INDIAN STATISTICAL INSTITUTE

End-Semester Examination : 2015–16

Course : Post Graduate Diploma in Business Analytics (First Year)

Subject : Computing for Data Sciences : BAISI–4 for PGDBA–I

Date : 27 November 2015 Maximum Marks : 100 Duration : 3 Hours

You are allowed to use the Scribes and/or the Lecture Notes during the examination. Answer ALL questions.

Problem 1 [25]

Suppose that a single-feature dataset contains 100 observations (x, y), where the feature values x are independent samples drawn from $X \sim \mathcal{N}(0, 4)$, and y are the corresponding values of the target/output variable. The following R code was executed to fit an *ordinary least squares* linear model, and the corresponding output was obtained in terms of the coefficients and the residuals. Figure 1a represents the dataset (x, y) and the linear model $\hat{y} = h_\theta(x) = \theta_0 + \theta_1 x$ obtained.

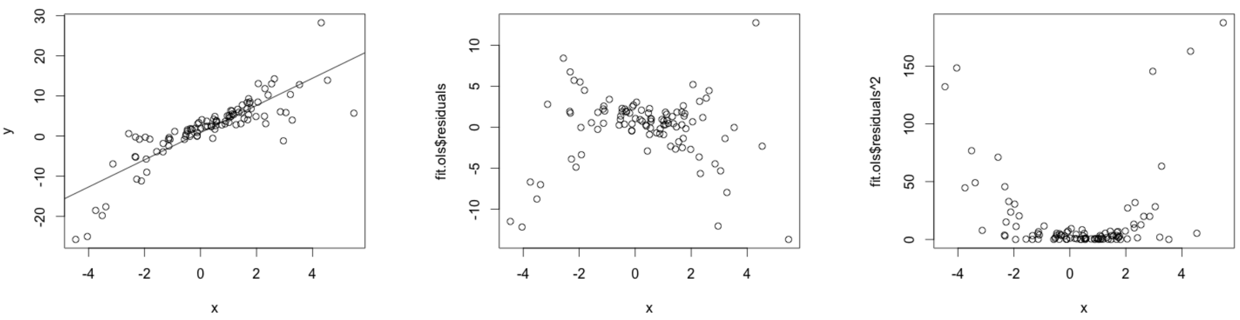
```
fit.ols <- lm(y~x)
fit.ols$coefficients
summary(fit.ols$residuals)
plot(x, y)
abline(fit.ols$coefficients)
plot(x, fit.ols$residuals)
plot(x, fit.ols$residuals^2)
```

```
(Intercept)           x
   0.859571    3.383583

    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-13.6900  -0.8885   0.6999   0.0000   1.9110  12.7600
```

Interpret the output. Suppose that the actual model for (x, y) is approximately linear, of the form $y = a_0 + a_1 x + \epsilon$, with a Gaussian error component $\epsilon \sim \mathcal{N}(\mu_\epsilon, \sigma_\epsilon^2)$. From the distribution of the *residuals* $(y - \hat{y})$, as in Figure 1b, and the distribution of the residuals squared, as in Figure 1c, infer if the distribution of ϵ, and in particular, if μ_ϵ and σ_ϵ, depend on x – explain your answer.

In such a scenario, will a *weighted least squares* model fit the dataset better? If not, explain why. If yes, suggest a suitable weight vector $\mathbf{w}(x)$ that can be used for the weighted linear regression.



(a) Linear model

(b) Residuals

(c) Residuals squared

Figure 1: Ordinary least squares regression on the dataset (x, y)

Problem 2 **[25]**

The following R code was executed on the stock iris dataset, to obtain the *decision tree* as shown in Figure 2. Given that there are exactly four features in the iris dataset – Sepal.Length, Sepal.Width, Petal.Length and Petal.Width – what can you say about the relative importance of the features in classifying the dataset into the three species – setosa, versicolor and virginica?

Note that there are 150 observations in total, with 50 observations each for the three species – setosa, versicolor and virginica. From the decision tree shown in Figure 2, calculate the *information gain* at the first level of the tree (i.e., at node $\boxed{1}$ or the root node), in terms of Shannon entropy.

```
irisFit <- ctree(Species ~ Sepal.Length + Sepal.Width + Petal.Length
                                        + Petal.Width, data=iris)
plot(irisFit)
```

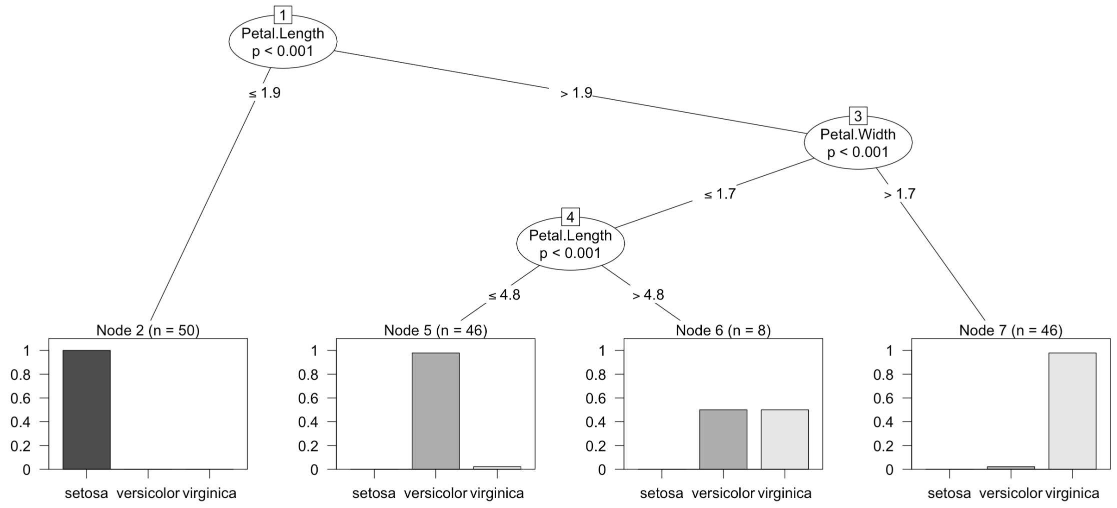


Figure 2: Decision Tree for the iris dataset

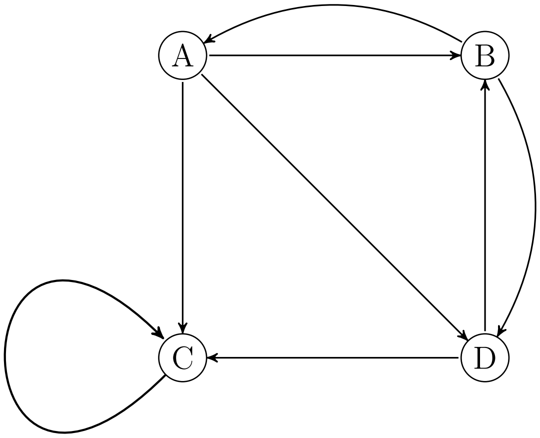
To test the classification accuracy of the decision tree on the training data itself (i.e., on the iris dataset), the following R code was executed, and the corresponding output was obtained. Comment on the accuracy of the classifier in terms of Type-I and Type-II errors in identifying each species.

```
trainPred <- predict(irisFit, iris)
table(trainPred, iris$Species)
```

```
trainPred    setosa versicolor virginica
  setosa         50          0         0
  versicolor      0         49         5
  virginica       0          1        45
```

Problem 3 [25]

Assuming the *random surfer* model of transition in a graph, where the surfer is equally likely to visit any connected node from the current position, create the transition matrix for the graph:



What do you expect to be the output of the following R code, where transMat denotes the transition matrix you created? You do not need to compute the eigenvectors. Justify your answer explaining the connection between the eigenvector transVec with the notion of PageRank in the network.

```
transEig <- eigen(transMat)
transVec <- transEig$vec[,1]
transVec
```

Does the eigenvector transVec provide a rational depiction of PageRank in this network? If so, justify your answer. If not, provide a solution to fix the computation for PageRank in the network.

Problem 4 **[25]**

Suppose that G is an undirected graph consisting of 6 vertices $\{A, B, C, D, E, F\}$, and an unknown number of edges. You are provided with neither the adjacency matrix of the graph, nor with the list of edges in it. However, you are provided with the following piece of R code that was executed on the graph G, and the corresponding output, as follows.

```r
degMat <- diag(degree(G))
adjMat <- get.adjacency(G)
lapMat <- degMat - adjMat
lapEig <- eigen(lapMat)
lapEig
```

```
$values
[1] 4.561553e+00 3.000000e+00 3.000000e+00 3.000000e+00 4.384472e-01 5.313210e-16

$vectors
           [,1]         [,2]          [,3]       [,4]       [,5]      [,6]
[1,]  0.6571923  0.066054535  0.000000e+00  0.5735592 -0.2609565 0.4082483
[2,]  0.1845241 -0.734936838  2.807975e-02 -0.2059434  0.4647051 0.4082483
[3,]  0.1845241  0.668882304 -2.807975e-02 -0.3676157  0.4647051 0.4082483
[4,] -0.1845241 -0.060922638 -7.065490e-01 -0.2835670 -0.4647051 0.4082483
[5,] -0.6571923  0.066054535  7.979728e-17  0.5735592  0.2609565 0.4082483
[6,] -0.1845241 -0.005131897  7.065490e-01 -0.2899922 -0.4647051 0.4082483
```

Interpret the output to take an educated guess about the structure of the graph G, and draw a regular vertex-edge layout of the graph to depict your guess. From your reconstruction of the graph G, find the adjacency matrix (denoted by `adjMat` in the R code), the degree matrix (denoted by `degMat` in the R code), and the corresponding Laplacian matrix (denoted by `lapMat` in the R code).

Good luck! ☺