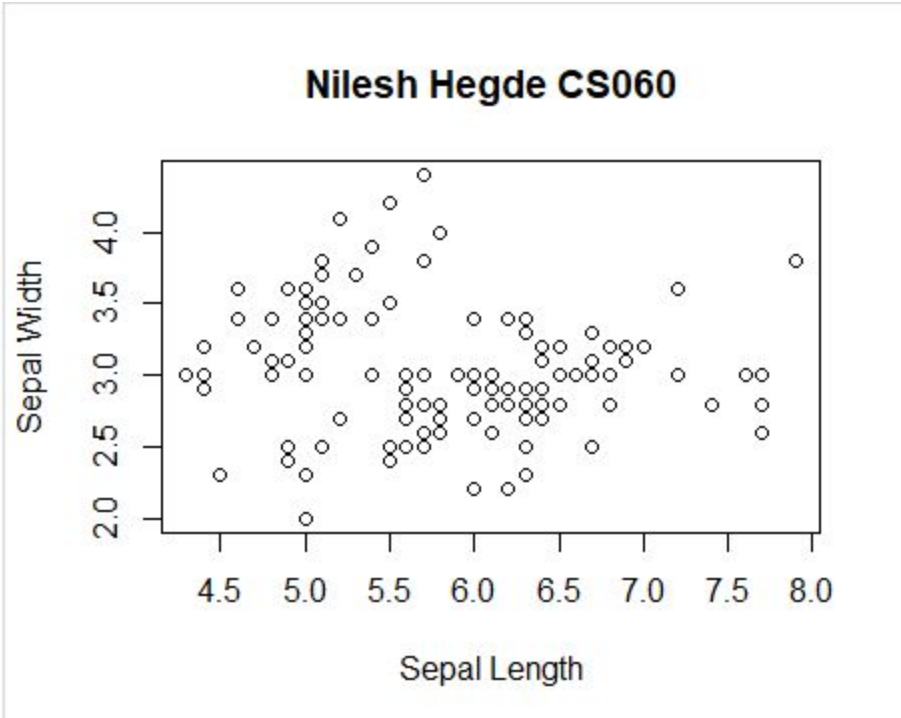
Program 1 a

```
> path="C:\\Users\\niles\\Desktop\\DSR-Lab"
> setwd(path)
> dataval=read.csv("petals.csv")
> summary(dataval)
  sepal.length    sepal.width     petal.length    petal.width          variety
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   Setosa    :41
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   Versicolor:44
 Median :5.800   Median :3.000   Median :4.400   Median :1.300   Virginica :42
 Mean   :5.843   Mean   :3.062   Mean   :3.769   Mean   :1.213
 3rd Qu.:6.400   3rd Qu.:3.400   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
> plot(dataval$sepal.length,dataval$sepal.width,xlab = "Sepal Length",ylab = "Sepal Width",main="Nilesh Hegde
 CS060")
> print("Nilesh Hegde 060")
[1] "Nilesh Hegde 060"
```



Program 1 b

```
> df2 = read.delim("C:\\Users\\niles\\Desktop\\DSR-Lab\\data.csv")
> df2
   play.outlook.temp.humidity.wind
1       yes,rainy,cool,normal,FALSE
2         no,rainy,cool,normal,TRUE
3       yes,overcast,hot,high,FALSE
4          no,sunny,mild,high,FALSE
5       yes,rainy,cool,normal,FALSE
6       yes,sunny,cool,normal,FALSE
7       yes,rainy,cool,normal,FALSE
8        yes,sunny,hot,normal,FALSE
9       yes,overcast,mild,high,TRUE
10         no,sunny,mild,high,TRUE
> str(df2)
'data.frame':   10 obs. of  1 variable:
 $ play.outlook.temp.humidity.wind: Factor w/ 8 levels "no,rainy,cool,normal,TRUE",..: 6 1 4 2 6 7 6 8 5 3
> v <- c('abc','def','ghi','jkl','mno')
> df2$description<-v
> write.csv(df2, file="modifiedfile2.csv")
> print("Nilesh Hegde CS060")
[1] "Nilesh Hegde CS060"
>
```

Program 2

```
> bufftail<-c(10,1,37,5,12)
> gardenbee<-c(8,3,9,6,4)
> redtail<-c(18,9,1,2,4)
> carderbee<-c(8,27,6,32,23)
> honeybee<-c(12,13,16,9,10)
> rowname<-c("Thristle","Vipers","Golden_Rain"
+            ,"Yellowalfala","Blackberry")
> bees<-data.frame(
+   row.names=rowname,
+   bufftail,gardenbee,redtail,carderbee,honeybee)
> bees
             bufftail gardenbee redtail carderbee honeybee
Thristle           10         8      18         8        12
Vipers              1         3       9        27        13
Golden_Rain        37         9       1         6        16
Yellowalfala        5         6       2        32         9
Blackberry         12         4       4        23        10
> print("Nilesh Hegde CS060")
[1] "Nilesh Hegde CS060"
>
```

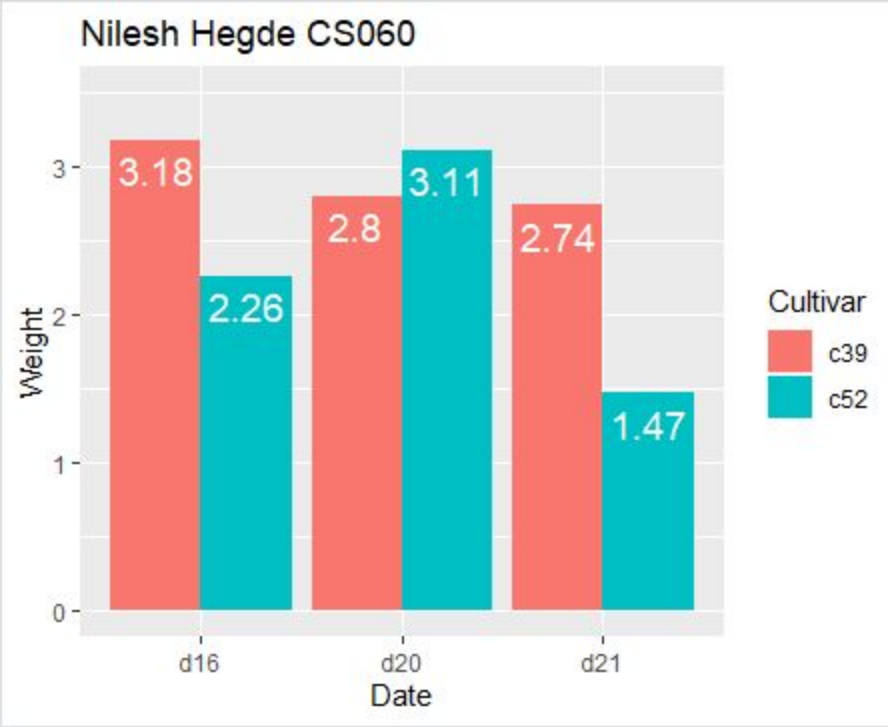
Program 3 a

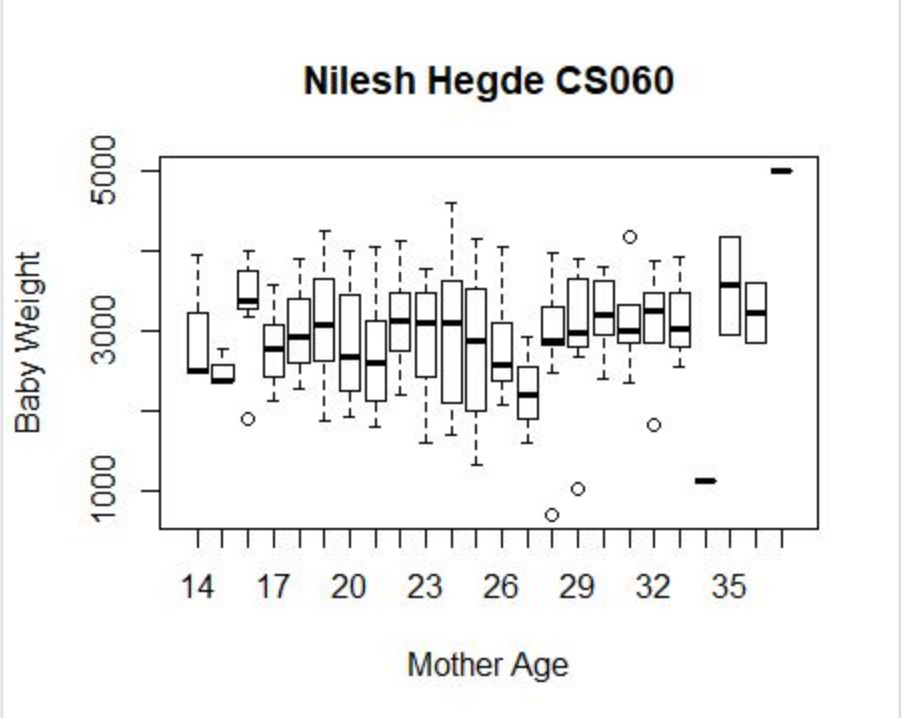
```
> print("3 a)")
[1] "3 a)"
> beesvector<-c(bufftail,gardenbee,redtail,
+               carderbee,honeybee)
> mat=matrix(beesvector,5,5,byrow = TRUE)
> mat
     [,1] [,2] [,3] [,4] [,5]
[1,]   10    1   37    5   12
[2,]    8    3    9    6    4
[3,]   18    9    1    2    4
[4,]    8   27    6   32   23
[5,]   12   13   16    9   10
> list_bees<-list(rowname,mat)
> list_bees
[[1]]
[1] "Thristle"     "Vipers"       "Golden_Rain"  "Yellowalfala" "Blackberry"

[[2]]
     [,1] [,2] [,3] [,4] [,5]
[1,]   10    1   37    5   12
[2,]    8    3    9    6    4
[3,]   18    9    1    2    4
[4,]    8   27    6   32   23
[5,]   12   13   16    9   10

> print("Nilesh Hegde CS060")
[1] "Nilesh Hegde CS060"
```

Program 3 b

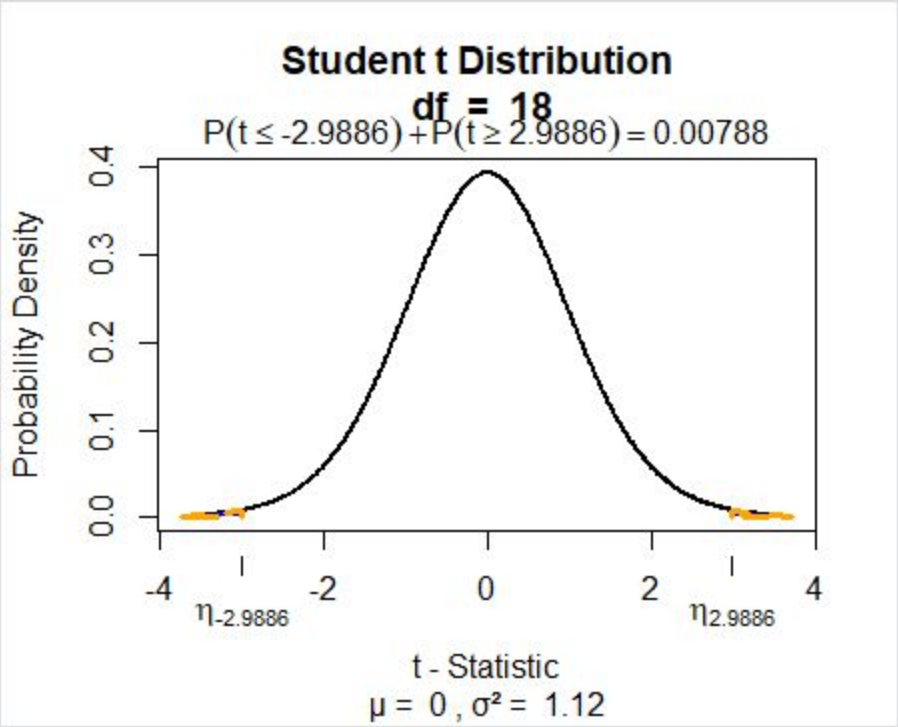
```
> print("3 b)")
[1] "3 b)"
> list_ex<-list("Numeric"=c(1,2,3),
+               "String"="This is a string",
+               "Real"=c(3.21,4.2))
> list_actual<-list(c(1,2,3),"String type",
+                   c(3.1,3.14))
> names(list_actual)<-c("Numeric","String","Real")
> list_actual
$Numeric
[1] 1 2 3

$String
[1] "String type"

$Real
[1] 3.10 3.14

> print("Nilesh Hegde CS060")
[1] "Nilesh Hegde CS060"
>
```

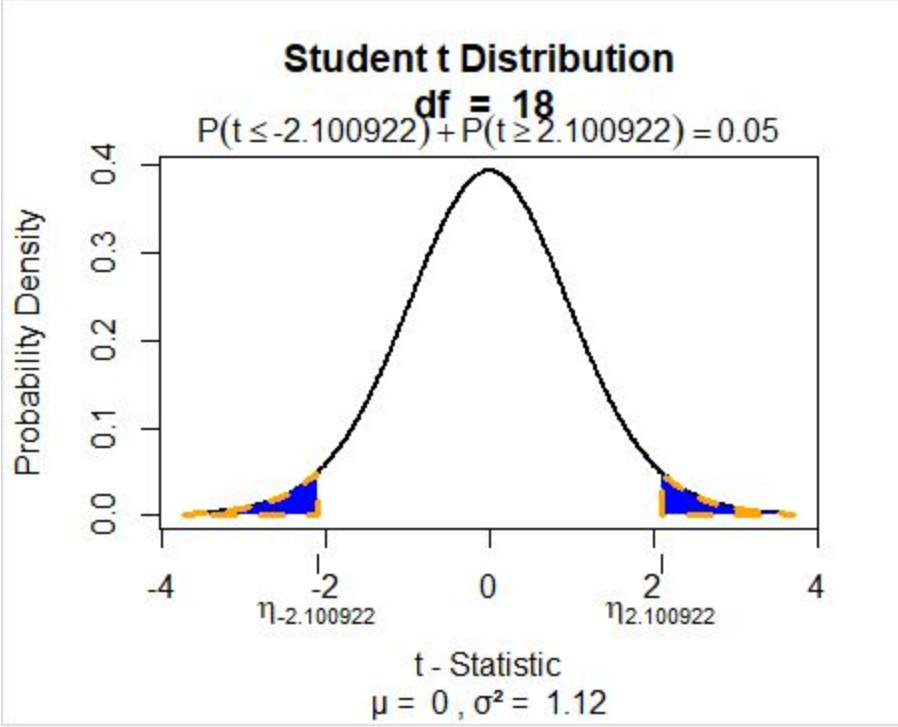
Program 4


Nilesh Hegde CS060
3.18
2.26
2.8
3.11
2.74
1.47
Weight
0
1
2
3
d16
d20
d21
Date
Cultivar
c39
c52


Nilesh Hegde CS060
Baby Weight
1000
3000
5000
14
17
20
23
26
29
32
35
Mother Age

Program 5

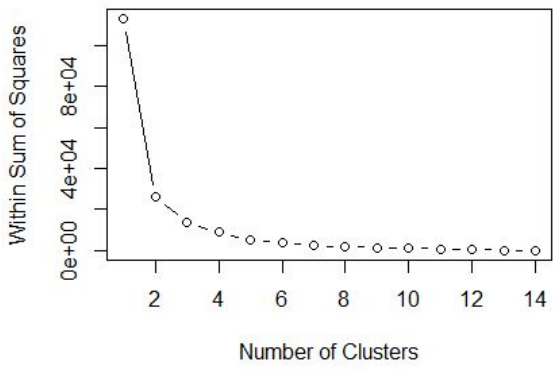

Student t Distribution
df = 18
P(t ≤ -2.9886) + P(t ≥ 2.9886) = 0.00788
Probability Density
0.0
0.1
0.2
0.3
0.4
-4
-2
0
2
4
η-2.9886
η2.9886
t - Statistic
μ = 0 , σ² = 1.12


Student t Distribution
df = 18
P(t ≤ -2.100922) + P(t ≥ 2.100922) = 0.05
Probability Density
0.0
0.1
0.2
0.3
0.4
-4
-2
0
2
4
η-2.100922
η2.100922
t - Statistic
μ = 0 , σ² = 1.12

```
> library(BSDA)
> rural<-c(3.1,2.9,2.8,3.0,2.7,3.1,2.6,2.8,2.9,3.0)
> urban<-c(3.5,3.0,3.1,3.2,2.9,3.4,3.0,3.4,2.8,3.4)
> xrbar=mean(rural)
> xrbar
[1] 2.89
> xurbar=mean(urban)
> xurbar
[1] 3.17
> var(rural)
[1] 0.02766667
> sd(rural)
[1] 0.166333
> var(urban)
[1] 0.06011111
> sd(urban)
[1] 0.2451757
> t.test(x=rural,y=urban,var.equal = TRUE,conf.level = 0.95)

        Two Sample t-test

data:  rural and urban
t = -2.9886, df = 18, p-value = 0.007878
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.47683496 -0.08316504
sample estimates:
mean of x mean of y
     2.89      3.17

> qt(p=0.05/2,df=18,lower.tail = FALSE)
[1] 2.100922
> visualize.t(stat=c(-2.9886,2.9886),df=18,section="tails")
> visualize.t(stat=c(-2.100922,2.100922),df=18,section="tails")
> print("Nilesh Hegde CS060")
[1] "Nilesh Hegde CS060"
>
```

Program 6

Nilesh Hegde CS060



Program 7

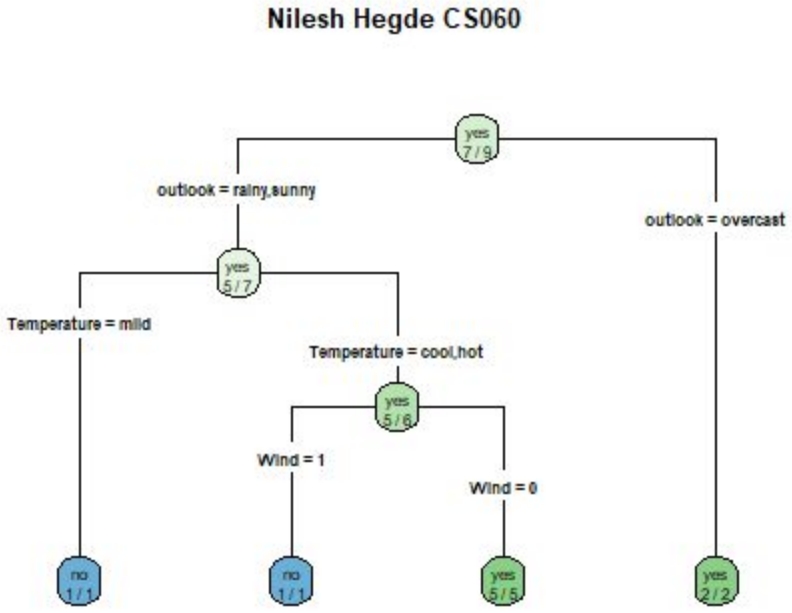
```
> fw.lm=lm(count ~ speed,data=fw)
> summary(fw.lm)

Call:
lm(formula = count ~ speed, data = fw)

Residuals:
    Min      1Q  Median      3Q     Max
-13.377  -5.801  -1.542   5.051  14.371

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   8.2546     5.8531   1.410   0.2081
speed         0.7914     0.3081   2.569   0.0424 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.16 on 6 degrees of freedom
Multiple R-squared:  0.5238,    Adjusted R-squared:  0.4444
F-statistic: 6.599 on 1 and 6 DF,  p-value: 0.0424

> names(fw.lm)
 [1] "coefficients"  "residuals"     "effects"       "rank"          "fitted.values" "assign"
 [7] "qr"            "df.residual"   "xlevels"       "call"          "terms"         "model"
> fw.lm$coefficients
(Intercept)       speed
  8.2545956   0.7913603
> newypred<-fitted(fw.lm)
> newypred
      Taw  Torridge      Ouse       Exe       Lyn     Brook     Ditch       Fal
 9.837316 10.628676 12.211397 15.376838 19.333640 27.247243 31.204044 35.160846
> confint(fw.lm)
                2.5 %    97.5 %
(Intercept) -6.06752547 22.576717
speed        0.03756445  1.545156

> confint(fw.lm,parm=c('(Intercept)','speed'),level =0.9)
                 5 %      95 %
(Intercept) -3.119113 19.628305
speed        0.192744  1.389977
> fitted(fw.lm)
      Taw  Torridge      Ouse       Exe       Lyn     Brook     Ditch       Fal
 9.837316 10.628676 12.211397 15.376838 19.333640 27.247243 31.204044 35.160846
> residuals(fw.lm)
        Taw    Torridge        Ouse         Exe         Lyn       Brook       Ditch         Fal
 -0.8373162  14.3713235   2.7886029 -13.3768382  -5.3336397  -2.2472426  -7.2040441  11.8391544
> plot(fw$speed,fw$count,col="red",xlab="Speed",ylab="Count",main = "Nilesh Hegde CS060")
> coef(fw.lm)
(Intercept)       speed
  8.2545956   0.7913603
> abline(coef(fw.lm),lty=1,col="blue")
> plot(fw.lm,which=1,main = "Nilesh Hegde CS060")
> print("Nilesh Hegde CS060")
[1] "Nilesh Hegde CS060"
```

Program 8

```
> bayesmodel <- naiveBayes(Enrolls ~ Age+Income+JobSatisfaction+Desire,traindata)
> bayesmodel

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
       No       Yes
0.3571429 0.6428571 0.0000000

Conditional probabilities:
     Age
Y          <=30       >40  31 to 40  35 to 40
  No  0.6000000 0.4000000 0.0000000 0.0000000
  Yes 0.2222222 0.3333333 0.3333333 0.1111111


     Income
Y          High       Low    Medium
  No  0.4000000 0.2000000 0.4000000
  Yes 0.2222222 0.3333333 0.4444444


     JobSatisfaction
Y            No       Yes
  No  0.8000000 0.2000000
  Yes 0.3333333 0.6666667


     Desire
Y     Excellent      Fair
  No  0.6000000 0.4000000
  Yes 0.3333333 0.6666667
```

```
> results <- predict (bayesmodel,testdata)
> results
[1] No
Levels: No Yes
> model1 = naiveBayes(Enrolls ~ ., traindata, laplace=.01)
> model1

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
       No       Yes
0.3571429 0.6428571 0.0000000


Conditional probabilities:
     Age
Y            <=30         >40    31 to 40    35 to 40
  No  0.597222222 0.398809524 0.001984127 0.001984127
  Yes 0.222345133 0.332964602 0.332964602 0.111725664
      0.250000000 0.250000000 0.250000000 0.250000000

     Income
Y          High       Low    Medium
  No  0.3996024 0.2007952 0.3996024
  Yes 0.2225914 0.3333333 0.4440753
      0.3333333 0.3333333 0.3333333

     JobSatisfaction
Y            No       Yes
  No  0.7988048 0.2011952
  Yes 0.3337029 0.6662971
      0.5000000 0.5000000

     Desire
Y     Excellent      Fair
  No  0.5996016 0.4003984
  Yes 0.3337029 0.6662971
      0.5000000 0.5000000

> results1<-predict(model1,testdata)
> results1
[1] No
Levels: No Yes
> print("Nilesh Hegde Cs060")
[1] "Nilesh Hegde Cs060"
>
```

Program 9



Program 10