mean = mean(x2), sd = sd(x2), lower.tail = FALSE)
plot(x2, v2)
v3 <- pnorm(x3, mean = mean(x3), sd = sd(x3), lower.tail = FALSE)
plot(x3, v3)
v4 <- pnorm(x4, mean = mean(x4), sd = sd(x4), lower.tail = FALSE)
plot(x4, v4)
```

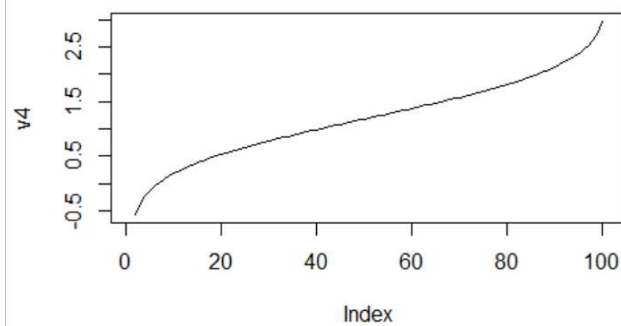
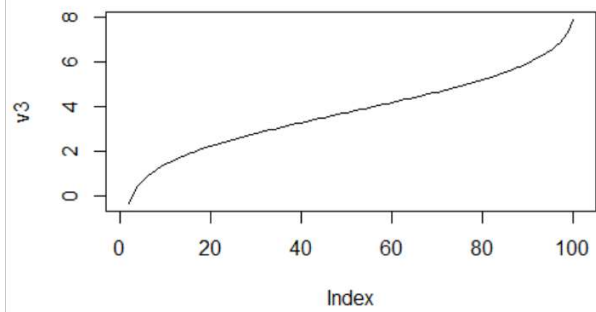




**c. qnorm()**

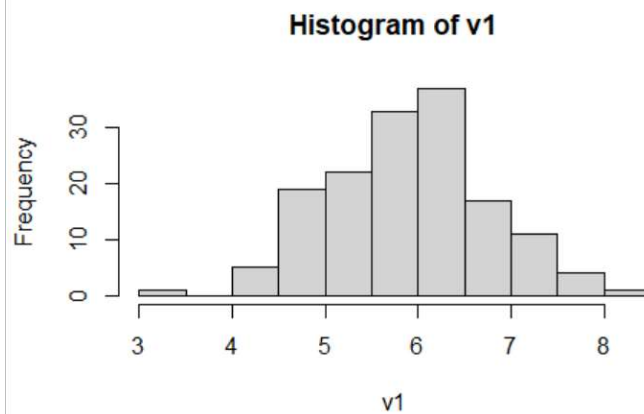
```
x <- seq(0, 1, by = 0.01)
v1 <- qnorm(x, mean = mean(x1), sd = sd(x1))
plot(v1,type = "l")
v2 <- qnorm(x, mean = mean(x2), sd = sd(x2))
plot(v2,type = "l")
v3 <- qnorm(x, mean = mean(x3), sd = sd(x3))
plot(v3,type = "l")
v4 <- qnorm(x, mean = mean(x4), sd = sd(x4))
plot(v4,type = "l")
```





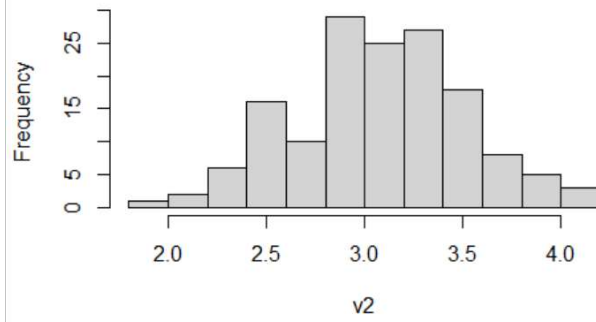
**d. rnorm()**

```
v1 <- rnorm(x1, mean = mean(x1), sd = sd(x1))  
hist(v1)  
v2 <- rnorm(x2, mean = mean(x2), sd = sd(x2))  
hist(v2)  
v3 <- rnorm(x3, mean = mean(x3), sd = sd(x3))  
hist(v3)  
v4 <- rnorm(x4, mean = mean(x4), sd = sd(x4))  
hist(v4)
```

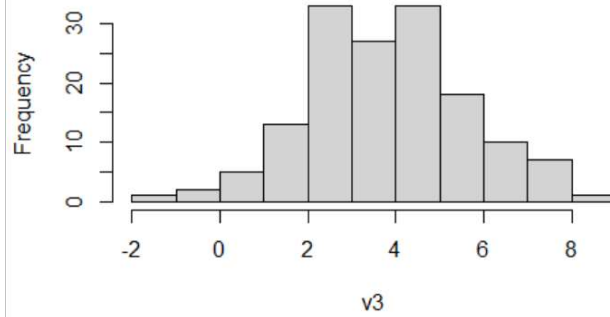




**Histogram of v2**



**Histogram of v3**



**Histogram of v4**

