Bài tập thực hành:

Weka với Java trong Eclipse

Thời gian làm bài: 1 tuần (Xem dealine trong link nộp bài trên moodle)

Nộp bài:

- Nộp bài lên moodle.
- Đặt tên bài nộp theo định dạng MSSV.rar. Trong đó bao gồm:
  - Tập tin báo cáo.
  - Các tập tin dữ liệu theo yêu cầu của bài tập
- Nếu sử dụng code trên Internet, trước đoạn code sử dụng phải có chú thích đường dẫn đến trang web chứa đoạn code. Trường hợp 2 bài có đoạn code giống nhau, bài không chú thích đường dẫn đến trang web chứa source code sẽ bị tính là gian lận, và bị 0 điểm.

Các hành vi sử dụng toàn bộ/một phần bài làm của người khác sẽ bị 0 điểm cho toàn bộ phần thực hành

1 Thêm Weka vào project Eclipse

1. Tạo project Java trong Eclipse
2. Click chuột phải lên project, chọn Properties
3. Chọn mục Java Build Path
4. Bấm vào nút Add External JARs
5. Chọn file weka.jar trong thư mục cài Weka

2 Đọc dữ liệu

```
import weka.core.converters.ConverterUtils.DataSource;
...
DataSource source = new DataSource("/some/where/data.arff");
Instances data = source.getDataSet();
```

3 Chọn thuộc tính làm nhãn (class)

```
if (data.classIndex() == -1)
   data.setClassIndex(data.numAttributes() - 1);
```

4 Thiết lập tham số

4.1 Bằng tay

```
String[] options = new String[2];
options[0] = "-R";
options[1] = "1";
```

4.2 Sử dụng splitOptions

```
String[] options = weka.core.Utils.splitOptions("-R 1");
```

5 Filter

```
 import weka.core.Instances;
 import weka.filters.Filter;
 import weka.filters.unsupervised.attribute.Remove;
 ...
 String[] options = new String[2];
 options[0] = "-R";                                    // "range"
 options[1] = "1";                                     // first attribute
 Remove remove = new Remove();                         // new instance of
filter
 remove.setOptions(options);                           // set options
 remove.setInputFormat(data);                          // inform filter about
dataset **AFTER** setting options
 Instances newData = Filter.useFilter(data, remove);   // apply filter
```

6 Phân lớp

```
 import weka.classifiers.trees.J48;
 ...
 String[] options = new String[1];
 options[0] = "-U";            // unpruned tree
 J48 tree = new J48();         // new instance of tree
 tree.setOptions(options);     // set the options
 tree.buildClassifier(data);   // build classifier
```

7 Đánh giá

```
 import weka.classifiers.Evaluation;
 import java.util.Random;
 ...
 Evaluation eval = new Evaluation(newData);
 eval.crossValidateModel(tree, newData, 10, new Random(1), new Object[] {});
```

8 Phân lớp dữ liệu mới

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import weka.core.Instances;
...
// load unlabeled data
Instances unlabeled = new Instances(
                      new BufferedReader(
                        new FileReader("/some/where/unlabeled.arff")));

// set class attribute
unlabeled.setClassIndex(unlabeled.numAttributes() - 1);

// create copy
Instances labeled = new Instances(unlabeled);

// label instances
for (int i = 0; i < unlabeled.numInstances(); i++) {
  double clsLabel = tree.classifyInstance(unlabeled.instance(i));
  labeled.instance(i).setClassValue(clsLabel);
}
// save labeled data
BufferedWriter writer = new BufferedWriter(
                        new FileWriter("/some/where/labeled.arff"));
writer.write(labeled.toString());
writer.newLine();
writer.flush();
writer.close();
```

9 Bài tập

Sử dụng Weka và Java, chạy thuật toán phân lớp NaiveBayesSimple và Id3, với cách đánh giá crossValidation với 10 folds, trên bộ dữ liệu contact-lens.arff. Xuất ra màn hình Console độ chính xác(%) của các phương pháp phân lớp.

Tài liệu tham khảo:

[1] http://stackoverflow.com/questions/3280353/how-to-import-a-jar-in-eclipse

[2] https://weka.wikispaces.com/Use+WEKA+in+your+Java+code