EECS 4412 Data Mining
Assignment 2 Report

Name: Chun Ho Li
Student ID: 214908800
Date: 22 March 2021

1. Cleaning

All of the datasets need to be ARFF format. First, I just manually merge the .data and .names files with the correct formatting before cleaning or handling any missing data. After loading the dataset into the WEKA explorer, there are few things to be preprocessed. First, I apply min-max normalization filter to all datasets and all attributes, since their min and max value are not in the same range. Second, if the datasets have continuous attributes, I used the discretize filter to smooth out the data by binning. lastly, if there are dataset with missing class instance, I just remove the class as there is no effect on the accuracy or performance.

1.1. Ecoli
- Id column has to be drop
- Apply min-max normalize filter

1.2. Iris Plant
- Apply min-max normalize filter

1.3. Glass Identification
- Id column has to be drop in order to be processed by WEKA
- Since class 4 has no instance, it has no effect on the accuracy, so I can just remove class 4 from the labels
- Apply min-max normalize filter

1.4. Yeast
- Drop the first column
- Apply min-max normalize filter

1.5. Gait
- Apply min-max normalize filter
- Apply discretize filter on skewed or unbalanced data
- Assign attribute names and class name to data in order to be processed by WEKA in ARFF format
- Use attribute selection filter to reduce the number of attributes

2. Method Utility

a) Error rate

Datasets / Methods	Ecoli	Iris	Glass	Yeast	Gait	AVERAGE
C4.5	15.7738%	4%	32.71%	43.8679%	41.6667%	27.60%
PIPPER	19.6429%	4.6667%	32.71%	42.1833%	54.1667%	30.67%
Naïve Bayesian Classification	13.6905%	5.333%	50.4673%	42.5202%	37.5%	29.90%
Bayesian Network	18.75%	7.333%	25.2336%	43.2615%	2.0833%	19.33%
k-Nearest Neighbor	19.6429%	4.6667%	29.4393%	47.7089%	4.1667%	21.12%
Neural Networks	13.9881%	2.6667%	32.7103%	40.566%	14.5833%	20.90%

b) Rank

1. Bayesian Network
2. Neural Network
3. k-Nearest Neighbour
4. C4.5
5. Naïve Bayesian Classification
6. PIPPER

i. Bayesian Network and Neural Network are the top two model with the lowest error rate; however, it is not significantly different compare to other methods. One reason might be from the dataset, these datasets are pretty similar in some way, they don't have a lot of attributes and class, and they are overall not really complex. Hence, some of the advantage of using a method might not show through with these datasets.

ii. The lowest error rate is the Bayesian Network, and the highest error rate is the PIPPER, though the different is only around 11%. When the structure of the DAG is not known, it works pretty much the same as a rule learning method, while Bayesian Network works by combine prior knowledge with observed data, rules learning PIPPER might suffer from overfitting problem more easily trying to cover all example with rules.

3. Dataset Differences

a) The Iris dataset has a significant better accuracy compare to all other datasets. Looking at the dataset, it has fewer classes, and each class has the same number of instances. This is a very balanced dataset, having enough entries and weight for each class, hence, the training for the classifier can be more balance as well, result in an accurate model.

b) Bayesian Network has a significant different error rate compare to other method specifically in the Gait dataset. I surmise this is due to a huge selection of attributes which are correlated. Since other method does not take account of the dependencies exist among attribute such as the Naïve Bayesian Classifier and PIPPER, they have a much higher error rate, hence, I believe that the attribute dependencies take a huge role in error rate with these datasets.

4. Accuracy Improvement

I am able to improve the accuracy of the K-Nearest Neighbour model with Ecoli dataset from 19.6429% to 11.9048% by tuning two parameters.

a) KNN: The default number of neighbours to use is 1, I slowly tune the number from 1 to 10, I can see the best accuracy with KNN setting to 9.
b) Distance Weighting: The default has no distance weighting; I am able to achieve better result with 1/distance weighting method.

The K value has a huge effect on how the model will adapt the noise of the data, since each instance is being classified by the nearest K neighbours. The improvement solely from tuning this parameter is significant compare to the distance weighting parameter, it went from 19.6429% to 12.5%. When K = 1, the model tends to adapt all noise from the data, an instance is being classified based on the nearest 1 neighbour, however, with a larger K value, the model is able to reduce the effect of noisy data, an instance will be classified by the majority 9 neighbours. Hence, I suppose that the Ecoli dataset has some noisy data and increasing the K value is able to get better accuracy.

The distance weighting takes account of the distance of the instance and the neighbour, if the neighbour is far, it has less weight and less effect on classifying the instance, in other word, the closer the neighbour to an instance, the more weight or effect on classification. Hence, this can be a fine tuning to addon with tuning the K value. Even with a higher K value, the model is able to account of the distance between neighbours and balance the distinct boundary on classification, hence, giving the model with a little higher accuracy, from 12.5% to 11.9048%.