Intelligent data Analysis

Homework #2

Due Date: Tuesday, Sept. 27th, 9PM

Richard Rossi

Please submit all the answers in the form of a single pdf file.

Consider the attached data file for the tasks described below. This dataset is taken from the UC-Irvine Machine Learning Repository (<https://archive.ics.uci.edu/ml/datasets/banknote+authentication>). The dataset has data derived from real and fake banknotes. Four different features are derived by processing the image of each banknote. The fifth column of the dataset contains the class label of 0 or 1 indicating whether the note is fake or real. Our goal is to build a classifier that can learn the model of a real banknote and then predict the class for a new banknote. The dataset contains 1,372 records.

Perform the following tasks:

1. (10) Load the dataset in MATLAB as table named d1 and compute the covariance matrix of the data by using the command: “cm = cov(d1);”. Write all the information that you can infer by interpreting the values in the covariance matrix cm.

cm =

8.0813 4.4051 -4.6663 1.6533 -1.0243

4.4051 34.4457 -19.9051 -6.4900 -1.2974

-4.6663 -19.9051 18.5764 2.8872 0.3340

1.6533 -6.4900 2.8872 4.4143 -0.0245

-1.0243 -1.2974 0.3340 -0.0245 0.2471

??

1. (6) Do a scatter plot of attribute-1 and attribute-2. Use different colors to mark the points from the two different classes. Write your interpretation of the separability of the two classes using attribute1 and attribute2, and also any other insights that you can obtain from this scatter plot.



The data are not linearly separable, but there are clearly two areas for each class. The data do not appear to have a clear relationship (e.g. linear, quadratic, exponential). These attributes are pretty good are separating the data.

1. (6) Repeat #2 above for attribute2 and attribute4.



This set is also not linearly separable, but it looks like the class 0 set is shifted to the right by a couple units compared to class 1. Both classes appear to have a negative quadratic relation. These attributes are mediocre at separating the data.

1. (6) Repeat #2 above for attribute3 and class label (attribute5).



Attribute 3 is not a good candidate the split the data on alone. All of values assigned to class 0 are inside the range of class 1. The only thing you know with attribute 3 is that over a certain value, the record must be class 1. Otherwise, you cannot easily decide.

1. (6) Repeat #2 above for attribute2 and class label (attribute5).



Attribute 2 is a better separator than attribute 3. Each class has a range that it is uniquely in. However, there is a large overlap, so this would still not be a great separator.

1. (6) From the dataset randomly select 1000 records for testing and leave the rest for testing. These selections MUST be random. Use a random number generator function to generate 1000 random numbers between 1 and 1372, and then choose records with these indices for training. Show the code used for selecting the training set of records.

%% Get random training data

train\_idx = randperm(length(d1)); % Random permutation of 1 to length(d1)

train\_idx = train\_idx(1:1000); % Pull out first 1000 random integers

training\_set = d1(train\_idx,:);

1. (20) Use the training set to learn decision trees (use fitctree command) by varying the parameter that controls the minimum number of records in a leaf node. For this parameter use the values of 5, 25, and 50. For each generated tree show its graph schematic (print it from Matlab), and report: (i) number of leaf nodes, (ii) the longest path from the root to a leaf node, its population, and its class purity.
   1. Min leaf node size = 5
      1. Number of leaf nodes:
      2. Longest path from root to a leaf node:
      3. Population:
      4. Class Purity:
   2. Min leaf node size = 25
      1. Number of leaf nodes:
      2. Longest path from root to a leaf node:
      3. Population:
      4. Class Purity:
   3. Min leaf node size = 50
      1. Number of leaf nodes:
      2. Longest path from root to a leaf node:
      3. Population:
      4. Class Purity:
2. (40) Test each of the trees generated in #7 with the test dataset and generate (and show) the confusion matrix. For each tree compute the accuracy, precision, and recall values. Give an interpretation of these performance numbers individually and also in the context of changes to these numbers as the size of the decision tree changes.