

CSCI 5702/7702 - Data Mining and Analytics

Fall 2015 - Assignment 4

Due: 12/04/2015 8:59pm

Project Objective

Your objective in this assignment is to build a general purpose classification framework from scratch. This framework contains two classification methods: a **basic method** and an **ensemble method**. Given certain training datasets following the **LIBSVM** data format, your classification framework should be able to generate a classifier, and use this classifier to assign labels to unseen test data instances.

Important notes before you start

- This is an extensive assignment and takes more time to do than previous assignments, so please **start early**. The deadline cannot be extended!
- This is an **individual assignment**, which means it is OK to discuss with your classmates, but it is NOT OK to work together or share code. All codes will be checked manually and by using the MOSS script.
- Similar libraries or programs of classification methods can be found online, but you are **prohibited from using these resources**, which means you **cannot include public libraries**, or **modify existing programs**, since the purpose of this programming assignment is to help you understand and implement classification algorithms step by step. You need to develop your own code from scratch.
- You should use **Java** as your programming language for this assignment; if you would like to use **C++** or **Python** as your programming language, you need to email the TA to obtain instructions; otherwise, your submission will not be graded.
- You may be asked to attend in the grading session and run your program in person.

Project Description

Roadmap for implementing this assignment

1. Know Your Data!
2. Download and install a **JAVA** IDE, such as **Eclipse** or **NetBeans**.
3. Implement your classification framework:
 - A. Read in a training dataset and test dataset, and store it in memory.
 - B. Implement a **basic classification method**, which includes both **training** process and **test** process. Given a training dataset, the classification method should be able to construct a classifier correctly and efficiently. Given a test dataset, the classification method should be able to predict the label of the unseen test instances with the aforementioned classifier.
 - C. Implement an **ensemble classification method** using the basic classification method in the previous step. The ensemble classification method should also be able to **train** a classifier and **predict** labels for the unseen data instances.
4. Test both the basic classification method and the ensemble method with all datasets provided, evaluate their performances, and write a report describing your analysis of the results.
5. Submit your assignment on Canvas.

1. Know Your Data

There are 3 datasets in the assignment package under the “[Datasets](#)” folder which are provided by the [UCI Machine Learning Repository](#). Notice that, these datasets are partitioned into training and test partitions for you.

All 3 dataset follow the [LIBSVM format](#):

`<label> <index1>:<value1> <index2>:<value2> ...`

- Each line contains an instance and is ended by a '\n' character.
- **<label>** is an integer indicating the class label (we only consider **two-class classification** in this assignment).
- The pair **<index>:<value>** gives a feature (attribute) value: **<index>** is an integer starting from 1 and **<value>** is a **number** (we only consider **categorical attributes** in this assignment).
- Following the LIBSVM format, if the value of an attribute is 0, we omit this attribute in the dataset. It does not mean the attribute is missing.
- Be careful about not using label information as an attribute when you build classifier, in which case you can get 100% precision easily, but **YOU WILL SCORE 0!**

Name	Attributes	Training	Test	Labels
breast_cancer	9	180	105	<ul style="list-style-type: none">• +1 recurrence-event• -1 no-recurrence-event
led	7	2087	1134	<ul style="list-style-type: none">• +1 is 3• -1 is other numbers
poker	10	1042	677	<ul style="list-style-type: none">• +1 one pair• -1 three of a kind

Table 1 – Datasets Properties

- **Name:** name of the dataset. To obtain more information about each dataset click on its name.
- **Attributes:** the number of attributes.

- **Training:** the number of records in the training dataset.
- **Test:** the number of records in the test dataset.
- **Labels:** the property of classes.

2• Download and Install a JAVA IDE

You could use any standard JAVA IDE of your preference. You are not limited to the Eclipse IDE.

For Eclipse:

1. Download **JAVA JDK** from [here](#).
2. Install JAVA JDK on your machine. You could find installation tutorials for different operating systems below:
 - [Windows](#)
 - [Linux \(Ubuntu\)](#)
 - [Mac OS](#)
3. You can download the **Eclipse** IDE (JAVA developer edition) from [here](#).

3· Implement classification Algorithms (60 points)

In this step, you will implement a basic classification and an ensemble classification method. You can choose one of the two following options to implement:

- A. **Decision Tree (C4.5)** as basic method, **Random Forest** as the ensemble version of the classification method.
- B. **Naive Bayes** as basic method, and use **AdaBoost** as ensemble method.

- Your classification methods should contain two steps, **training** (building a classifier) and **test** (assigning labels to unseen data instances).

3·1· Input

Assume the following inputs for your application:

- **Training dataset location:** the location of the training dataset file. You need to define it as a **constant** in your main class. Please note that only one dataset (among 3) will be used at a time.
- **Test dataset location:** the location of the test dataset file. You need to define it as a **constant** in your main class. Please note that only one dataset (among 3) will be used at a time, i.e., the one corresponding to the chosen training dataset.
- **Number of attributes:** the number of attributes in the dataset.
- **Output location:** your application must generate and save all output files in this directory inside of your application directory named **"Output"**.

3·2· Output:

Your application needs to generate the following files:

- **Basic – Dataset X.txt:** your application must generate a separate text file (named Basic - Dataset X.txt) for each dataset "X". The text file must include the following:
 - The performance of your classifier in the following format:
 - The total number of instances in the test dataset.
 - The number of true predictions.
 - The number of false predictions.
 - The accuracy value in percentage.
 - Test data in the **LIBSVM format** (including only the predicted label by your algorithm not the correct label provided in the original test

data). **Include only one instance per line.** At the end of each line put an F (T) if the predicted label was false (true).

- *Ensemble - Dataset X.txt*: your application must generate a separate text file (named *Ensemble - Dataset X.txt*) for each dataset “X”. The text file must include the following:
 - The performance of your classifier in the following format:
 - The total number of instances in the test dataset.
 - The number of true predictions.
 - The number of false predictions.
 - The accuracy value in percentage.
 - Test data in the **LIBSVM format** (including only the predicted label by your algorithm not the correct label provided in the original test data). **Include only one instance per line.** At the end of each line put an F (T) if the predicted label was false (true).
- ❖ **Note:** Your output should follow the same **format** as the example output files located under the “**Output**” folder of the assignment 4 package; otherwise you WILL NOT receive any points for this assignment.

4. Evaluation (40 points)

You should prepare an evaluation report and include the following sections in your report:

- A. The classification methods used in your classification framework.
Evaluation of the following measures for each algorithm/dataset combination (i.e., 2 classification algorithms \times 3 dataset = 6 sets of evaluation measures):
- Accuracy (total number of instances, the number of true and false predictions, accuracy in the percentage formant).
 - Error Rate
 - Sensitivity
 - Specificity
 - Precision
 - F-1 Score
 - F_β score ($\beta = 0.5$ and 2)
- B. Your conclusion on whether the ensemble method improves the performance of the basic classification method you chose, why or why not? How much was the amount of improvement (in percentage) for each dataset? Which datasets gained the highest and lowest improvements and why?
- C. For each classification, mention the dataset which had the highest accuracy. Justify your answer.

5. Submission Guideline

You must submit your assignment through **Canvas** before the deadline; late submissions are not accepted! You are allowed to submit your assignment multiple times, but only the last submission (**before the deadline**) will be recorded and graded.

Your submission should be a single **zip** file named **<Your CU Denver Portal ID>-A4.zip**. For example, if your CU Denver Portal ID is "john", the file name would be: **john-A4.zip**.

Your submission must contain the following materials:

- ❖ **Code directory**: which contains your classification framework code.
- ❖ **Output directory**: which contains the following 6 files:
 - **Basic – breast cancer.txt**
 - **Basic –led.txt**
 - **Basic – poker.txt**
 - **Ensemble – breast cancer.txt**
 - **Ensemble –led.txt**
 - **Ensemble –poker.txt**
- ❖ **Report.pdf**

Notes:

- ❖ If your output is not correct, you will not receive any points for your code.
- ❖ Please download your assignment after submission and make sure it is not corrupt. We won't be responsible for the corrupted submissions and will not be able to take a resubmission after the deadline.

You are highly encouraged to ask your question on **Piazza** under the "a4" folder. Please **DO NOT** include your solutions in the comments you share on Piazza. Feel free to help other students with general questions.