DATA MINING

TEST 2

INSTRUCTIONS:

- this test consists of 4 questions
- you may attempt all questions.
- maximum marks = 100
- bonus marks available = 10

Question 1 (30 marks)

Learning algorithms can be classified as *supervised* or *unsupervised*.

a) Describe the difference between *supervised* and *unsupervised* learning. [5]

b) What is an R dataframe and what is its role in supervised learning. [5]

c) Discuss the use of training and testing subsets for supervised learning. [5]

d) Explain why the `iris` dataset is popular for testing supervised learning algorithms. [5]

e) Contrast k-means clustering with hierarchical clustering outlining similarities and differences. [10]

Marking schedule

a) In both cases we try to predict the value of one attribute in a dataset observation using information from the other attributes in the observation. In supervised learning, the attribute we are trying to predict is used to train the learning algorithm whist in unsupervised learning the target attribute is hidden from us.

b) A data frame is an enhanced 2d array, (a list of equal length column vectors), with each column representing some attribute and each row representing a measurement of a collection of attributes. both rows and columns can be indexed by names or numbers. various sub-setting routines are available.

c) In supervised learning, to prevent over-learning, the dataset is split into training and testing subsets. (split proportions vary) the algorithm is trained on one subset and evaluated on the other.

d) The iris dataset consists of 5 attributes with the target attribute being `Species`. The task is to predict species from the remaining 4 attributes. There are 3 classes of species, one of which is linearly separable from the other two which are not linearly separable. Thus the dataset is useful for testing linear and non-linear classifiers.

e) Both k-means and hierarchical clustering are unsupervised learning techniques.

k-means is a *top down* clustering algorithm where k clusters are assigned at random and then the algorithm iterates between an assignment phase where observations are assigned to nearest cluster mean and a mean re-computation phase. The algorithm suffers from having to know how many cluster to look for from the start although there are methods to estimate optimum number of clusters

hierarchical clustering is a *bottom up* approach in which all observations start in their own cluster and then nearest clusters are merged. Distances between the updated set of clusters are computed and the merging process continues. The hierarchy is complete when only one cluster remains.

Question 2 (35 marks)

Consider a training set of n observations, $\{\vec{x}_i, y_i\}_{i=1\ldots n}$, where the $\vec{x}_i$ are 2 dimensional feature vectors and $y_i \in \{-1, +1\}$ is the target attribute that places each observation in one of two possible classes.

In linear classification, we seek to divide the two classes by a linear separator in the feature space.

a) Explain with the aid of a diagram, the role played by the parameters b and $\vec{w}$ for a linear separator. [5]

b) Prove that the decision boundary generated by $(\vec{w}, b)$ is identical to the decision boundary generated by $(s\vec{w}, sb)$ for any scaling parameter s. [5]

c) Dr. Magwasa says that the perceptron algorithm is too complicated and instead we should just use the linear classifier $\vec{w} = \vec{u} - \vec{v}$ where $\vec{u}$ is the average of the positive examples and $\vec{v}$ is the average of the negative examples.

Draw a simple sketch of a small linearly separable dataset that will cause Dr. Magwasa's suggestion to fail. [5]

c) Define the *scatter matrix*, S, for the collection of points $\{\vec{x}_i\}_{i=1\ldots n} \in R^2$ and write down the dimensions of S. [5]

d) Explain what is computed by the *action* of the scatter matrix, S, on a *unit vector*, $\vec{w}$. [5]

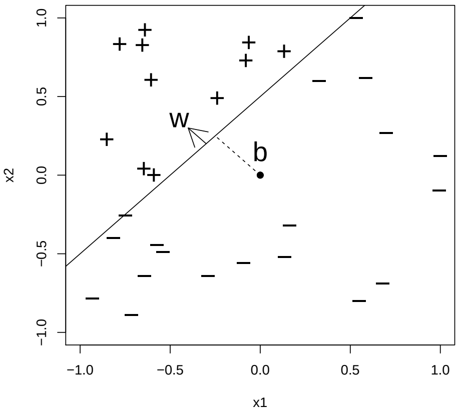
e) R.A. Fisher defined a linear discriminant as the vector $\vec{w}$ that maximizes the ratio:

$$J(\vec{w}) = \frac{\vec{w}^T S_B \vec{w}}{\vec{w}^T S_W \vec{w}}$$

explain in English the roles of the matrices S_B and S_W in the above expression and why it is a good idea to maximize this ratio. [5]

f) What is *Mahalonobis* separation and what role does it play in the construction of a linear discriminant? [10]

Marking schedule



a) In the diagram feature vectors are two-dimensional. We wish to find a line that separates the positive observations form the negative ones. Any line can be specified by a unit normal vector, $\vec{w}$, and a signed perpendicular distance from the origin, b. Classification of a point $\vec{x}$ is then accomplished according to the value of $sign(\vec{x} \cdot \vec{w} - b)$. In the diagram above, a particular $\vec{w}$ and b is given that accomplishes the separation. The perceptron algorithm will discover such a separator. Note that in the above diagram b is negative with respect to the direction of $\vec{w}$.

b) Consider the positive observations at $(1, 4)$, $(1, -2)$ with mean at $(1, 1)$ and negative observations at $(-1, -4)$, $(-1, 2)$ with mean at $(-1, -1)$. According to Dr. Magwasa, the normal to the separating line is now $(1, 1) - (-1, -1) = (2, 2)$ and the best offset is zero which will not separate $(1, -2)$ from $(-1, 2)$.

c) In this case the scatter matrix is a 2×2 matrix defined by $S = \sum_i (\vec{x}_i - \vec{\mu})(\vec{x}_i - \vec{\mu})^T$. Note that the $\vec{x}_i$ are column vectors.

d) $\vec{w}^T S \vec{w}$ is the action of S on $\vec{w}$ and this computes the the *variance* of the projections of the $\vec{x}_i - \vec{\mu}$ onto $\vec{w}$.

e) S_B is the between-class scatter (the scatter of the class means)

S_W is the within-class scatter (computed as the sum of the scatter matrices for each class)

f) Once the optimal separation direction $\vec{w}$ is found, we must project the feature vectors onto $\vec{w}$ and then using the *Mahalonobis* bias b to separate the two classes:

$$b = \frac{\mu_+ \sigma_- + \mu_- \sigma_+}{\sigma_+ + \sigma_-}$$

where the various μ and σ are the means and standard deviations of the two classes.

Assuming that the distribution of the projection onto $\vec{w}$ is double humped in the form of two overlapping normal distributions, this Mahalonobis bias will yield the point of overlap which provides the best offset for the separation plane.

Question 3 (20 marks)

The entropy of a sample D with respect to a target variable of k possible classes is defined as:

$$H(D) = -\sum_{i=1}^{k} P(C_i|D)\log_k(P(C_i|D))$$

a) Explain the term $P(C_i|D)$ and outline how you would go about computing it for a given dataset. [5]

b) Show that if the observations are evenly split amongst all k classes then $H(D) = 1$. [3]

c) Show that if all the observations are from one class then $H(D) = 0$. [2]

d) Define the term *information gain* and outline how it is used in the determination of a single decision split when generating a decision tree under supervised learning. [10]

Marking schedule

a) $P(C_i|D)$ is the probability of class C_i in D and can be computed directly from the dataset according to:

$$P(C_i|D) = \frac{number observations in D with label C_i}{total number of observations in D}$$

b) if the observations are evenly split amongst all k classes then $H(D) = -\sum_{i=1}^{k} \frac{1}{k}\log_k(\frac{1}{k}) = 1$.

c) if all the observations are from one class then $H(D) = 0 \times \log_k(0) + \ldots + 1 \times \log_k(1) + \ldots + 0 \times \log_k(0) = 0$

d) The weighted entropy of a decision/split as follows:

$$H(D_L|D_R) = \frac{|D_L|}{|D|}H(D_L) + \frac{|D_R|}{|D|}H(D_R)$$

and then the **information gain** for the split is:

$$IG(D, D_L, D_R) = H(D) - H(D_L|D_R)$$

In other words IG is the expected reduction in entropy caused by knowing the value a attribute. A decision tree is generated by recursively splitting the data set so as to generate maximum information gain at each split. In this sense it is a greedy algorithm.

Question 4 (25 marks)

The R package, `tm`, provides a suite of routines for undertaking text mining tasks.

a) In the `tm` package text for data mining is commonly stored in a *corpus*. Describe the *corpus* data structure and explain how a corpus is indexed and how it is stored in a directory. [5]

b) Outline 5 procedures that are commonly used for cleaning a corpus. [5]

c) What is a *term-document matrix* and what is it used for? [5]

d) Outline two real life examples, one where you might want to cluster terms from a corpus and the other where you might want to cluster documents from a corpus? [10]

Marking schedule

a) A corpus is a collection of documents. If `myCorpus` is a corpus containing n documents then each document can be accessed via integer indexing, `myCorpus[[i]]`, or named indexing `myCorpus[["ovid_2.txt"]]`. If the corpus is written to disc then all the documents are stored in a directory and each document name is used as that documents filename.

b) Via the `tm_map` function, the tm packages provides common text cleaning operations:

1) `tolower`, transform all characters to lower case.
2) `stripWhitespace`, remove white space
3) `removePunctuation`, get rid of punctuation characters
4) `removeWords, myStopwords`, get rid of common words that may distort later analysis
5) `gsub, pattern="search_string", replacement="repacement_string"`, homegrown cleaning

c) A term-document matrix represents the relationship between terms (or words) and documents, where each row stands for a term and each column for a document, and an entry is the number of occurrences of the term in the document. Correlations, associations, clustering and classification is usually carried out on the term document matrix and not the original corpus.

d)
1) Consider tweets to a controversial hash tag such as `#GuptaGate`, where each tweet is a document in your corpus. A clustering of terms used in such a context might reveal the different conversations taking place and the political affiliations of the tweeters.
2) Consider each work of Shakespeare as a corpus document and each work of Marlow as a corpus document. Combine the two corpora into one corpus and clueter the documents. Hopefully two clusters appear and any disputed work can be attributed to the author of the nearest cluster.