Machine learning course project

Cécile Chavane, Jehan de Bryas, Antoine Goupil de Bouillé
Projects EPFL, Machine Learning Course, Fall 2022

***Abstract*—This report develops our preprocessing, modeling and classification methodology for the identification of Higgs Boson in CERN measurements.**

I. Introduction

This first project aims at classifying observations made in one of the sensors of the CERN Particle Accelerator in Geneva. These observations can be of two types: some correspond to Higgs bosons, an elementary physical particle, the others correspond to the background, to other particles of other nature.

The database corresponds to measurements of particle physics properties of Particle Accelerator. We can interpret these data thanks to a classification algorithm.

Our understanding of the transversal objectives of the project seems to be structured around several key points: understanding the functioning of the basic machine learning mechanisms: preprocessing of data, visualizations, variables, selection of the best drivers, elaboration of models, selection and evaluation of models and parameters, training on the whole data set and finally predictions on the test data.

Our report will try to clarify the functioning of our classification model, by evoking particularly the decisions of preprocessing and methods of selection of the models, allowing to reach the results that we had.

II. Vizualisation of the data

A. Scatters of all features

Some scatter plots of all the features, in order to have a first insight of the data. We can see that there is a lot of -999

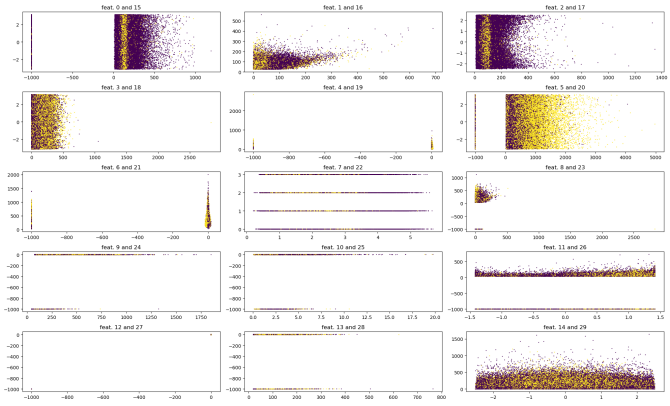

Fig. 1. Features and labels of classification

values. Let's remove them just to have a better look at the data (Fig. 2.)

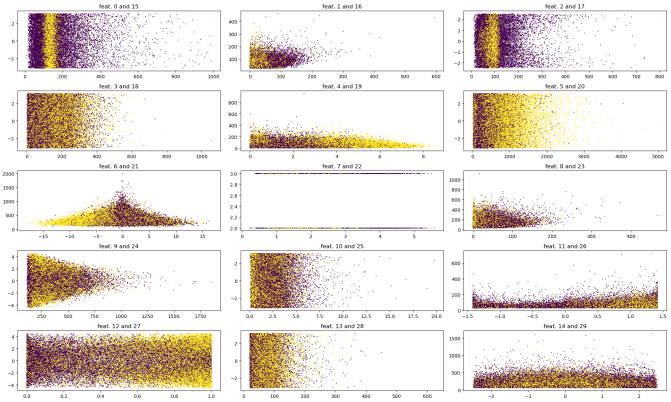

Fig. 2. Features and labels of classification

III. Preprocessing the dataset

Preprocessing of the dataset is important in order to improve our machine learning predictions. We have chosen to apply different transformations to the data set. It is mendatory to apply the same transformations to the whole data set (training and testing data).

A. Labelling the data

When the observation corresponds to a Bozon particle, we assign the label 1. When the observations corresponds to other particle of the "background", we assign the label -1. We have to change the labels to 0 and 1 when using logistic regression.

B. Deleting outliers

We have chosen at the beginning to replace the outlier values, -999 by the average of the column, to avoid that our models fit on the outliers. This methodology gives us good results in prediction.

C. Standardization

We have chosen to apply standardization to all of the drivers columns. This preproprocessing transformation allows all the drivers to be as significant in the models.

D. Polynomial transformation

The preprocessing pipeline includes a polynomial transformation of the drivers. This allows the linear models (SVM or logistic regression) to be able to capture non-linear relations between the drivers and the labels.

E. Principal Component Analysis

The PCA is used to reduce the number of drivers by keeping only the principal components directions. We have chosen to keep only the directions of data that explained 95% of the variance.

F. Separation of the data into subsets

At the beginning, we decided to replace all outliers (-999 values) by the mean value of all the non-outliers for each feature. Knowing that there is approximately 68'000 observations that doesn't have a single outlier, over 250'000 , we knew that this was bad for our predictions. It was creating a bias for the classifier of all the non-outliers, and classifying sort of randomly the outliers. After observing at the scatter-plots of all features (Fig. 1.), we decided to separate data into subsets for which there was not a single feature mixing -999 and non -999 values, that was for us the problem. After processing that, we had 6 independent subsets for which the union correspond to the all data set. Because feature 22 takes only values in $\{1, 2, 3, 4\}$, which is not adapted for linear classifiers, we decided to separate data again. And finally, we separate data for feature 0, one subset for values superior than 118, and the other subset for values inferior than 118. At the end, we have 2 x 4 x 6 = 48 subsets (some are empty) to treat, and so 48 classifiers to make, for each case. This method is used in "Boosted" classifiers.

IV. Models and method

A. Logistic regression with gradient descent

Logistic regression is the first model used for the classification of observations. In statistics, logistic regression is a binomial regression model. We have used gradient descent to find the best parameters to fit the training data.

B. Soft Vector Machine (SVM) with gradient descent

A support vector machine (SVM) is machine learning algorithm that analyzes data for classification and regression analysis. SVM is a supervised learning method that looks at data and sorts it into one of two categories. An SVM outputs a map of the sorted data with the margins between the two as far apart as possible. We have developed a gradient descent function to minimize the Hinge loss function. This algorithm allows to find the parameters allowing the separation of the space by a hyperplane.

C. "Boosted" SVM and Least squares

After realizing that we should divide the input data into subsets for which the behaviors were similar (see part. 3.F), we decided to associate each type of subsets with its own classifier, but all of the same model type. To make prediction, we then assign each points of the test set to its corresponding classifier, according to its characteristics. This technique gives us a significant accuracy improvement of approximately 6% compared to the use of one classifier for all train set.

This "Boosted" classifier was trained on SVM and Least squares models. it gives us both an accuracy of 81.5%.

D. Selection of hyper-parameters

We use Ridge regularization in the regression, allowing to reduce the extreme values of the regression parameters for the SVM and logistic regression. One difficulty was to find the best hyper-parameters in order to have the best trade-off between fitting the training set and having good predictions.

We have used a K-fold cross validation methodology to train and test our models and be able to find the best hyper-parameters. The K-fold cross validation process gives the average of all folds. We have tested different values of k, but 5 gives us good results to compare the models.

(see Table 1. for detailed results)

V. Results

Model	Ridge coefficient	Learning rate	Accuracy
Logistic regression	0.1	0.01	49.8%
Logistic regression	1.0	0.01	50.7%
Logistic regression	0.1	0.001	50.7%
Logistic regression	1.0	0.001	50.7%
SVM	0.1	0.01	74.44%
SVM	1.0	0.01	67.24%
SVM	0.1	0.001	67.32%
SVM	1.0	0.001	67.12%

TABLE I
PRESENTATION OF K-FOLD RESULTS (K = 5, N° ITERATIONS = 100)

The ridge coefficient allows to penalize high values of coefficients. We can see that it has not a low impact on the testing accuracy.

The learning rate has impact on the accuracy.

Model	Accuracy	F1 Score
SVM Boosted	81.5%	70.5%
Least squares Boosted	81.5%	70.2%
SVM	69.7%	54.9%
Logistic regression	55.4%	49.4%

TABLE II
PRESENTATION OF SUBMISSION RESULTS

We used the 3 models above, and observe that the Boosted SVM is the best in classification. We have used a small learning rate (even if it is time consuming), to assure better predictions. Our final parameters for Ridge and Learning rate were 0.1 and 0.001

VI. Summary

The methods and models used are the result of an iterative process. We started by using the functions of the practical work. We then focused our work on classification models (SVM and logistic regression) versus prediction models (Linear Model).