Data Mining Project 1: Data Preprocessing

Released on May 25
Due on June 10 at 11:55pm

Specification

In this project, the students are to write programs to apply data pre-processing and feature selection techniques to gene expression datasets for cancers/diseases.

An example of gene expression datasets is the colon cancer dataset, which contains 62 samples collected from colon-cancer patients; among those samples, 40 are tumor biopsies labeled as "positive" and 22 are normal tissue biopsies labeled as "negative". For simplicity, we assume that each dataset has just two classes in this project. Each tuple (row) in the data consists of the readings for the genes, and the class (which is the last column). Each gene is an attribute. The columns are separated by ",", which is a commonly used format in data mining. The dataset can be found on pilot under "Projects" called p1data.txt. Other datasets may be provided under "Projects" later.

Note: Your program will need to be able to handle other datasets. That means that your program will need to go through the data once to determine the number of samples/rows and the number of attributes.

In your discussions and reports, refer to the genes as g1, ..., gN, in the left-to-right order as given in the original data file.

Your project should address the following tasks:

Task 1. Use a method implemented in weka to select K features. This is better done either using procedure call to weka inside your program; a separate procedure call outside your program can be used with a small loss of marks. Information about weka can be found at
https://svn.cms.waikato.ac.nz/svn/weka/trunk/weka/
Indicate which method you used in your report.

Task 2. Implement the entropy-based method to discretize all genes and to select the top K genes with highest information gain, listed in decreasing information gain order. Use 3 bins for each gene for this task. So you will need to pick one of the two bins computed by the first split point for the second split.

Task 3. Compute the correlation coefficients between the numerical-valued genes selected by Task 1 and the genes selected by Task 2. List the K^2 pairs of genes in increasing similarity order.

Students should first develop their code by working with a small number of genes.

Your program should ask the user to give the name of the input file (containing the gene expression data, which is assumed to be located in the same folder of the executable). The executable should be called **PoneDP**. It should produce the following output files: topkfeatures1.txt (for the selected genes produced for Task 1), topkfeatures2.txt (for the selected genes produced for Task 2), entropybins.txt and entropydata.txt (produced for Task 2), correlationgenes.txt (produced for Task 3).

Your program should get the input file name and the value of K as two command line parameters.

In the entropybins.txt file, you should have the following information for each of the K genes: gene number, the information gain, (bin_1_lb, bin_1_ub]: bin_1_count in positive and bin_1_count in negative; plus similar info for bin2 and bin3. Have one gene per line and have the genes ordered in decreasing information gain order. The information gain should be weighted average involving the three bins.

The entropydata.txt file should contain the discretized data; the genes should be re-ordered based on information gain.

The first line below is an example line for the entropybins.txt file, and the next line is an example line for the entropydata.txt file.

```
G1250: Info Gain: 0.435072; Bins: (-,35.959], 6, 5; (35.959, 95], 8, 2; (95,+)  20,10

b, a, c, a, c, b, c, a, a, c, c, a, a, b, a, a, b, a, a, b, negative
```

In the correlatedgenesdata.txt file, you should have one line for each pair of the genes; each line contains the two genes and the associated correlation coefficient.