CompareML

A comparator for machine learning algorithms libraries and services

User Manual

for CompareML 1.0

A. J. Fernández-García,
J. C. Preciado,
J. M. Conejero,
R. Rodríguez-Echeverria,
F. Sánchez.

This user guide is for *CompareML* version 1.0

Copyright (c) 2019 – 2020

Antonio Jesús Fernández-García, Juan Carlos Preciado, Jose María Conejero, Roberto Rodríguez-Echeverria, Fernando Sánchez.

Quercus Research Group, University of Extremadura
Applied Computing Group, University of Almeria

Welcome to the *CompareML* User Manual

Table of Contents (no actualizado)

Notas:

- **Comentar algoritmos disponibles**
- **Comentar proveedores disponibles**

1. Introduction

CompareML is a comparator for machine learning algorithms libraries and services. It makes it easy for users to create a test model of their dataset in three of the most widespread options such as Scikit-Learn, Turi Graphlab and R libraries and, at the same time, allows selecting different well-known classification and regression algorithms available in all providers.

The characteristics of *CompareML* facilitates the task of choosing the most suitable provider for their data, improving notably the experiment results while reducing time and costs. Furthermore, *CompareML* helps in selecting the algorithms which are liable to produce the best results for their datasets.

2. Installation

CompareML is a web application, hence it is not necessary to install additional software or driver in order to be used, just a web browser is enough. We suggest using the newest version of the Chrome browser for a better experience.

CompareML does not require users to be registered so, anonymous users can directly access the web application functionalities without a previous sign in.

The fact that *CompareML* users do not need to sign up for the website and it is accessible through conventional web browsers make it easy to researchers, data scientist and users, in general, to make use of this useful tool for their own purposes without losing time in unnecessary tasks.

3. User Interface

The user interface of *CompareML* has been designed to maximize usability being simple, consistent and offering cross-browser compatibility. It has an interface to set up the experiments and an interface to show the results of the experiments once they have been completed. The user interfaces are described in the next subsections.

3.1. 'Experiments set up' interface

In this interface, it is possible to set up experiments that want to be carried out. Through this interface, it is possible to upload data, choose the machine learning services and libraries that need to be covered in the experiment, and select the algorithms on which the test models will be built. Figure 1 illustrates this interface:
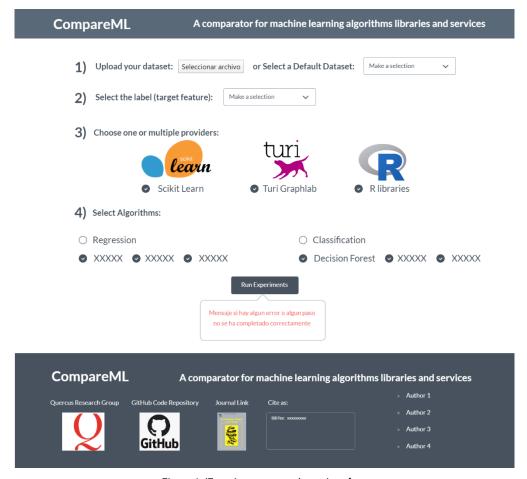

Figure 1: 'Experiments set up' user interface

The functionalities and tasks that can be performed in this interface are described in Section X: Functionalities and Section X: Illustrative Example.

3.2. 'Experiments Results' interface

In this interface, it can be analyzed the results of the experiments previously set up and executed. Through this interface is displayed the information and metrics obtained from the experiments which are the key to deduce which provider and algorithm suits better the provided data. Figure 2 illustrates this interface:

```
                Turi Graphlab                                    Scikit-learn                                          R
{                                                    {                                                    {
  "Random Forest": {                                   "Random Forest": {                                   "Random Forest": {
    "accuracy": 0.9,                                     "Iris-setosa": {                                      "result": "Confusion Matrix and Statistics
    "auc": 0.9838958994969226,                             "precision": 1,
    "confusion_matrix": "                                  "recall": 1,                                                       Reference
                                                           "f1-score": 1,                                    Prediction       Iris-setosa Iris-versicolor Iris-virginica
    +----------------+----------------+-------+            "support": 10                                        Iris-setosa         10              0              0
    | target_label   | predicted_label | count |        },                                                      Iris-versicolor      0             11              1
    +----------------+----------------+-------+          "Iris-versicolor": {                                    Iris-virginica       0              2              6
    | Iris-virginica | Iris-versicolor  |   1   |          "precision": 0.9166666666666666,
    | Iris-virginica | Iris-virginica   |   6   |          "recall": 0.8461538461538461,                      Overall Statistics
    | Iris-versicolor| Iris-virginica   |   1   |          "f1-score": 0.8799999999999999,
    | Iris-versicolor|   Iris-setosa    |   1   |          "support": 13                                                        Accuracy : 0.9
    |   Iris-setosa  |   Iris-setosa    |  10   |        },                                                                 95% CI : (0.7347, 0.9789)
    | Iris-versicolor| Iris-versicolor  |  11   |        "Iris-virginica": {                                      No Information Rate : 0.4333
    +----------------+----------------+-------+            "precision": 0.75,                                   P-Value [Acc > NIR] : 1.255e-07
    [6 rows x 3 columns]                                   "recall": 0.8571428571428571,
    ",                                                     "f1-score": 0.7999999999999999,                                        Kappa : 0.8469
    "f1_score": 0.8965079365079364,                       "support": 7
    "log_loss": 0.3947187115725084,                     },                                                   Mcnemar-s Test P-Value : NA
    "precision": 0.8943001443001443,                    "accuracy": 0.9,
    "recall": 0.9010989010989011
```

Figure 2: 'Experiments Results' user interface

The functionalities and tasks that can be performed in this interface are described in Section X: Functionalities and Section X: Illustrative Example.

3. Functionalities

The main functionalities of *CompareML* are:

4.1. Dataset Selection

This functionality allows users to select the dataset on which they want to perform the experiments. To provide the dataset they can either upload a file directly from their computers or choose between some defaults dataset that \textit{CompareML} put at their disposal. The default dataset provided are are the \textcolor{red}{xxx, xxx, and xxx datasets}. The customized dataset uploaded by users should be given in a CSV file format.

4.2. Label Selection

This functionality allows us to define the label feature, *i.e.*, the column of the dataset which will be predicted by the models built by the machine learning algorithms. Only one label can be selected, and users can select them through a drop-down menu after selecting the dataset.

4.3. Providers Selection

Through the provider selection functionality, users can choose between the variety of providers available in *CompareML*. As commented before, in this version 1.0, the libraries and tools are Turi Graphlab Create, Scikit-Learn and R.

4.4. Algorithms Selection

Through the algorithms selection functionality, users can choose between the variety of algorithms available in \textit{CompareML}. As commented in Subsection \ref{algorithms}, in this version the \textit{Random Forest}, \textit{xxx} and \textit{xxx} \texttt{Classification} algorithms and the \textit{xxxx}, \textit{xxx} and \textit{xxx} \texttt{Regression} algorithms are available. In each experiment only one type of algorithms (\texttt{Classification} or \texttt{Regression}) is allowed, due to their different predicting nature (categories and numbers respectively).

4.5. Experiments Running

This functionality compiles all the data entered by the user, ensures that it is correct and processes it to perform the experiments that users need to carry out. If the data is wrong or incomplete, the user is notified.

4.6. Results Visualization

This functionality report with the results of the experiments carried out. Where \texttt{Classification} algorithms are concerned, metrics such as \textit{accuracy}, \textit{auc}, \textit{f1_score}, \textit{precision}, \textit{recall} or a \textit{confusion matrix}\cite{ConfusionMatrix} are shown. Where \texttt{Regression} algorithms are concerned, metrics such as \textit{xxx} or \textit{auc} are shown.

4. Illustrative Example

% As an illustrative example, we are going to use \textit{CompareML} to choose the provider that, a priory, is best suited for dealing with a dataset. The example dataset, called \texttt{Graduates.csv}, includes data obtained across several years of 6867 students of the University of Almería (Spain) during their first year in college labeled as graduated or not.

% This real-world dataset was generated by the authors of \cite{FernandezGarciaWorldCist2018} with the aim of creating a decision model that could be used to predict if a university student will graduate after finishing the first year of her college studies.

% Table \ref{table:SetOfFeatures} shows the features that make it up.

Feature Name	Description	Type
Age	Age	Numeric
BirthProvince	Place of Birth	String
Nationality1	First Nationality	String
DoubleNationality	Has the student more than a nationality?	Boolean
Degree	Degree Name	String
DegreeField	Degree Field	String
Faculty	Faculty Name	String
UniversityAccessType	How the student access to the degree	String
CreditsEnrolled	Number of Credits Enrolled	Numeric
AverageScore	Average Score in the Subjects Enrolled (0-10)	Numeric
SuccessRate	Percentage of Credits Pass	Numeric
Graduated	Has the student graduated? *(Label)*	Boolean

% The first step consist of providing the dataset to \textit{CompareML}. To do this, the option of selecting a default dataset is ignored, and the the \texttt{Graduates} dataset is directly upload from our computer.

% In the second step, we choose the label feature that we are interested to predict, which is the \texttt{Graduated} feature. This boolean feature has two classes \textit{(Y, N)}, that indicates whether a student has finished a concrete university degree.

% In the third step, the providers supported by \textit{CompareML} are enumerated. In this version, as previously commented, Scikit Learn, Turi Graphlab and R providers are supported. At least one of them has to be selected. In our example we are interested in comparing the 3 providers, so we select all of them.

% The fourth step consists of selecting the algorithms that will be used to build the models. There are two options available, \texttt{Regression} or \texttt{Classi\-fication}. Depending on the type of algorithm selected we can choose from a variety of algorithms. In our example, we are interested in predicting a categorical value and, for that reason, we mark the \texttt{Classification} algorithms checkbox. After selecting this option, we have to select at least one \texttt{Classification} algorithm. In our

example we want to perform an experiment including all possible algorithms, so we select all of them (xxx, xxx y xxx).

% If everything is correct and the parameters are set correctly, we can run the experiments by pressing the ``Run Experiments" button. If not, an error message will be shown describing how to solve the problem.

% When the experiments are performed, \textit{CompareML} will show the results obtained. In our example, we have the results shown in Figure \ref{fig:ExperimentResults}. The results returned for \texttt{Classification} algorithms include metrics such as XXX, XXXX and XXX.