Machine Learning Course - Project 1

Quentin Laville, Valentin Moullet, Jol M. Fonseca
Department of Computer and Communication Sciences, EPFL, Switzerland

I. Introduction

The whole process described here concerns the first project of the CS-433 Machine Learning class. The goal is to apply various machine learning models on a dataset provided by the CERN, generated by the Large Hadron Collider, in order to distinguish experiences resulting in background from Higgs boson decay.

II. The Data

A. Provided

Two datasets are given: a train and test set. The train set contains more than $N_1 = 250k$ experiences with outcomes (either a 'b' for background, and 's' for signal − i.e., a Higgs boson decay). The test set contains more than $N_2 = 500k$ experiences without the outcome. Each sample was described by a row of $D = 30$ features.

B. Pre-processing

1) Standardization: We standardize both the training and the test sets using the training mean and variance.

2) Dataset split: After analyzing the data sets, one can notice that only one feature is categorical: PRI_jet_num takes value in $[0, 3]$. Moreover, some features seems to match a pattern following this jet number. Thus, data points having $jet = 0$ miss a lot of values, the same when $jet = 1$ but in a lesser proportion and when $jet = 2$ or $jet = 3$, the vales for the features are almost complete (as well as being quite correlated). After this exploration, we decided to train the data points that have different jet numbers separately, as they look like different in nature.

3) Missing values: As experiences could not always give relevant data, missing values (-999.0) were filled with a NaN values. As we have seen above, some features don't make sense anymore for some jet values, hence they are removed. For the others, the NaN values are replaced by the median of the corresponding column.

4) Feature engineering:

III. Machine Learning Models and Methods

A. Models

The following models were used throughout the project:

1) Gradient descent (γ, $iterations$)
2) Stochastic gradient descent (γ, $iterations$)
3) Least square with a polynomial basis ($degree$)
4) Ridge regression with a polynomial basis ($degree$, λ)
5) Logistic ($degree$, $iterations$, γ)
6) Regularized logistic regression ($degree$, $iteration$, γ, λ)

All the parameters inside the parentheses are exhaustively tested by grid search.

1) $iterations \in \{50, 100, ..., 1000\}$
2) $degree \in \{1, 2, ..., 15\}$
3) $\gamma \in \{0.05, 0.1, ..., 1.0\}$
4) $\lambda \in \{1e{-}4..., 1e{-}1\}$

Note that the last parameter is computed in logarithmic space.

B. Cross validation

To train and test the different models, the 10-fold cross validation is used. This technique consists in splitting the data set in 10 size-equivalent bins. Then each bin is selected as the test bin, one after the other, and the 9 remaining ones are used to train the model. When the model is ready, you can check its validity again the test bin. This is helpful to avoid the over-fitting of a particular model over the training set.

C. Accuracy estimation

To compute accuracy, we simply computed the ratio between the resulting y_{train} and the true y. By the 10-fold cross validation, the final accuracy was done by doing the mean of each y_{train} obtained for each iteration. On top of that, we also computed the variance in order to show that the accuracy is indeed stable or not among the different folds.

D. Best Model

After using all models created with all related parameters with grid search, ridge regression give the best results. It's quite surprising, as we expect the logistic models to win this round, the problem being a classification problem (tell if a sample results in a Higgs boson decay or not). Note that the balancing of the train dataset isn't helping the overall accuracy against the test set, which could mean that the proportion is the same in the latter.

IV. Results

A. Figures and Tables

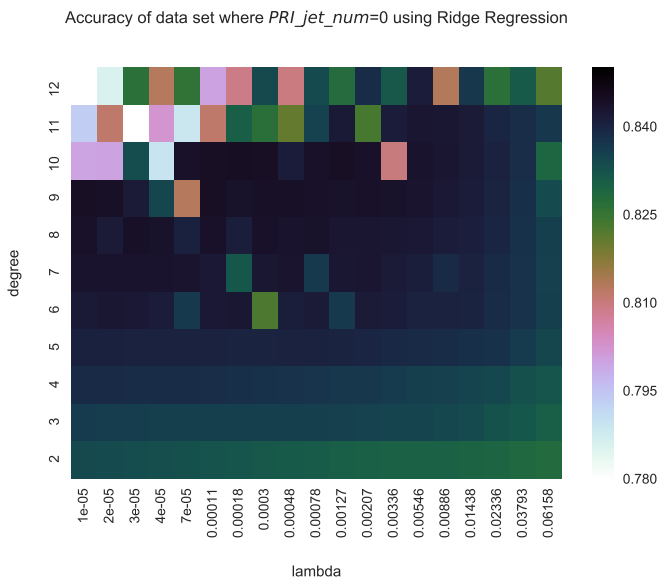

Fig. 1. Some test.

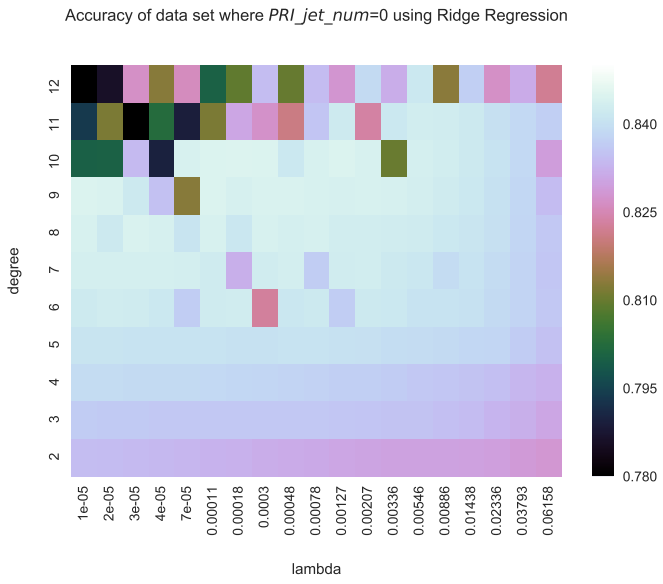

Fig. 2. Another test.