Assignment 7: Logic-based Approaches

Yichao Chen (yic 85) & Jingyi Lin (jil173)

Introduction

There are two sets of data, one is iris, which provided some basic information of each kind of iris flowers with all numeric data, and is needed to classify its iris class. Another data set is house votes, with different attitude on various issues. The data are binary, and the belonging party of the votes should be classified.

In this analysis report, we use Weka to generate the classification models, and use the 10-fold validation method to test the models.

Data set I: Iris

This data set has 150 rows of data with no missing values. So the clean data process can be omitted.
Firstly, we classify the data by various rules. Following is a summary of the accuracy of each approach.

Approach	*Decision Table*	*NNge*	*JRip*	*PART*
Accuracy	96%	96%	94%	94%

- *Decision Table* is for building and using a simple decision table majority classifier, and changing the parameters that to choose the search type of genetic search, it generates the highest accuracy.
- *NNge* is a nearest-neighbor-like algorithm using non-nested generalized exemplars, and its accuracy remains unchanged after modifying its parameters.
- *JRip* implements a propositional rule learner, changing parameters cannot impact much on the accuracy.
- *PART* uses separate-and-conquer for generating a PART decision list.

Following example is the run information and output of NNge model:

```
Classifier output

=== Run information ===

Scheme:weka.classifiers.rules.NNge -G 5 -I 5
Relation:     iris.txt
Instances:    150
Attributes:   5
              SL
              SW
              PL
              PW
              class
Test mode:10-fold cross-validation

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

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.0267
Root mean squared error                  0.1633
Relative absolute error                  6      %
Root relative squared error             34.641  %
Total Number of Instances              150
```

Secondly, we classify the data by different kinds of trees. Following is a summary of the accuracy of each approach.

Approach	*Random Tree*	*LMT*	*FT Tree*	*J48graft*	*LAD Tree*
Accuracy	95.33%	97.33%	96.67%	95.33%	94%

- *Random Tree* constructs a tree considering K randomly chosen attributes at each node. We use a 5 kvalue and 10 folds model to give the mentioned accuracy.
- *LMT* builds classification trees with logistic regression functions at the leaves. Set split on residuals and use AIC to True, and set the weight trim beta to 0.2, the accuracy will change from 94% to 97.33%.
- *FT Tree is* a classifier for building 'Functional trees', which are classification trees that could have logistic regression functions at the inner nodes. The change of parameters will cause the decrease of accuracy.
- *J48graft* is a class for generating a grafted C4.5 decision tree. The unpruned tree gives the highest accuracy of 95.3%.
- *LAD Tree* is used for generating a multi-class alternating decision tree using the LogicBoost Strategy. The modification of number of boosting iteration cannot change the model accuracy.

Following example is the comparison run information and output of LMT model before and after parameter modifications:

Before modification:

```
Classifier output

=== Run information ===

Scheme:weka.classifiers.trees.LMT -I -1 -M 15 -W 0.0
Relation:     iris.txt
Instances:    150
Attributes:   5
              SL
              SW
              PL
              PW
              class
Test mode:10-fold cross-validation

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

Correctly Classified Instances         141               94      %
Incorrectly Classified Instances         9                6      %
Kappa statistic                          0.91
Mean absolute error                      0.0439
Root mean squared error                  0.1542
Relative absolute error                  9.8675 %
Root relative squared error             32.7159 %
Total Number of Instances              150
```

After modification:

```
Classifier output

=== Run information ===

Scheme:weka.classifiers.trees.LMT -R -I -1 -M 15 -W 0.2 -A
Relation:     iris.txt
Instances:    150
Attributes:   5
              SL
              SW
              PL
              PW
              class
Test mode:10-fold cross-validation

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

Correctly Classified Instances         146               97.3333 %
Incorrectly Classified Instances         4                2.6667 %
Kappa statistic                          0.96
Mean absolute error                      0.0395
Root mean squared error                  0.1243
Relative absolute error                  8.8907 %
Root relative squared error             26.3771 %
Total Number of Instances              150
```

Data set II: House-Votes-84

This data set holds 435 instances with discrete elements. There is no missing value as well. There are 16 voting issues influence the classification of the party refers to either republican or democrat.
Similary, at first we classify the data by various rules. Decision Table, NNge, JRip, and PART, which are approaches used in former data test, can still be used in this data set since they are suitable methods for discrete data. The descriptions of each approach are omitted.
Following is a summary of the accuracy of each approach.

Approach	*Decision Table*	*NNge*	*JRip*	*PART*
Accuracy	95.63%	94.7%	96.1%	96.32%

- In *Decision Table*, when using the default search method of Best First Search, the accuracy is 93.33%, while after modifying the search method to Rank Search, the accuracy rise to 95.63%.
- In *NNge*, the modification of generalization attempt numbers and mutual information folder numbers will not affect the accuracy.
- In *JRip*, the default set has the accuracy of 95.8%. The change of some parameters, the accuracy can reach 96.1%.
- In *PART*, the minimum number of instances from 2 to 10, and the number of folds from 3 to 5, the accuracy can change from 96.86% to 96.32%.

Following example is the comparison run information and output of Decision Table model before and after changing the search pattern:

Before modification: (Use BestFirst search)

```
Classifier output

=== Run information ===

Scheme:weka.classifiers.rules.DecisionTable -X 1 -S "weka.attributeSelection.BestFirst -D 1 -N 5"
Relation:     house-votes-84.txt
Instances:    435
```

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

Correctly Classified Instances         264               60.6897 %
Incorrectly Classified Instances       171               39.3103 %
Kappa statistic                          0.0335
Mean absolute error                      0.3494
Root mean squared error                  0.4204
Relative absolute error                 96.8549 %
Root relative squared error             99.075  %
Total Number of Instances              435
```

After modification: (Use RankSearch search)

```
Classifier output

=== Run information ===

Scheme:weka.classifiers.rules.DecisionTable -X 1 -S "weka.attributeSelection.RankSearch -S 1 -R 0 -A weka.attributeSelection.GainRatioAttributeEval --"
Relation:     house-votes-84.txt
Instances:    435


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

Correctly Classified Instances         416               95.6322 %
Incorrectly Classified Instances        19                4.3678 %
Kappa statistic                          0.9088
Mean absolute error                      0.0837
Root mean squared error                  0.2004
Relative absolute error                 17.6399 %
Root relative squared error             41.1559 %
Total Number of Instances              435
```

Second, we classify the data by different kinds of trees. Following is a summary of the accuracy of each approach.

Approach	*ADTree*	*BFTree*	*DecisionStump*	*LMT*	*FT Tree*
Accuracy	95.86%	95.63%	95.6%	96.78%	97.01%

- *ADTree*'s accuracy becomes even lower after modifying the parameters.
- *In BFTree*, the accuracy can change from 95.17% to 95.63% after modifying the default minimum number of objects from 2 to 10.
- Default value of parameters in *DecisionStump* can generate the highest accuracy.
- Similar to DecisionStump, *LMT* does not need further modifications.
- In *FT Tree*, the minimum instances is changed from 10 to 112, the accuracy increased from 96.78% to 97.01%.

Following example is the comparison run information and output of BFTree approach before and after parameter modifications:

Before modification: (Use minNumObj = 2)

```
Classifier output

=== Run information ===

Scheme:weka.classifiers.trees.BFTree -S 1 -M 2 -N 5 -C 1.0 -P POSTPRUNED
Relation:     house-votes-84.txt
Instances:    435
```

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

Correctly Classified Instances         414               95.1724 %
Incorrectly Classified Instances        21                4.8276 %
Kappa statistic                          0.8985
Mean absolute error                      0.0677
Root mean squared error                  0.2043
Relative absolute error                 14.2813 %
Root relative squared error             41.9498 %
Total Number of Instances              435
```

After modification: (Use minNumObj = 10)

```
Classifier output

=== Run information ===

Scheme:weka.classifiers.trees.BFTree -S 1 -M 10 -N 5 -C 1.0 -P POSTPRUNED
Relation:     house-votes-84.txt
Instances:    435


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

Correctly Classified Instances         416               95.6322 %
Incorrectly Classified Instances        19                4.3678 %
Kappa statistic                          0.9088
Mean absolute error                      0.0821
Root mean squared error                  0.2081
Relative absolute error                 17.306  %
Root relative squared error             42.736  %
Total Number of Instances              435
```