EAS 507: STATISTICAL DATA MINING II

Report for Homework #1

Kishore Ravisankar
UBIT: kravisan

1.

In this exercise, we have the MovieLense dataset from the recommenderlab package, which contains about 100,000 ratings from 943 users on 1664 movies. The aim of this exercise is to design a user based collaborative filtering recommendation system, which does the following:

- For each user, find the movies that the user has not watched, and find similar users who have watched those movies.
- Based on the ratings by the other users, predict the ratings that the user in context might assign to those movies.

In this exercise, we will be creating two data models, one based on the holdout approach and the other based on cross-validation approach. We shall compare both the errors, to evaluate the performance. For both models, we consider the number of similar users (k) to be 50.

Holdout approach:

The training dataset is created by using 80% of the data. The remaining dataset is used as test dataset.

We create a UBCF recommendation system using the Recommender() function, and use the cosine similarity of users. The created model learnt the ratings of 754 users.

Predictions are made on the known ratings in the test set, and the error is calculated by using the calcPredictionAccuracy() function with the unknown ratings in the test set.

The RMSE obtained was 1.194375. The MSE and MAE obtained are as follows:

```
> pred_error_holdout
          RMSE     MSE       MAE
UBCF 1.194375 1.42653 0.9386679
```

Cross-validation approach:

In this approach, the dataset is divided into 5 parts. Three parts are used as training data, one part is used for validation, and the other part is used as testing data. This is repeated for 5 times. This is achieved by using the cross-validation parameter in the evaluationScheme() method.

Following the previous approach, we create a UBCF recommendation system taking into account the cosine similarity of users, using the Recommender() function and the cross-validated dataset. The created model learnt the ratings of 752 users.

Predictions are made on the known ratings in the test set, and the error is calculated by using the calcPredictionAccuracy() function with the unknown ratings in the test set.

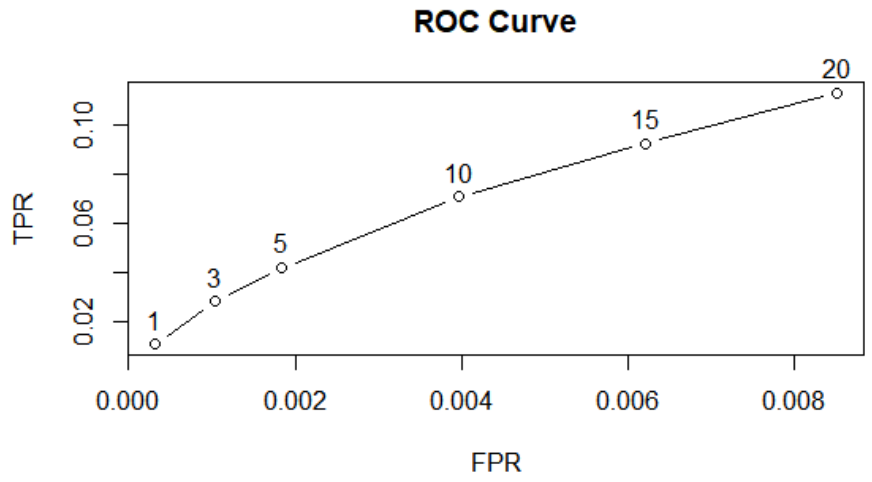
The RMSE obtained was 1.170152. The MSE and MAE obtained are as follows:

```
> pred_error_kfold
          RMSE      MSE       MAE
UBCF 1.170152 1.369255 0.9319196
```

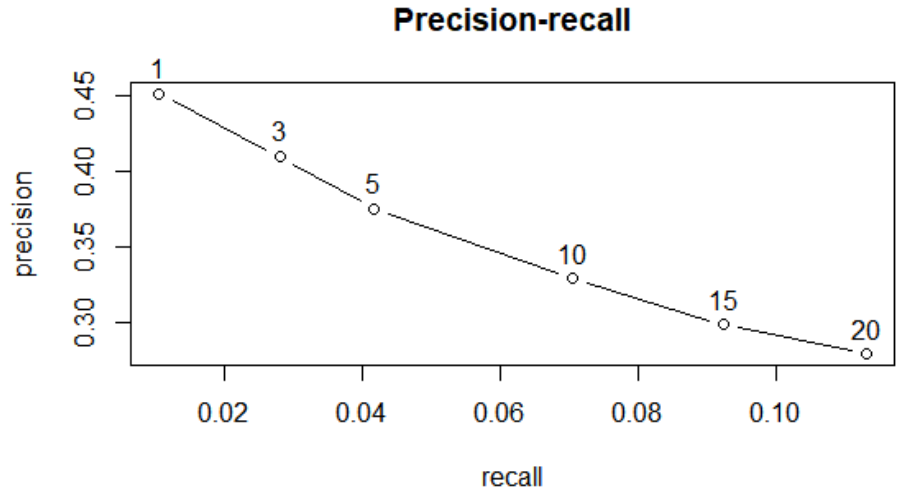
Now we evaluate the accuracy of the recommendations made by the model, by obtaining the ROC curve, and the precision-recall curve. The evaluate() method is used for this purpose. The number of recommendations used are 1,3,5,10,15 and 20 respectively. The ROC measures obtained are as follows:

```
> avg(model_error_kfold)
          TP         FP       FN       TN precision     recall        TPR          FPR
1  0.4314136  0.5235602 56.22827 1603.817 0.4517455 0.01055105 0.01055105 0.0003224249
3  1.1748691  1.6900524 55.48482 1602.650 0.4100859 0.02797037 0.02797037 0.0010418336
5  1.7916230  2.9832461 54.86806 1601.357 0.3752346 0.04167757 0.04167757 0.0018403779
10 3.1455497  6.4041885 53.51414 1597.936 0.3293326 0.07053256 0.07053256 0.0039542986
15 4.2848168 10.0397906 52.37487 1594.301 0.2990725 0.09236859 0.09236859 0.0062027978
20 5.3413613 13.7581152 51.31832 1590.582 0.2796232 0.11313484 0.11313484 0.0085048290
```

The ROC curve is plotted and obtained as follows:



The precision-recall plot is obtained as follows:



We see that the performance of the model is poor, due to the low area under the curve (AUC). However, we see that the RMSE of the cross-validated model is slightly better than the holdout model.

2.

Using the recommenderlab package, we create collaborative filtering models to predict the unknown ratings of user 2.

a)

We predict the unknown ratings of user 2 using user-based collaborative filtering, by taking into account the Pearson correlation of users, with mean centering. The unknown ratings are as follows:

- 4.080462 for Item 2
- 3.844854 for Item 4

b)

We predict the unknown ratings of user 2 using item-based collaborative filtering, by taking into account the cosine similarity of users. The unknown ratings are as follows:

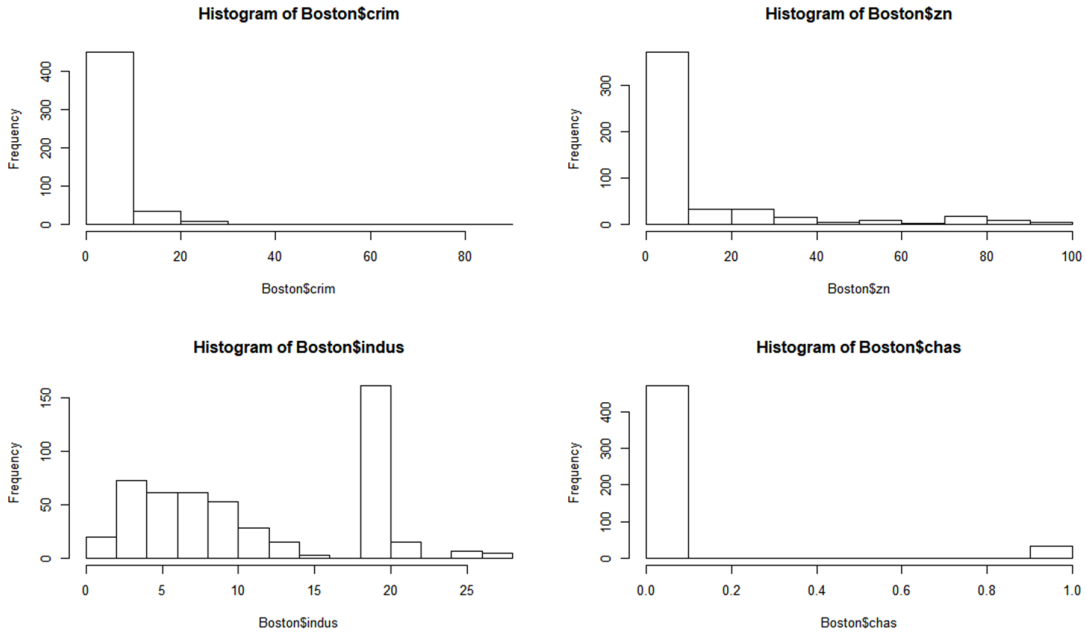
- 3.999355 for Item 2
- 3.98319 for Item 4

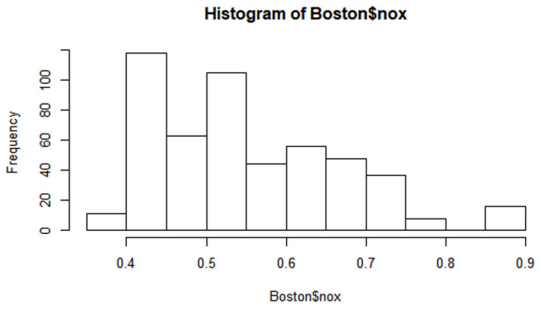
3.

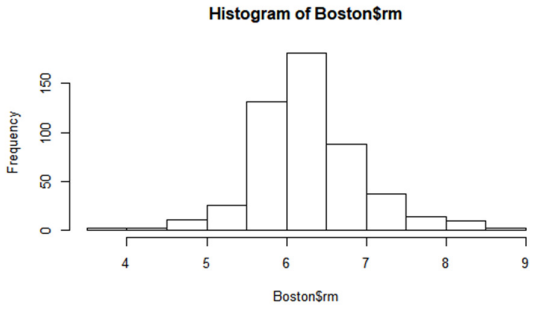
In this exercise, we have the Boston dataset, which contains 14 predictors for 506 observations. We will be using association rules and linear regression model on the dataset, and analyse our results.

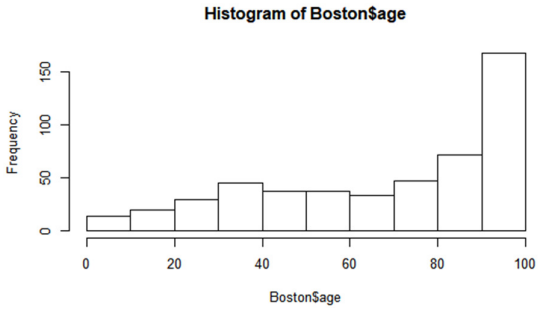
a)

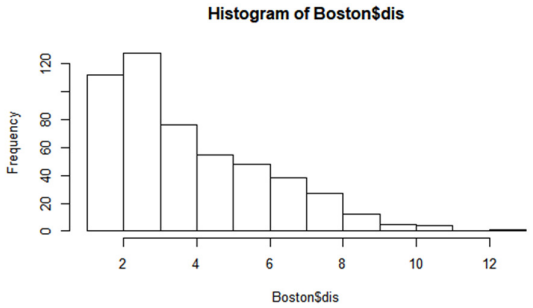
All the variables in the dataset are visualised using histograms, and represented as follows:

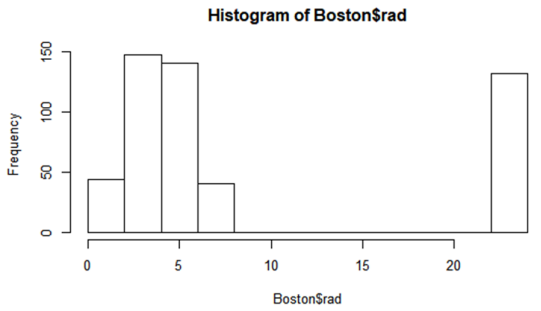


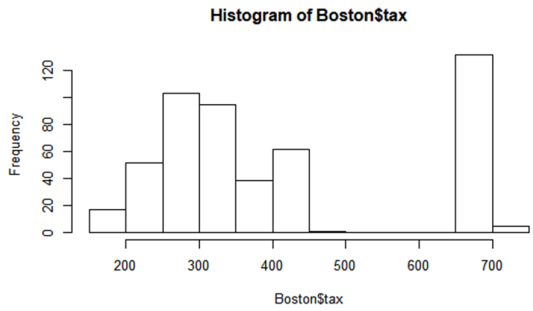

Histogram of Boston$nox
Frequency
0
20
40
60
80
100
0.4
0.5
0.6
0.7
0.8
0.9
Boston$nox

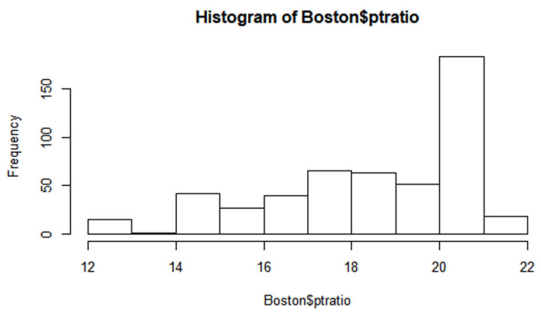

Histogram of Boston$rm
Frequency
0
50
100
150
4
5
6
7
8
9
Boston$rm

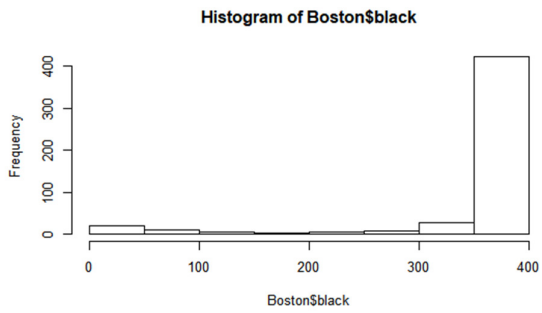

Histogram of Boston$age
Frequency
0
50
100
150
0
20
40
60
80
100
Boston$age

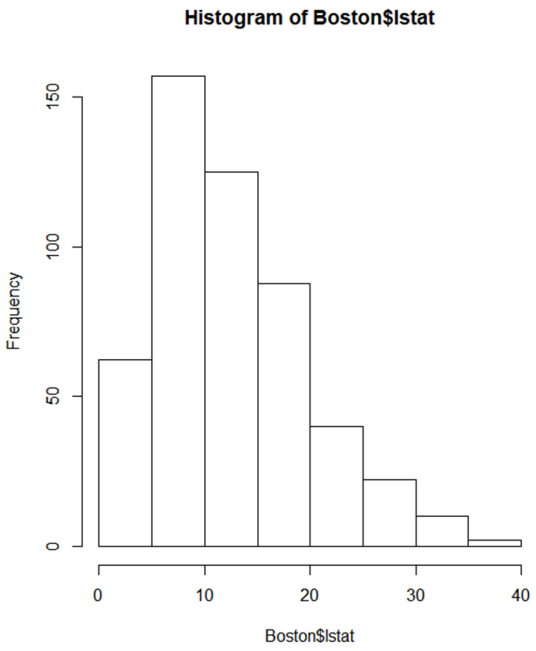

Histogram of Boston$dis
Frequency
0
20
40
60
80
100
120
2
4
6
8
10
12
Boston$dis

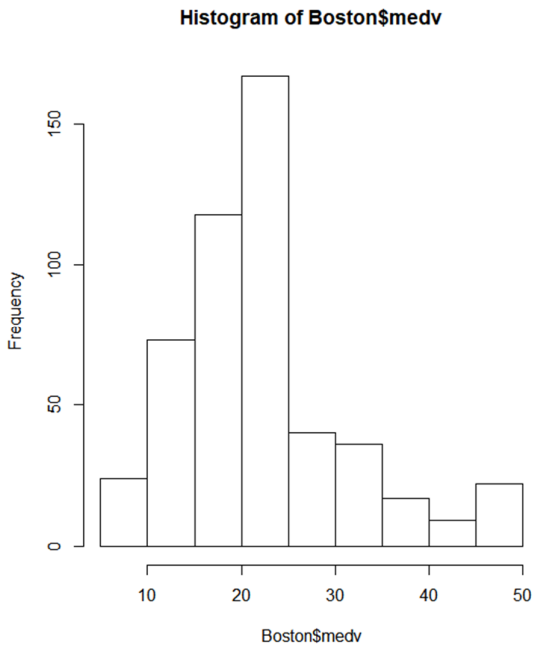

Histogram of Boston$rad
Frequency
0
50
100
150
0
5
10
15
20
Boston$rad


Histogram of Boston$tax
Frequency
0
20
40
60
80
100
120
200
300
400
500
600
700
Boston$tax


Histogram of Boston$ptratio
Frequency
0
50
100
150
12
14
16
18
20
22
Boston$ptratio


Histogram of Boston$black
Frequency
0
100
200
300
400
0
100
200
300
400
Boston$black





Before creating a binary incidence matrix for the given dataset, we categorise each variable into different levels. The categorisation of each of the 14 variables is explained as follows:

crim: The per capita crime rate is categorised into three levels; Low, Moderate and High. Low crime rates are categorised between 0 and 10 (which is where most of the data lies in), moderate between 10 and 20, and high between 20 and 90.

zn: The proportion of residential land zoned is categorised into three equal ranges, Small (0-30), Medium (30-60) and Large (60-90). Again, we see that most of the data lies in the first category, like in the previous case.

indus: The proportion of non-retail business acres is almost spread out in the histogram above. Therefore, this variable is categorised into three equal levels: Small (0-10), Medium (10-20) and Large (20-30).

chas: The Charles river variable will not be considered while forming association rules, because almost all of the observations have the same value, that is, 0 or the tract does not bound the river. This does not really help in forming association rules, as stated before.

nox: The nitric oxide concentration variable is spread out throughout the histogram, so it is categorised into Low (0.3-0.5 ppm), Medium (0.5-0.7 ppm) and High (0.7-0.9 ppm).

rm: The average number of rooms per dwelling is categorised into three levels: Low (3-5 rooms), Medium (5-7 rooms) and High (7-9 rooms). We observe that most of the observations fall in the second category.

age: The proportion of owner-occupied units built prior to 1940 is classified into three groups: New (0-25 years), Fairly old (25-60 years) and old (60-100 years). We see that all the observations are evenly scattered across the histogram. It can also be observed that the age of many units are more than 90 years.

dis: The weighted distance (in miles) to employment centres in Boston is classified into three levels: Close (3-5 miles), Near (5-7 miles) and Far (7-9 miles). Most of the observations fall under the first two categories.

rad: The index of radial accessibility to highways is classified in a binary manner, into Accessible (0-10), and Not Accessible (10-24). Most of the observations fall in the first category.

tax: The property tax rate per $10000 is classified into three levels: Low ($0-$300), Medium ($300-$500) and High ($500-$900). Most of the observations fall in the first two categories.

ptratio: The pupil-teacher ratio is distributed across the histogram evenly, so three categories of equal ranges are created: Low (0-16), Medium (16-20) and High (20-24).

black: The proportion of blacks by town is classified into two categories, Low (0-200) and High (200-400). We see that most of the observations fall under the second category.

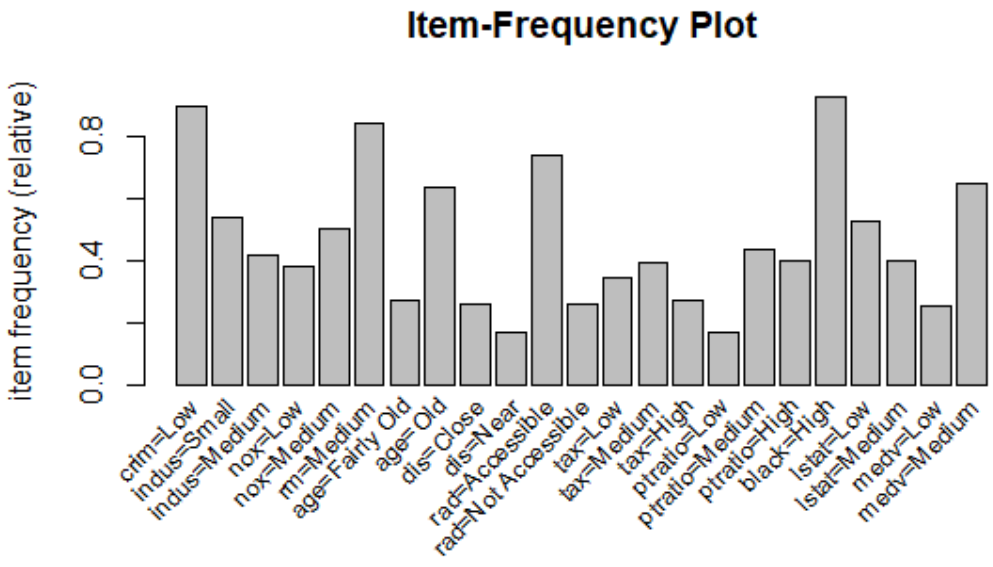
lstat: The lower status of the population (in %) is classified into three levels, Low (0-12%), Medium (12-24%) and High (24-40%). Most of the observations are evenly distributed.

medv: The median value of owner occupied homes (multiples of $1000) is classified into three equal ranges: Low (0-17), Medium (17-34) and High (34-51). Again, most observations are evenly spaced in the histogram.

Based on the above categorisation, the data is converted into a binary incidence matrix.

b)

Using the above created binary incidence matrix, the data is presented as an item-frequency plot, which is as follows:



It should be noted that categories with a minimum support of 0.15 were only plotted in the above chart.

Now, the apriori algorithm is applied on the categorised dataset, that is, the binary incidence matrix. The minimum support used while creating the rules is 0.05, because 14 categories had a support of less than 0.15, which is quite significant in nature. Therefore, the minimum support was lowered to include more categories. The value of confidence threshold used was 0.75. Using these parameters, 48723 rules were created.

The summary of the rules is as follows:

```
> summary(rules)
set of 48723 rules

rule length distribution (lhs + rhs):sizes
    1     2     3     4     5     6     7     8     9    10
    3   189  1561  5633 11329 13765 10288  4620  1178   157

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   5.000   6.000   5.911   7.000  10.000

summary of quality measures:
    support          confidence          lift            count
 Min.   :0.05138   Min.   :0.7500   Min.   :0.8126   Min.   : 26.00
 1st Qu.:0.05929   1st Qu.:0.8667   1st Qu.:1.1195   1st Qu.: 30.00
 Median :0.07312   Median :0.9733   Median :1.3529   Median : 37.00
 Mean   :0.09549   Mean   :0.9325   Mean   :1.5659   Mean   : 48.32
 3rd Qu.:0.10870   3rd Qu.:1.0000   3rd Qu.:1.7593   3rd Qu.: 55.00
 Max.   :0.92292   Max.   :1.0000   Max.   :6.8521   Max.   :467.00

mining info:
          data ntransactions support confidence
 bin_inc_matrix           506    0.05       0.75
```

c)

Here, we need to identify rules for areas with low crime rate and close proximity to the city. We accordingly specify those conditions using the subset() function. The rules obtained ordered by high confidence are as follows:

```
> inspect(head(subset(rules, subset=rhs %in% "crim=Low"),5,by="confidence"))
    lhs                rhs          support    confidence lift     count
[1] {indus=Large}  => {crim=Low} 0.05335968 1          1.119469 27
[2] {zn=Medium}    => {crim=Low} 0.06521739 1          1.119469 33
[3] {zn=Large}     => {crim=Low} 0.06916996 1          1.119469 35
[4] {dis=Far}      => {crim=Low} 0.07707510 1          1.119469 39
[5] {age=New}      => {crim=Low} 0.09683794 1          1.119469 49
```

If areas with low crime rates should be chosen, areas with large proportions of industrial and residential land should be chosen. The age of the house should be new.

```
> inspect(head(subset(rules, subset=rhs %in% "dis=Close"),5,by="confidence"))
    lhs                                  rhs           support confidence     lift count
[1] {indus=Small,
     nox=Medium,
     tax=Medium,
     black=High,
     lstat=Low}                      => {dis=Close} 0.05335968  0.7714286 2.979716    27
[2] {indus=Small,
     nox=Medium,
     rad=Accessible,
     tax=Medium,
     black=High,
     lstat=Low}                      => {dis=Close} 0.05335968  0.7714286 2.979716    27
[3] {crim=Low,
     indus=Small,
     nox=Medium,
     tax=Medium,
     black=High,
     lstat=Low}                      => {dis=Close} 0.05335968  0.7714286 2.979716    27
[4] {crim=Low,
     indus=Small,
     nox=Medium,
     rad=Accessible,
     tax=Medium,
     black=High,
     lstat=Low}                      => {dis=Close} 0.05335968  0.7714286 2.979716    27
[5] {indus=Small,
     nox=Medium,
     tax=Medium,
     lstat=Low}                      => {dis=Close} 0.05335968  0.7500000 2.896947    27
```

Based on the above rules, if an area close to the city should be selected, the proportion of blacks living should be high, and the nitric oxide concentration should be medium. The property tax rate of houses in the area should be moderate, and the area should be easily accessible by highways. The proportion of industrial land should be small, the crime rate should be low, and the lower status of the population should be low.

Based on the above observations, we can advise the student about selecting an area with low crime rate, as close to the city as possible by using the following association rules:

- The proportion of industrial land should be moderate, and the area should have medium nitric oxide concentrations.
- The house should be new, and the property tax rate should be moderate.
- The area should be easily accessible by highways.
- The proportion of lower status population should be low, and proportion of blacks living should be high.

d)

Here, we need to identify areas with low pupil-teacher ratios. The appropriate condition is specified using the subset() function, which results in the following rules:

```
> inspect(head(subset(rules, subset=rhs %in% "ptratio=Low"),5,by="confidence"))
    lhs                    rhs              support confidence     lift count
[1] {indus=Medium,
     age=Old,
     tax=Medium}       => {ptratio=Low} 0.05928854       0.75 4.464706    30
[2] {indus=Medium,
     age=Old,
     rad=Accessible,
     tax=Medium}       => {ptratio=Low} 0.05928854       0.75 4.464706    30
[3] {crim=Low,
     indus=Medium,
     age=Old,
     tax=Medium}       => {ptratio=Low} 0.05928854       0.75 4.464706    30
[4] {crim=Low,
     indus=Medium,
     age=Old,
     rad=Accessible,
     tax=Medium}       => {ptratio=Low} 0.05928854       0.75 4.464706    30
```

Based on the above association rules, the family should consider the following suggestions while moving in to a new area:

- The crime rate in the area should be low.
- The property tax rate of the house should be medium, and its age should be old.
- The area should have moderate proportions of industrial lands and should be easily accessible by highways.

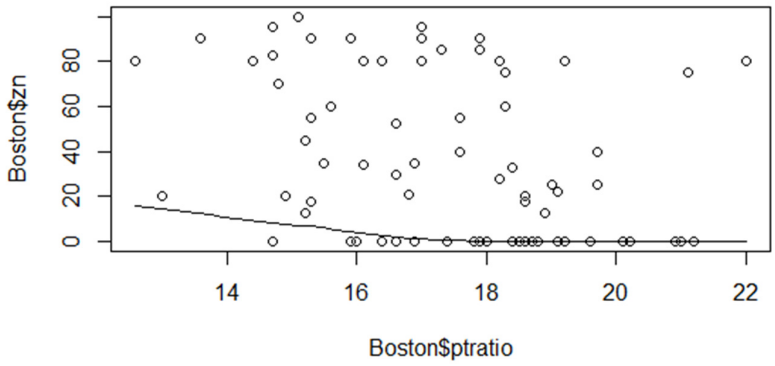
e)

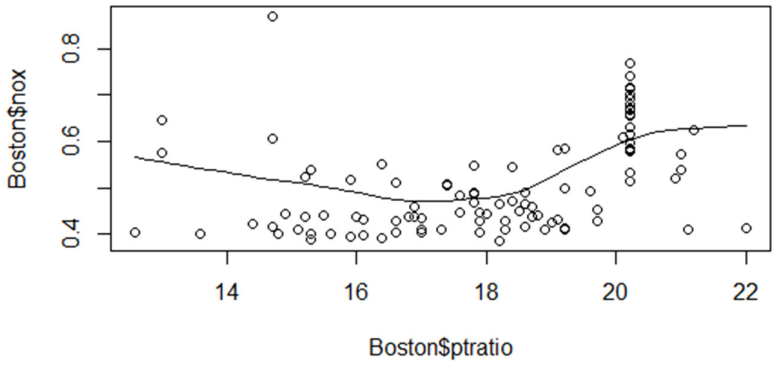
A linear regression model is created using the pupil-teacher ratio as the target variable. The data is split into train and test sets with 80% and 20% of the data respectively. The summary of the linear regression model is as follows:

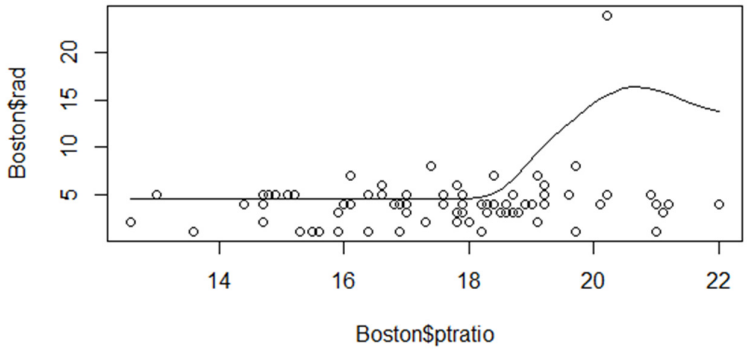
```
> summary(linear_model)

Call:
lm(formula = ptratio ~ ., data = train_data)

Residuals:
    Min      1Q  Median      3Q     Max
-4.0270 -0.9393 -0.0079  0.9164  4.6188

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.603e+01  1.521e+00  17.115  < 2e-16 ***
crim        -1.864e-02  1.219e-02  -1.529   0.1271
zn          -2.339e-02  4.953e-03  -4.722 3.27e-06 ***
indus        4.476e-02  2.248e-02   1.991   0.0472 *
chas        -4.425e-01  3.223e-01  -1.373   0.1705
nox         -1.052e+01  1.286e+00  -8.177 4.14e-15 ***
rm          -2.073e-01  1.586e-01  -1.307   0.1919
age          9.388e-03  4.825e-03   1.946   0.0524 .
dis         -9.023e-03  7.830e-02  -0.115   0.9083
rad          1.390e-01  2.373e-02   5.857 1.00e-08 ***
tax          1.976e-05  1.379e-03   0.014   0.9886
black        1.536e-03  9.687e-04   1.585   0.1137
lstat       -5.128e-02  2.037e-02  -2.518   0.0122 *
medv        -1.083e-01  1.522e-02  -7.115 5.40e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.54 on 390 degrees of freedom
Multiple R-squared:  0.5237,    Adjusted R-squared:  0.5078
F-statistic: 32.99 on 13 and 390 DF,  p-value: < 2.2e-16
```

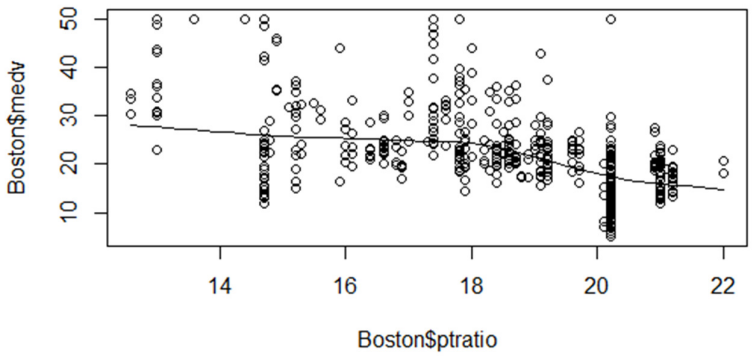
We make predictions of the pupil-teacher ratio on the test data, and obtain a mean square error of 2.679254.

We plot the pupil-teacher ratio versus the most significant variables obtained on creating the regression model. The resulting graphs are as follows:









We observe that the relationship between the significant variables and the pupil-teacher ratio is not linear in nature, and the model does not fit all of the data points. So a proper relationship cannot be established between the variables. Therefore, it can be said that the interpretability of this model is poor, despite the low mean square error.

In the case of association rules, we see that we can clearly identify relationships based on the antecedent and consequent, and parameters like support, lift and confidence. This can help establish an interpretable relationship.

To summarise, we can say that association rules can be used when we need to know exact relationships between two variables based on confidence and lift. Linear regression should be preferred, when we need to establish a linear relationship between the predictors and the response variable, provided the error of the model is very low. However, in terms of interpretability, association rules have the upper hand, and should be preferred over linear regression.

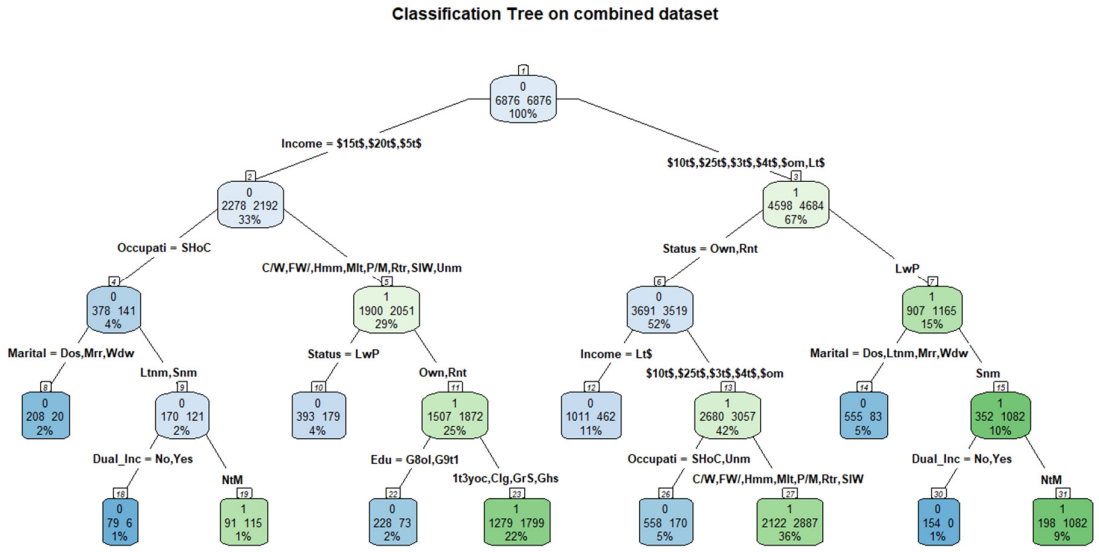
4.

In this exercise, we will be dealing with the marketing dataset. Before we create a reference sample, we will create a column named class in the dataset, with randomly filled zeroes and ones. And to avoid issues while creating association rules, we remove observations with NA values.

Following the above step, we categorise the variables of the dataset into different levels, based on their values.

Now, we create a training dataset with the class column equal to one. The reference sample is created by permuting over observations from the original dataset. The class column is equated to zero in this sample.

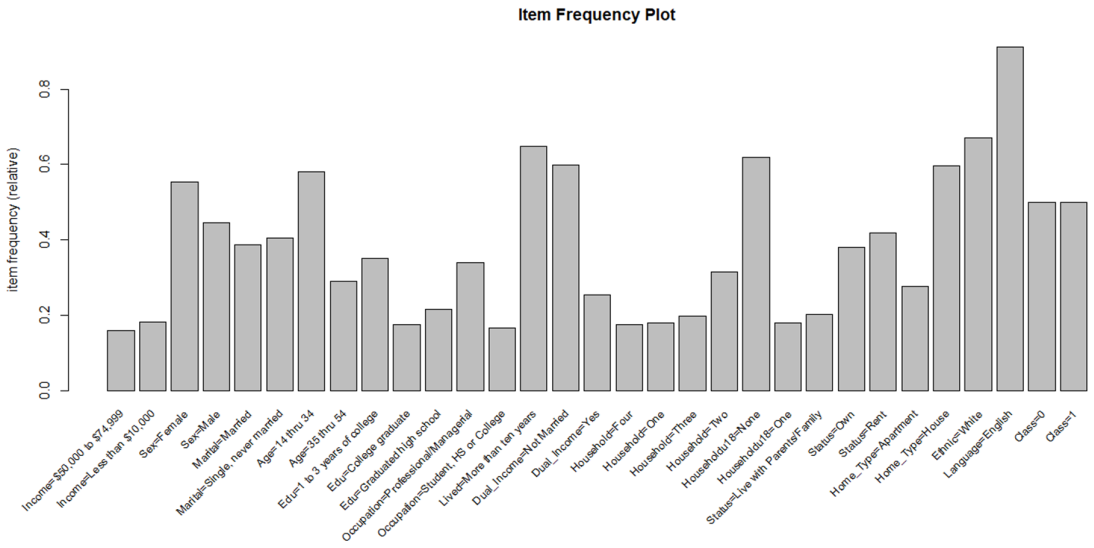
Both the datasets are combined together, and a classification tree is created accordingly, which is presented as follows:



We see that there is one node with a probability of 0.36 (or 36%). The node can be identified based on the following variables:

- The income belongs to the following groups: $10,000 to $14,999, $25,000 to $29,999, $30,000 to $39,999, $40,000 to $49,999, and $75,000 or greater.
- The status of the houses are Own and Rented.
- The occupation belongs to the following categories: Professional/Managerial, Sales Worker, Factory Worker/Laborer/Driver, Clerical/Service Worker, Homemaker, Military, Retired

Now we use association rules to see if we can obtain the same rules. In order to proceed, we create a binary incidence matrix of the dataset. The item frequency plot of items with a support of at least 0.15 is as follows:



Now, Apriori algorithm is applied, with a minimum support of 0.05 and confidence of 0.75.

We get 6421 rules. Their summary is as follows:

```
> summary(rules)
set of 6421 rules

rule length distribution (lhs + rhs):sizes
   1    2    3    4    5    6    7    8
   1   72  687 2039 2403 1051  160    8

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   4.000   5.000   4.651   5.000   8.000

summary of quality measures:
    support          confidence          lift            count
 Min.   :0.05003   Min.   :0.7500   Min.   :0.8255   Min.   :  688
 1st Qu.:0.05621   1st Qu.:0.8233   1st Qu.:1.0185   1st Qu.:  773
 Median :0.06515   Median :0.9124   Median :1.0710   Median :  896
 Mean   :0.07866   Mean   :0.8883   Mean   :1.2443   Mean   : 1082
 3rd Qu.:0.08493   3rd Qu.:0.9435   3rd Qu.:1.3672   3rd Qu.: 1168
 Max.   :0.91158   Max.   :1.0000   Max.   :3.9267   Max.   :12536

mining info:
          data ntransactions support confidence
 bin_inc_matrix         13752    0.05       0.75
```

Now we identify rules where the class equals one, which is what we require. On looking at the first observation below, we see that the association rules and the classification tree rules do not match. Also, the probabilities of node 1 do not match.

```
> inspect(head(subset(rules, subset=rhs %in% "Class=1"),5,by="confidence"))
    lhs                                    rhs          support confidence     lift count
[1] {Marital=Single, never married,
     Age=14 thru 34,
     Occupation=Student, HS or College,
     Dual_Income=Not Married,
     Status=Live with Parents/Family}   => {Class=1} 0.05686446  0.9726368 1.945274   782
[2] {Income=Less than $10,000,
     Marital=Single, never married,
     Age=14 thru 34,
     Dual_Income=Not Married,
     Status=Live with Parents/Family}   => {Class=1} 0.05468296  0.9567430 1.913486   752
[3] {Marital=Single, never married,
     Occupation=Student, HS or College,
     Dual_Income=Not Married,
     Status=Live with Parents/Family}   => {Class=1} 0.05693717  0.9525547 1.905109   783
[4] {Marital=Single, never married,
     Age=14 thru 34,
     Occupation=Student, HS or College,
     Status=Live with Parents/Family}   => {Class=1} 0.05686446  0.9478788 1.895758   782
[5] {Income=Less than $10,000,
     Marital=Single, never married,
     Dual_Income=Not Married,
     Status=Live with Parents/Family}   => {Class=1} 0.05511926  0.9334975 1.866995   758
```