

## Data Mining- Lab Exam

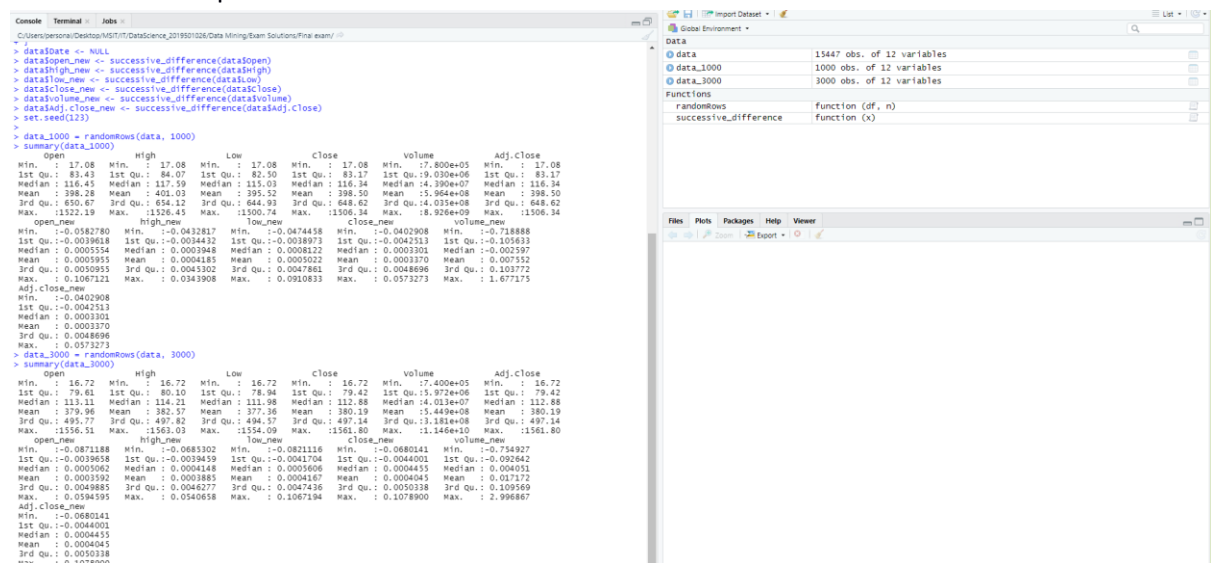
Time: 24 hours

Marks:100

Open a document and update document with your answers for each question and submit it.

1. a) For the dataset BSE\_Sensex\_Index.csv, create an extra column of successive differences for each column of numeric values in this data file. Extract two simple random samples with replacement of 1000 and 3000 observations (rows). Show your R commands for doing this. Do the same thing by using Excel. Show your Excel commands.

**Note:** Successive difference for date d1= (date d1 value-immediate available previous date of d1 value)/immediate available previous date of d1. For the last row fill up values with mean of its immediate three previous row values.



```
> dataDate <- NULL
> data$open_new <- successive_difference(data$open)
> data$high_new <- successive_difference(data$high)
> data$low_new <- successive_difference(data$low)
> data$close_new <- successive_difference(data$close)
> data$volume_new <- successive_difference(data$volume)
> data$adj_close_new <- successive_difference(data$adj_close)
> set.seed(123)
> data_1000 = randomrows(data, 1000)
> summary(data_1000)
```

open	high	low	close	volume	adj_close
Min.: 17.08	Min.: 17.08	Min.: 17.08	Min.: 17.08	Min.: 17.800e+05	Min.: 17.08
1st Qu.: 83.43	1st Qu.: 84.07	1st Qu.: 82.50	1st Qu.: 83.17	1st Qu.: 9.030e+06	1st Qu.: 83.17
Median: 116.45	Median: 117.59	Median: 115.03	Median: 116.34	Median: 4.390e+07	Median: 116.34
Mean: 386.28	Mean: 401.03	Mean: 395.32	Mean: 398.50	Mean: 15.864e+08	Mean: 398.50
3rd Qu.: 650.67	3rd Qu.: 654.12	3rd Qu.: 644.93	3rd Qu.: 648.62	3rd Qu.: 4.035e+08	3rd Qu.: 648.62
Max.: 1522.19	Max.: 1526.45	Max.: 1500.74	Max.: 1506.34	Max.: 16.926e+09	Max.: 1506.34

```
open_new      high_new      low_new      close_new      volume_new      adj_close_new
Min.: -0.058780  Min.: -0.0432817  Min.: -0.0474458  Min.: -0.0402908  Min.: -0.718888
1st Qu.: -0.0039618  1st Qu.: -0.0034432  1st Qu.: -0.0038973  1st Qu.: -0.0042513  1st Qu.: -0.105693
Median: 0.0005554  Median: 0.0003948  Median: 0.0008122  Median: 0.0003301  Median: -0.002597
Mean: 0.0005955  Mean: 0.0004185  Mean: 0.0005022  Mean: 0.0003370  Mean: 0.007552
3rd Qu.: 0.0050955  3rd Qu.: 0.0043302  3rd Qu.: 0.0047861  3rd Qu.: 0.0048696  3rd Qu.: 0.103772
Max.: 0.1067121  Max.: 0.0343908  Max.: 0.0910833  Max.: 0.0573273  Max.: 1.677175
Adj_close_new
Min.: -0.0402908
1st Qu.: -0.0042513
Median: 0.0003301
Mean: 0.0003370
3rd Qu.: 0.0048696
Max.: 0.0573273
> data_3000 = randomrows(data, 3000)
> summary(data_3000)
```

open	high	low	close	volume	adj_close
Min.: 16.72	Min.: 16.72	Min.: 16.72	Min.: 16.72	Min.: 17.400e+05	Min.: 16.72
1st Qu.: 79.61	1st Qu.: 80.10	1st Qu.: 78.94	1st Qu.: 79.42	1st Qu.: 15.972e+06	1st Qu.: 79.42
Median: 113.11	Median: 114.21	Median: 111.98	Median: 112.88	Median: 4.021e+07	Median: 112.88
Mean: 379.96	Mean: 382.57	Mean: 377.36	Mean: 380.19	Mean: 15.449e+08	Mean: 380.19
3rd Qu.: 495.77	3rd Qu.: 497.82	3rd Qu.: 494.57	3rd Qu.: 497.14	3rd Qu.: 15.181e+08	3rd Qu.: 497.14
Max.: 1556.51	Max.: 1561.03	Max.: 1554.09	Max.: 1561.80	Max.: 1.146e+10	Max.: 1561.80

```
open_new      high_new      low_new      close_new      volume_new      adj_close_new
Min.: -0.0871188  Min.: -0.0685302  Min.: -0.0821116  Min.: -0.0680241  Min.: -0.754927
1st Qu.: -0.0039658  1st Qu.: -0.0039459  1st Qu.: -0.0041704  1st Qu.: -0.0044001  1st Qu.: -0.092642
Median: 0.0005062  Median: 0.0004148  Median: 0.0005606  Median: 0.0004453  Median: 0.004031
Mean: 0.0003592  Mean: 0.0003885  Mean: 0.0004167  Mean: 0.0004045  Mean: 0.017172
3rd Qu.: 0.0048885  3rd Qu.: 0.0046277  3rd Qu.: 0.0047436  3rd Qu.: 0.0050338  3rd Qu.: 0.109569
Max.: 0.0584595  Max.: 0.0340658  Max.: 0.1007194  Max.: 0.1078900  Max.: 2.996867
Adj_close_new
Min.: -0.0680141
1st Qu.: -0.0044001
Median: 0.0004453
Mean: 0.0004045
3rd Qu.: 0.0050338
Max.: 0.1078900
```

- b) For your samples, use the functions mean(), max(), var() and quartile(.,25) to compute the mean, maximum, variance and 1st quartile respectively for each column which has successive differences. Show your R code and the resulting values.

Do the same thing by using Excel. Show your Excel commands.

```
> mean(data_1000$open_new)
[1] 0.0005955025
> mean(data_1000$high_new)
[1] 0.0004184797
> mean(data_1000$low_new)
[1] 0.0005022487
> mean(data_1000$close_new)
[1] 0.0003369592
> mean(data_1000$volume_new)
[1] 0.007551912
> mean(data_1000$Adj.close_new)
[1] 0.0003369592
> var(data_1000$open_new)
[1] 8.714339e-05
> var(data_1000$high_new)
[1] 6.119132e-05
> var(data_1000$low_new)
[1] 8.313995e-05
> var(data_1000$close_new)
[1] 7.637739e-05
> var(data_1000$volume_new)
[1] 0.0327711
> var(data_1000$Adj.close_new)
[1] 7.637739e-05
> max(data_1000$open_new)
[1] 0.1067121
> max(data_1000$high_new)
[1] 0.03439077
> max(data_1000$low_new)
[1] 0.09108332
> max(data_1000$close_new)
[1] 0.05732732
> max(data_1000$volume_new)
[1] 1.677175
> max(data_1000$Adj.close_new)
[1] 0.05732732
> quantile(data_1000$open_new,0.25)
      25%
-0.003961827
> quantile(data_1000$high_new,0.25)
      25%
-0.003443228
> quantile(data_1000$low_new,0.25)
      25%
-0.003897353
> quantile(data_1000$close_new,0.25)
      25%
-0.004251294
> quantile(data_1000$volume_new,0.25)
      25%
-0.1056329
> quantile(data_1000$Adj.close_new,0.25)
      25%
-0.004251294
>
```

```

>
> mean(data_3000$open_new)
[1] 0.0003591911
> mean(data_3000$high_new)
[1] 0.0003884621
> mean(data_3000$low_new)
[1] 0.0004167
> mean(data_3000$close_new)
[1] 0.0004044752
> mean(data_3000$volume_new)
[1] 0.0171718
> mean(data_3000$Adj.close_new)
[1] 0.0004044752
> var(data_3000$open_new)
[1] 8.509529e-05
> var(data_3000$high_new)
[1] 6.81047e-05
> var(data_3000$low_new)
[1] 8.768766e-05
> var(data_3000$close_new)
[1] 8.588174e-05
> var(data_3000$volume_new)
[1] 0.03939109
> var(data_3000$Adj.close_new)
[1] 8.588174e-05
> max(data_3000$open_new)
[1] 0.05945946
> max(data_3000$high_new)
[1] 0.05406578
> max(data_3000$low_new)
[1] 0.1067194
> max(data_3000$close_new)
[1] 0.10789
> max(data_3000$volume_new)
[1] 2.996867
> max(data_3000$Adj.close_new)
[1] 0.10789
> quantile(data_3000$open_new,0.25)
      25%
-0.003965834
> quantile(data_3000$high_new,0.25)
      25%
-0.003945885
> quantile(data_3000$low_new,0.25)
      25%
-0.004170403
> quantile(data_3000$close_new,0.25)
      25%
-0.00440009
> quantile(data_3000$volume_new,0.25)
      25%
-0.09264194
> quantile(data_3000$Adj.close_new,0.25)
      25%
-0.00440009
> |

```

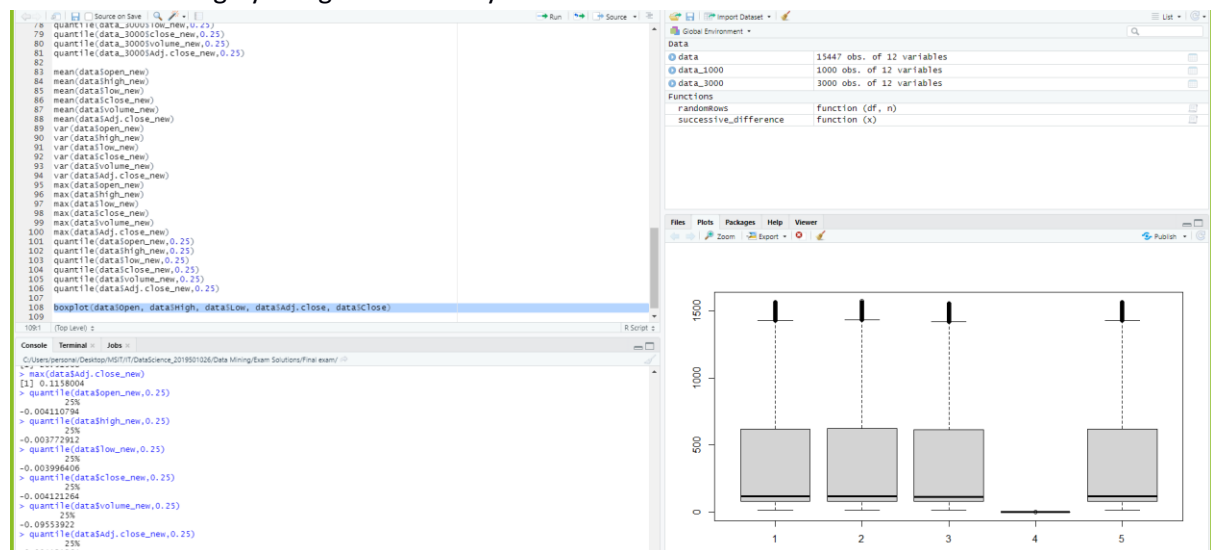
c) Compute the same quantities in part b on the entire data set and show your answers. How much do they differ from your answers in part b? Do you find any significant difference between two sample values like mean in comparison with entire data? If so what explanation you can give for that?

Do the same thing by using Excel. Show your Excel commands.

```
-0.00440009  
> mean(data$open_new)  
[1] 0.000329528  
> mean(data$high_new)  
[1] 0.0003188991  
> mean(data$low_new)  
[1] 0.0003266191  
> mean(data$close_new)  
[1] 0.0003303709  
> mean(data$volume_new)  
[1] 0.02062874  
> mean(data$Adj.close_new)  
[1] 0.0003303709  
> var(data$open_new)  
[1] 9.027493e-05  
> var(data$high_new)  
[1] 6.939914e-05  
> var(data$low_new)  
[1] 8.646474e-05  
> var(data$close_new)  
[1] 9.350347e-05  
> var(data$volume_new)  
[1] 0.09080738  
> var(data$Adj.close_new)  
[1] 9.350347e-05  
> max(data$open_new)  
[1] 0.1067121  
> max(data$high_new)  
[1] 0.08037943  
> max(data$low_new)  
[1] 0.1067194  
> max(data$close_new)  
[1] 0.1158004  
> max(data$volume_new)  
[1] 26.51968  
> max(data$Adj.close_new)  
[1] 0.1158004  
> quantile(data$open_new,0.25)  
25%  
-0.004110794  
> quantile(data$high_new,0.25)  
25%  
-0.003772912  
> quantile(data$low_new,0.25)  
25%  
-0.003996406  
> quantile(data$close_new,0.25)  
25%  
-0.004121264  
> quantile(data$volume_new,0.25)  
25%  
-0.09553922  
> quantile(data$Adj.close_new,0.25)  
25%  
-0.004121264  
> |
```

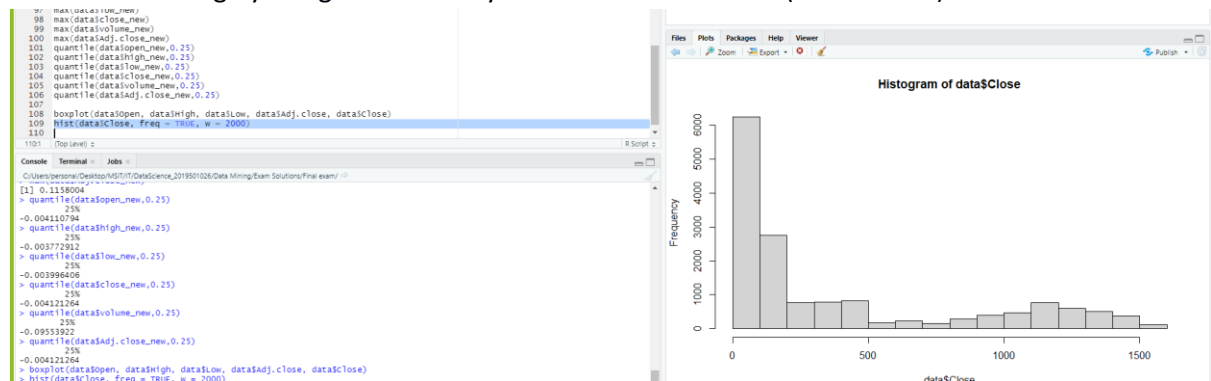
d) Use R to produce a single graph displaying a boxplot for open, close, high and low. Include the R commands and the plot.

Do the same thing by using Excel. Show your Excel commands



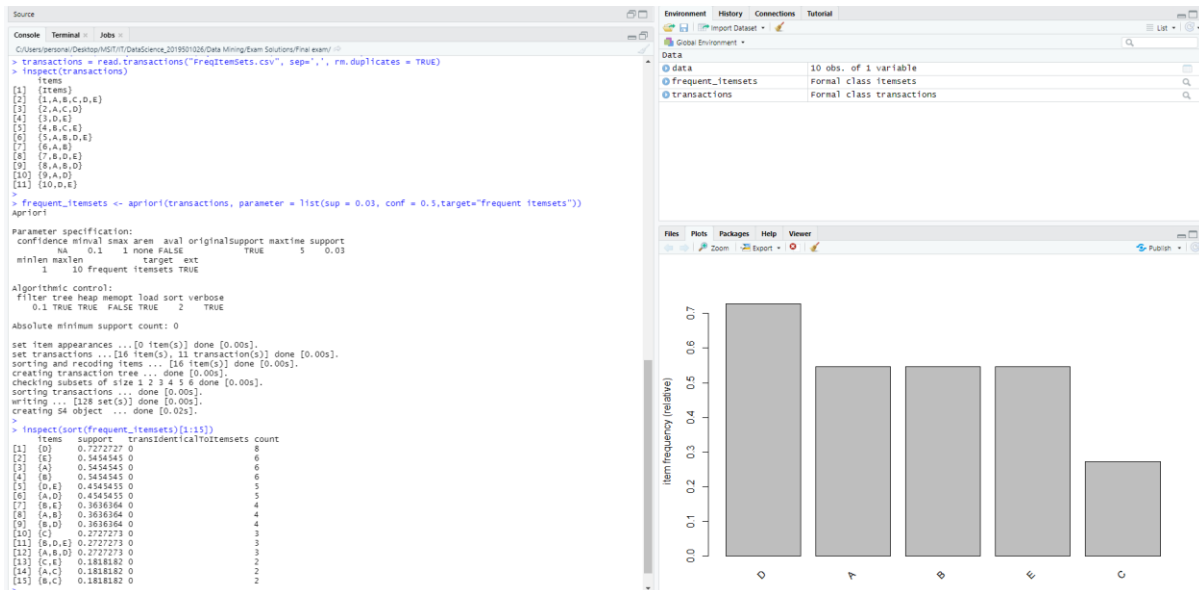
e) Use R to produce a frequency histogram for Close values. Use intervals of width 2000 beginning at 0. Include the R commands and the plot.

Do the same thing by using Excel. Show your Excel commands. (10+10=20M)

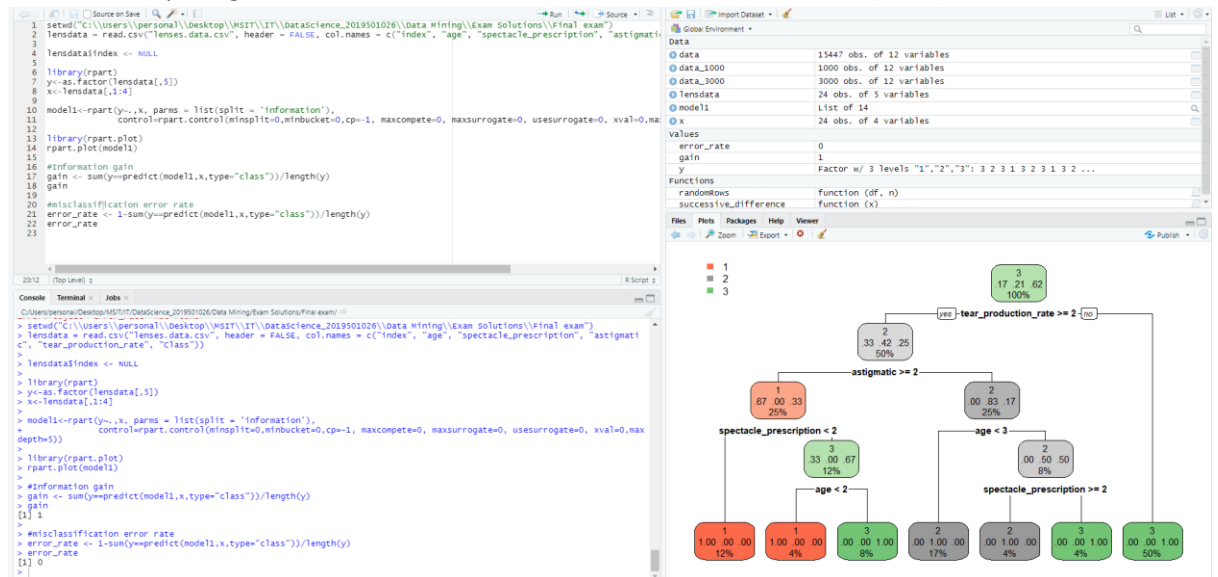


2. Implement Apriori Algorithm or use built in packages to find out the frequent itemsets and generate rules for frequent itemsets. Trace and submit the program output for the following given dataset of transactions with a minimum support of 3. (10M)

TID, Items  
101, A,B,C,D,E  
102, A,C,D  
103, D,E  
104, B,C,E  
105, A,B,D,E  
106, A,B  
107, B,D,E  
108, A,B,D  
109, A,D  
110, D,E



- Build Decision Trees by using i) information gain and ii) misclassification error rate for Lenses Data Set provided at <http://archive.ics.uci.edu/ml/datasets/Lenses>. In terms of tree size what do you conclude comparing these two? (10M)



- Fit 1, 2 and 3-nearest-neighbor classifiers to the Liver Disorders Data Set at <http://archive.ics.uci.edu/ml/datasets/Liver+Disorders> for measures Euclidean and cosine. Last but one column is a decision attribute. Replace decision values in to 4 classes ( $0 \leq c_1 < 5$ ,  $5 \leq c_2 < 10$ ,  $10 \leq c_3 < 15$ ,  $15 \leq c_4 \leq 20$ ). Last column is a data split column in to training and test sets. 1 means the object is used for training. 2 means the object is used for testing. Explain the input parameters you provided for the classifier. Compute the misclassification error on the training data and also on the test data. Annotate your program. (10M)

```

> setwd("C:/Users/personal/Desktop/MSIT/IT/DataScience_2019501026/Data Mining/Exam Solutions/Final exam/")
> data = read.csv("liver_data.csv", header = FALSE, col.names = c("mcv", "alkphos", "sgpt", "sgot", "gammagt", "drinks", "selector"))
> data = na.omit(data)
> data$drinks = cut(data$drinks, breaks = c(0,5,10,15,20,25), labels = c("c1", "c2", "c3", "c4", "c4"), right = FALSE)
> data = na.omit(data)
> #training and test sets
> traindata = subset(data, data$selector == 1)
> testdata = subset(data, data$selector == 2)
>
> x_train = subset(traindata, select = -c(selector, drinks))
> x_test = subset(testdata, select = -c(selector, drinks))
>
> y_train = traindata[,6, drop = TRUE]
> y_test = testdata[,6, drop = TRUE]
>
> #For Training Data
> library(class)
> model1 = knn(x_train, x_test, y_train, k = 1)
> 1-sum(y_train==model1)/length(y_train)
[1] 0.2827586
> #printing messages:
1: In ==.default*(y_train, model1) :
  longer object length is not a multiple of shorter object length
2: In 1-sum(y_train==model1)/length(y_train) :
  longer object length is not a multiple of shorter object length
> model2 = knn(x_train, x_test, y_train, k = 2)
> 1-sum(y_train==model2)/length(y_train)
[1] 0.1310345
> model3 = knn(x_train, x_test, y_train, k = 3)
> 1-sum(y_train==model3)/length(y_train)
[1] 0.2
> #For Test Data
> model4 = knn(x_train, x_test, y_test, k = 1)
> 1-sum(y_test==model4)/length(y_test)
[1] 0.445
> model5 = knn(x_train, x_test, y_test, k = 2)
> 1-sum(y_test==model5)/length(y_test)
[1] 0.43
> model6 = knn(x_train, x_test, y_test, k = 3)
> 1-sum(y_test==model6)/length(y_test)
[1] 0.42
>

```

Global Environment

Object	Class	Attributes
data	data.frame	345 obs. of 7 variables
testdata	data.frame	200 obs. of 7 variables
traindata	data.frame	145 obs. of 7 variables
x_train	data.frame	200 obs. of 5 variables
x_test	data.frame	145 obs. of 5 variables
y_train	factor	Factor w/ 4 levels "c1","c2","c3",...: 1 1 1 1 2 2 2 1 1 2 ...
y_test	factor	Factor w/ 4 levels "c1","c2","c3",...: 1 1 1 1 1 1 1 1 1 1 ...
model1	factor	Factor w/ 4 levels "c1","c2","c3",...: 1 1 1 1 1 1 1 1 1 1 ...
model2	factor	Factor w/ 4 levels "c1","c2","c3",...: 1 1 1 1 1 1 1 1 1 1 ...
model3	factor	Factor w/ 4 levels "c1","c2","c3",...: 1 1 1 1 1 1 1 1 1 1 ...
model4	factor	Factor w/ 4 levels "c1","c2","c3",...: 1 1 1 1 2 2 2 1 1 2 ...
model5	factor	Factor w/ 4 levels "c1","c2","c3",...: 1 1 1 1 1 2 2 1 1 1 ...
model6	factor	Factor w/ 4 levels "c1","c2","c3",...: 1 1 1 1 1 2 2 1 1 1 ...

- Use Support Vector machine for above problem. And compare the performance of both. Explain the input parameters you provided for the classifier. (10M)

```

1 data = read.csv("liver_data.csv", header = FALSE, col.names = c("mcv", "alkphos", "sgpt", "sgot", "gammagt", "drinks", "selector"))
2 data = na.omit(data)
3 data$drinks = cut(data$drinks, breaks = c(0,5,10,15,20,25), labels = c("c1", "c2", "c3", "c4", "c4"), right = FALSE)
4 data = na.omit(data)
5
6 #training and test sets
7 traindata = subset(data, data$selector == 1)
8 testdata = subset(data, data$selector == 2)
9
10 x_train = subset(traindata, select = -c(selector, drinks))
11 x_test = subset(testdata, select = -c(selector, drinks))
12
13 y_train = traindata[,6, drop = TRUE]
14 y_test = testdata[,6, drop = TRUE]
15
16 library(e1071)
17
18 #For training
19 model = svm(x_train, y_train)
20 1-sum(y_train==predict(model,x_train))/length(y_train)
21
22 #For test data
23 1-sum(y_test==predict(model,x_test))/length(y_test)
24

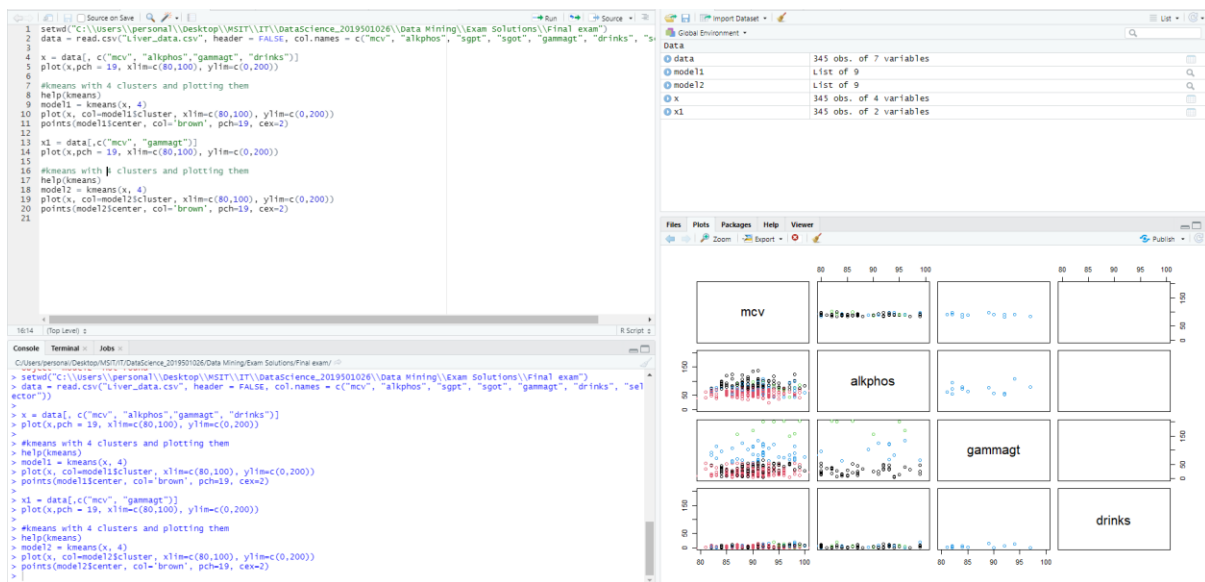
```

Global Environment

Object	Class	Attributes
data	data.frame	345 obs. of 7 variables
model	svm	List of 29
testdata	data.frame	200 obs. of 7 variables
traindata	data.frame	145 obs. of 7 variables
x_train	data.frame	200 obs. of 5 variables
x_test	data.frame	145 obs. of 5 variables
y_train	factor	Factor w/ 4 levels "c1","c2","c3",...: 1 1 1 1 1 1 1 1 1 1 ...
y_test	factor	Factor w/ 4 levels "c1","c2","c3",...: 1 1 1 1 1 1 1 1 1 1 ...

- Create k-means clusters for k=4 for the Liver Disorders Data Set at <http://archive.ics.uci.edu/ml/datasets/Liver+Disorders> . Explain the input parameters you provided for the clustering algorithm. Plot the fitted cluster centers using a different color. Finally assign the cluster membership for the points to the nearest cluster center. Color the points according to their cluster membership. (10+10=20M)





7. Compute the misclassification error that would result if you used your clustering rule to classify the data by assigning the majority class of the cluster. (10M)

8. Consider the dataset BSE\_Sensex\_Index.csv. Create an extra column of successive growth rate for column close where the successive growth rate is defined as  

$$\text{value of day } x - \text{value of day } x-1 / \text{value of day } x-1$$
Use a z score cut off of 3 to identify any outliers. List the respective dates from the csv file on which day these outliers fall. (10M)

