

Back2Back testing project

Juliette CHOUDJAYE
François MATVIENKO
Armel MOTH
Lydia VIJAYARAJAH

28 mars 2018

1 Introduction

The purpose of our project was to allow a user to compare the performances (thanks to the accuracy) of three machine learning libraries (Weka, Renjin, SparkML).

This project was divided into two parts : the implementation of the application and later on, the extension to a dynamic web project using HTML pages. We will describe the thoughts and process that went into developing such an application, from modelisation to execution.

2 Implementation of the project

2.1 Classes

We started by implementing the classes corresponding to each libraries but soon realized that it would be difficult to have them inherit the same abstract class. We decided to look through the different design patterns to find one that would work for us.

We decided to use the "Adapter" design pattern. It allowed us to freely implement the classes of each library. We were therefore able to put each library class in the format of the abstract class Library using adaptor classes.

The list of the classes we implemented can be found in our previous report, as this is not the main subject here, we will not discuss it in details.

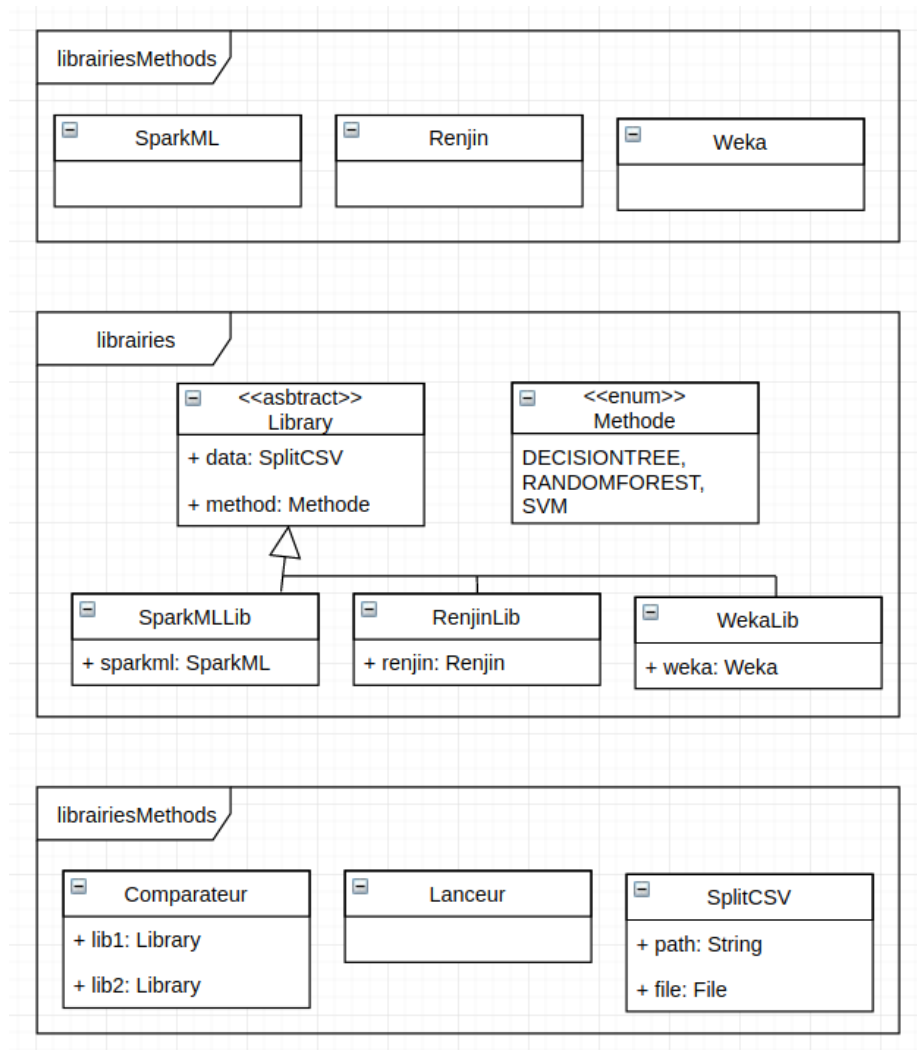


FIGURE 1 – Classes diagram

2.2 Librairies

2.2.1 SparkML

At the moment, our implementation that uses SparkML can calculate accuracy ratios for the methods Random Forest, Decision Tree or SVM.

As a result of the way our algorithm works, if the user ask for an unknown parameters it will not affect the program but if the user ask to change an existing parameter to a problematic value the program will fail.

At the end, the class SparkMLLib simply calls the function of the object AlgoSparkML which is link to the statistical method ask by the user with the parameters define as before.

2.2.2 Weka

With Weka, we implemented the method to obtain J48 Tree, which is a decision tree specific to Weka, and Random Forest, this library is very interesting because it arranges its own methods to read the CSV, what makes the import of the data very effective and fast. Indeed, thanks to the CSV Loader, it is rather simple to give the test and learning samples.

2.2.3 Renjin

With Renjin, we defined our Java arguments (or R variables) on one side, then our functions making decision tree and random forest on the other side. In a call to our R script, when we initialized our arguments in Java, there were converted and recorded as R variables. Then these variables were used as arguments in the call of all the functions restrained in the script. The result of each of these functions (the accuracy) was then converted and collected by Java.

3 Extension to a Dynamic Web Project

3.1 What we planned to do

When we started designing our web application, we decided to include as much options as possible knowing that we would probably have to do a selection according to the time we had left.

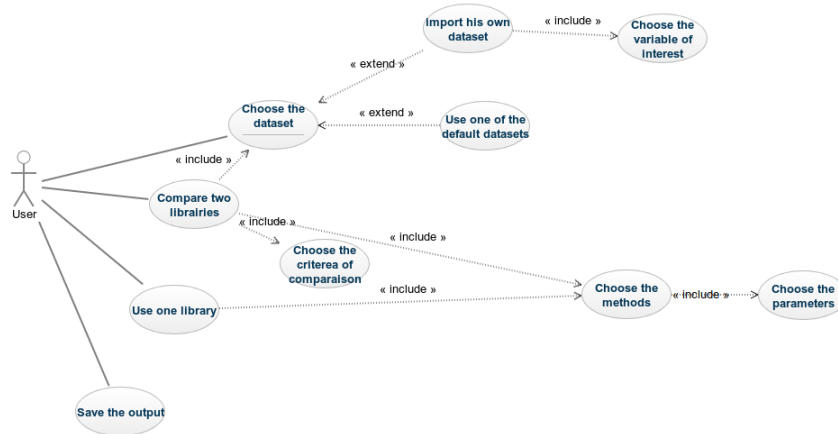


FIGURE 2 – Use-case diagram

It seemed interesting to us to allow the user to use our application not only to compare two libraries, but also to obtain results with a single library.

The user could choose his own dataset and set his variable of interest.

At the end it would have been possible to save the output in a file, in order to read it later.

3.2 Web page

The application that we have been able to produce have less options that what we planned at the beginning. You can see below the use-case diagram of the effective application.

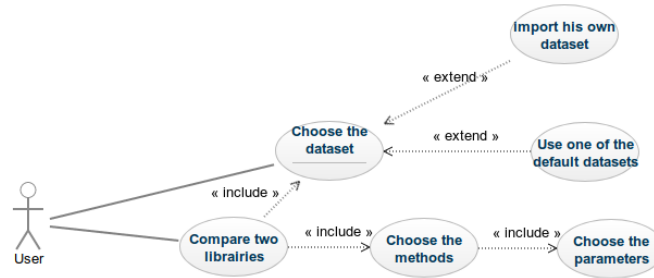


FIGURE 3 – Use-case diagram

Our design is very simple, because we wanted it to be efficient and easy to use more than anything. In the page, you can choose two libraries, then choose the methods and start the analysis. There is an option to import your own datasets if needed, otherwise you can use iris.csv or statsFSEVary.csv. After that, the results appear on the page : the accuracy for the chosen methods and other relevant information.

Would you like to import your own data set? : ☐ Yes ☒ No

What is the first library that you want to use?

– Select a library – ▾

What is the second library that you want to use?

– Select a library – ▾

What is the method that you want to use for the first library?

– Select a method – ▾

What is the method that you want to use for the first library?

– Select a method – ▾

Start Analysis

FIGURE 4 – Web page

We used a HTML page with CSS style to do the page of our application.

3.3 Java Servlet

The main part of the web implementation was linking the HTML data that we obtain through a form and the Java application. We used a Tomcat server and a Java Servlet to do the link between the two.//

As you can see below, the user goes on the web page (HTML form) and submits the form with their answers. The Tomcat server then does the link with the Java Servlet.

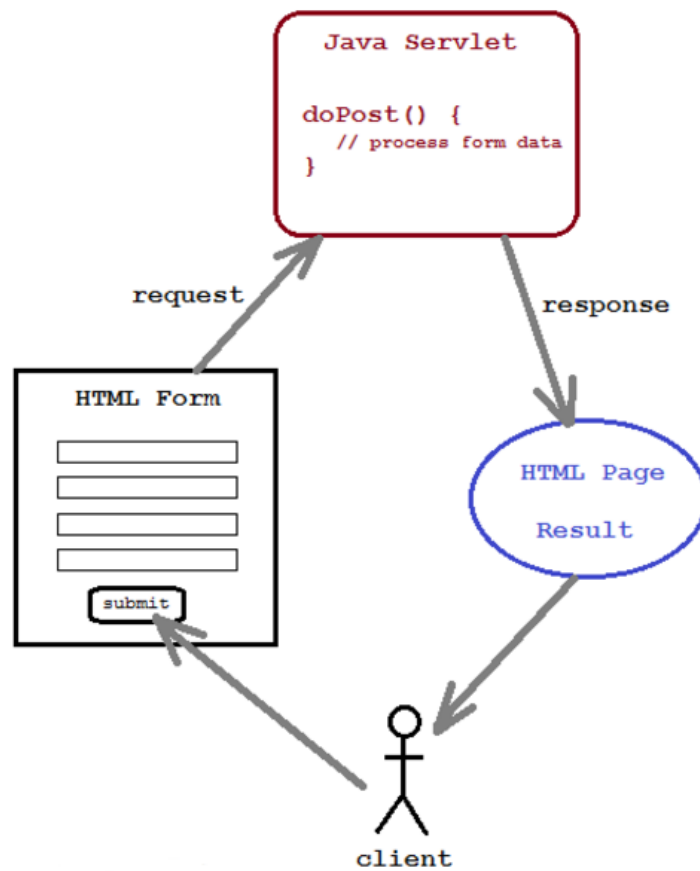


FIGURE 5 – Modèle de la Servlet

Our Java Servlet is a class extending the `HttpServlet` class. It has several Servlet-specific methods (`doGet()`, `doPost()`), as well as one method for each library.

There are two types of requests that can be sent to a Servlet : GET requests and POST requests. The former requests data from a specified resource and the latter submits data to be processed to a specified resource.

In our case :

- `doGet()` : this method allows a servlet to handle a GET request. This method redirects the user to the right HTML form.

```

public void doGet(
    HttpServletRequest request,
    HttpServletResponse response )
    throws ServletException, IOException {
    /* Displays the form */
    this.getServletContext().
        getRequestDispatcher( VUE ).
        forward( request, response );
}
  
```

- `doPost()` : this method allows a servlet to handle a POST request. Our HTML form's type is POST. Therefore, we can collect the data submitted through the form using the method `request.getParameter("ParameterName")`. We create an attribute for each HTML parameter. We can now use two "switch case" on the values of both libraries to calculate the accuracy while taking into account the methods and their arguments. One we get the accuracy of both methods we use the `setAttribute("name",value)` method to be able to display the results on a new HTML page using a JSP file.