R•I•T	Rochester Institute of Technology Golisano College of Computing and Information Sciences School of Information

Name: ________________________________

Lab #3: Classification II (updates in Green)

Introduction:

In this lab, we will go through standard classifiers and explore different test options available. You will encounter many situations in Data Mining problems where you need to choose an appropriate algorithm/ model suitable for data. In this lab, you will study some well-known classification algorithms and try to find out suitable models for your dataset.
Before starting the lab, make sure you are familiar with the Weka environment and the necessary procedures, such as loading the data and data preparation, etc.

Submit **a scanned copy of answered this Lab #3** to the Lab #3 DropBox by Sunday, 10/4 @11:59PM.

Background reading: http://bit.ly/2ykAUFj

Note Record classification accuracy in the table provided on page 4 (Table A).

Part I. Partitioning Datasets for Training and Testing

Dataset used: *diabetes* dataset (Check data folder under Weka installation)

Cross-Validation:

In this model validation method, the dataset is divided into k subsets (these are called "folds" and k< number of instances). It performs k number of iterations with k-1 subsets as training set and the remaining one as the test set. All the iterations are averaged, which can be seen in the Weka output window after performing this task.

- Follow steps performed to load data to Weka from Lab 2 (Note: Use *diabetes* dataset)
- Navigate to the classify tab, then choose the J48 classifier and select cross-validation with ten folds with all other default options.
- Click on Start and observe the results on the right pane.
- Similarly, perform the 10-fold Cross-validation for NaiveBayes (in Bayes folder) and jRip (in rules folder).

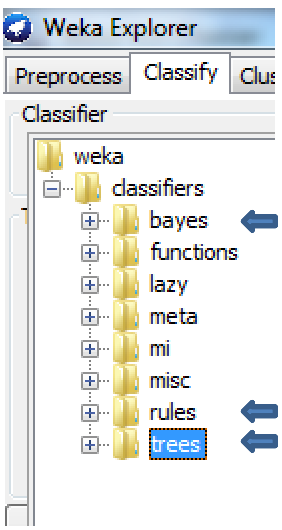
- Record classification accuracy in **Table A** (page 4).

Percentage Split:

In this test method, the data is split into two parts. First, n% of the dataset is reserved for training data to build the model and the remaining (100-n)% as test data.

Follow the steps performed in Lab2 to load data and perform analysis using three different classification algorithms:

- NaiveBayes (in Bayes folder)
- J48 (in trees folder)
- jRip (in rules)



For each algorithm, use two test options:

- Training set
- The percentage split ~~66%~~ 75%

Record classification accuracies in **Table A** (page 4).

R•I•T	Rochester Institute of Technology Golisano College of Computing and Information Sciences School of Information

<u>Sample output from a Weka classification model:</u>

Classifier output

```
Number of Leaves  :     3

Size of the tree :      5


Time taken to build model: 0.05 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          42               73.6842 %
Incorrectly Classified Instances        15               26.3158 %
Kappa statistic                          0.4415
Mean absolute error                      0.3192
Root mean squared error                  0.4669
Relative absolute error                 69.7715 %
Root relative squared error             97.7888 %
Coverage of cases (0.95 level)          91.2281 %
Mean rel. region size (0.95 level)      85.9649 %
Total Number of Instances               57

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.700    0.243    0.609      0.700    0.651      0.444    0.695     0.559     bad
                 0.757    0.300    0.824      0.757    0.789      0.444    0.695     0.738     good
Weighted Avg.    0.737    0.280    0.748      0.737    0.740      0.444    0.695     0.675

=== Confusion Matrix ===

  a  b   <-- classified as
 14  6 |  a = bad
  9 28 |  b = good
```

Classification accuracy

Confusion Matrix

The essential numbers that you should focus here are:
Correctly classified Instances, Incorrectly Classified Instances, and the confusion matrix.

Write down classification accuracy (in percent; rounded to the nearest integer) obtained in the table below:

Table A

Test Options	**Training set**	**Percentage Split 75%**	**10-fold cross-validation**
J48			
NaiveBayes			
jRip			

Q1.1 Compare using training set and percentage split approaches. Did the accuracy increase? Comment on the changes.

Q1.2 Based on the results obtained, which classifier will be best for this dataset? Support your answer.

Q1.3 What is the significance of the number of folds 10 in cross-validation?

Q1.4 Assume a dataset consists of 36instances, what is the size of test data used for:

1) 9-fold cross validation - ________________

2) 75% percentage split - _________________

Part 2. Nearest-Neighbor Classifiers

Datasets used: *breast-cancer* & ***glass*** (Check data folder under Weka installation)

Open the ***breast-cancer*** dataset and click the Classify tab.

Q2.1 Select a lazy classifier, IBK. What is its accuracy in percent evaluated using 10-fold cross-validation?

____

Q2.2 IBK's KNN parameter specifies the number of nearest neighbors to use when classifying a test instance, and the majority vote determines the outcome. The default value is 1. Evaluate KNN's performance with 2, 3, and 5 nearest neighbors. What accuracies do you get in percent?

K=2: _____________ K=3: _____________ K=5: _____________

Q2.3 Do you think these accuracy differences (Q2.2) are significant? Support your answer by running IBK with the default value of 1 for KNN using random number seeds: 1,2,3,4,5 recording the accuracies: (Click the "More options …" in Test options to enter a random seed)

Seed #1: ______________ Seed #2: ______________ Seed #3: ______________
Seed #4: ______________ Seed #5: ______________

Are the differences significant? Yes/No

Q2.4 One of the issues with IBK is how to choose a suitable value for the number of nearest neighbors. If it is too small, the method is susceptible to noise in the data. If it's too large, it covers a too great area of instance space.

Let's simulate by adding noise artificially to the ***glass*** dataset to determine the nearest neighbor's optimal value.

- Open the ***glass*** dataset.
- Select the unsupervised attribute filter ***addNoise*** (see Figure 1 below).
- Observe from its configuration panel that it adds 10% noise to the *last* attribute by default, which is the class. You will change this value to add 20 & 30% to test and ***Apply*** the filter (see Figure 2 below).
- Now run the IBK to find the optimal number of neighbors as you add noise. When you run IBK with added noise, change ***crossValidate*** to True, and 20 is a safe value

to use for KNN in these experiments (Figure 3). Don't forget to ***Undo*** the effect of the ***addNoise*** filter after each experiment.

- What is the best number of neighbors for each? You can find it in the "Classified model" section of the output.

Noise 0%: _________ 10%: __________ 20%: __________ 30%: _________

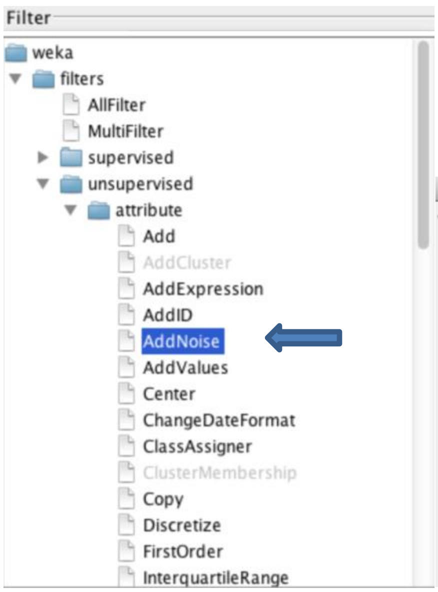


Figure 1

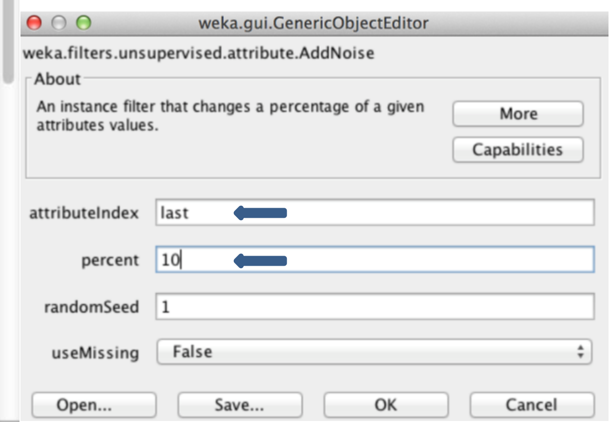


Figure 2

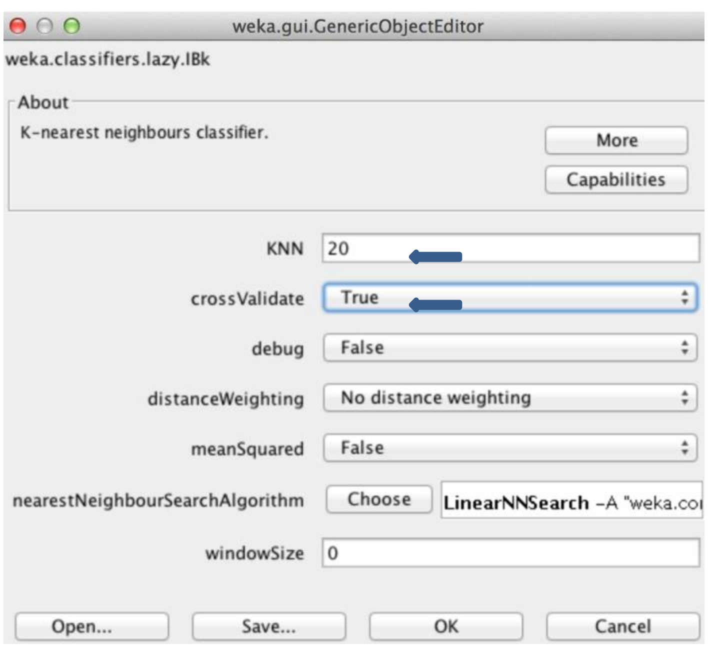


Figure 3

Part 3. Naïve Bayes Approach

Consider the dataset shown in Table B.

Table B

Record	A	B	C	Class
1	0	0	0	Y
2	0	0	1	N
3	0	1	1	N
4	0	1	1	N
5	0	0	1	Y
6	1	0	1	Y
7	1	0	1	N
8	1	0	1	N
9	1	1	1	Y
10	1	1	0	Y

Q3.1 Estimate the conditional probabilities below. Apply the Laplacian smoothing if applicable (show all calculations)

P(A|Y): P(A=1|Y)= P(A=0|Y)=

P(B|Y): P(B=1|Y)= P(B=0|Y)=

P(C|Y): P(C=1|Y)= P(C=0|Y)=

P(A|N): P(A=1|N)= P(A=0|N)=

P(B|N): P(B=1|N)= P(B=0|N)=

P(C|N): P(C=1|N)= P(C=0|N)=

Q3.2 Use the Estimate of conditional probabilities given in the previous question (Q3.1) to predict the class label for a test sample (A = 0, B = 1, C = 0) using the Naïve Bayes approach. (Show all calculations)

R•I•T	Rochester Institute of Technology Golisano College of Computing and Information Sciences School of Information

ISTE-600 LAB 3 Gradesheet Name: ______________________________

Graded By: ______________________________

Questions	Max. Points	Points Earned	Comments
Part 1			
Table A	10		
Q1.1	5		
Q1.2	5		
Q1.3	10		
Q1.4	5		
Part 2			
Q2.1	10		
Q2.2	10		
Q2.3	10		
Q2.4	15		
Part 3			
Q3.1	10		
Q3.2	10		
TOTAL	100		