Machine Learning Project 1 - Higgs Boson

Sami Ben Hassen, Firas Kanoun, Ali Fessi
CS-433 Machine Learning, EPFL, Switzerland

***Abstract*—With faster and more powerful processors, Machine learning has become very important and offers useful tools and techniques to deal with a lot of problems in almost every scientific field. Regression is perhaps one of the most well-known and well understood algorithms in statistics and machine learning. In this paper we explore different types of regression and how they deal with a real life data-set from CERN in the field of physics.**

I. Introduction

The Higgs boson is an elementary particle in the Standard Model of particle physics, produced by the quantum excitation of the Higgs field, one of the fields in particle physics theory [1].

The aim of this project is to distinguish between Higgs Boson and background noise. We started with a data-set that had 30 features describing a proton collision event. Through data cleaning and feature engineering we sought to remove the noise and implement machine learning methods to solve this binary classification problem.

II. Models and Methods

A. Mandatory algorithms implementation

This project is part of a machine learning course, so before we dive in any predicting, we had the task to implement 6 machine learning algorithms presented in the lecture notes [2]. Depicted in Table I are the results we got after the implementation without any data cleaning nor feature engineering, the parameters chosen by hand without any optimization and the results obtained after doing a random 80-20 split on the data.

Methods	Parameters			Pred (%)
	λ	γ	max_iter.	
Least Squares	-	-	-	74.3
Gradient Descent	-	10^{-7}	2000	69.8
Stochastic Gradient Descent	-	10^{-7}	1000	69.3
Ridge Regression	10^{-5}	-	-	74.3
Logistic Regression	-	10^{-6}	3000	75.4
Reg Logistic Regression	10^{-2}	10^{-6}	3000	75.4

Table I
Initial test of the algorithms we were asked to implement.

The results we got are poor as expected but still better than a flip of a coin. We can see that all the methods are close in their accuracy. On the raw data-set the Logistic Regression methods work best, trailed by Least Squares and Ridge Regression while we found that The Gradient Descent methods were less accurate. This is pretty logical since Logistic Regression is made to deal with this kind of classification problem.

B. Analysis of the data-set

Once we took a closer look at the data-set, there were a couple of observations that grabbed our attention :

1) The values feature 22 takes are all in $\{0, 1, 2, 3\}$.
2) The existence of NaN values.
3) The correlation between certain features (Figure 1)
4) Features that take unique values.
5) Features that behave as a 'Gaussian' once we apply certain mathematical functions on them.

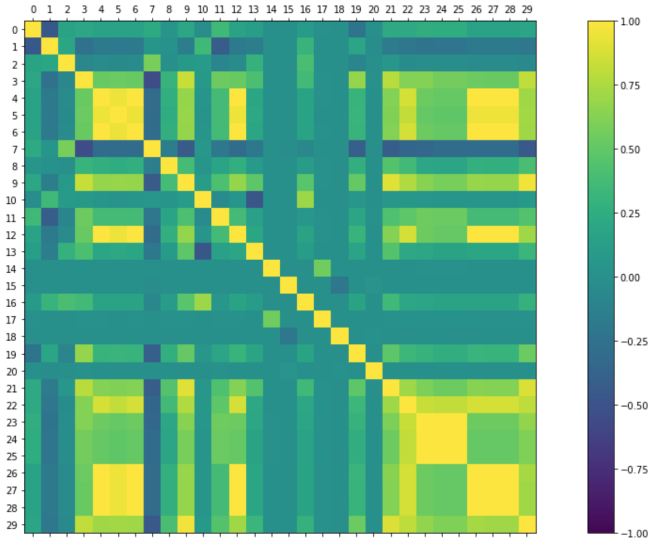

Figure 1. Correlation between different features (yellow means high correlation).

We decided to divide the data-set into 4 different subsets according to the values of column 22 that represent the jet number. Once we do that we notice that there were entire columns that take the value NaN or a unique value like 0 especially for jet 0 and jet 1. Getting rid of them was the only option. After this, we moved on to the feature engineering task where we had to apply some uni-variate transforms to better expose the linear relationship between the inputs and the output [3]. The transforms that worked the best for the different features were :

1) The Logarithm function.
2) The Square root.
3) The Classification (Brings the values that are ranging between to 2 distinct extremes to discrete numbers).

4) Division by the absolute maximum.
5) Replacement of the NaN values by the mean.
6) Standardization.

Fig 2 is a perfect example of how one can transform a certain feature.

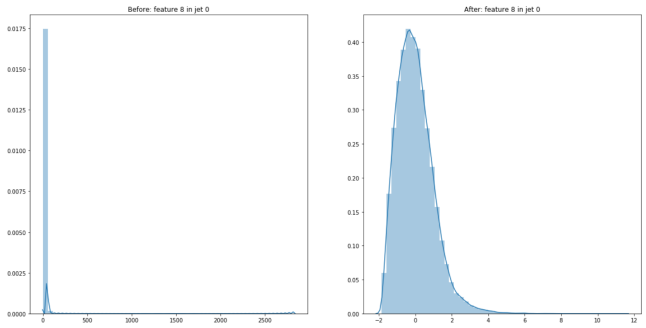

Figure 2. An example of a feature transformation.

C. Cross validation

Since we studied Logistic Regression very late, we tried to perfect the Ridge Regression method. To do this we used cross validation to find the best lambdas and degrees for each subset of the data-set. One example of a Cross Validation to find the lambdas is displayed in Figure 3.

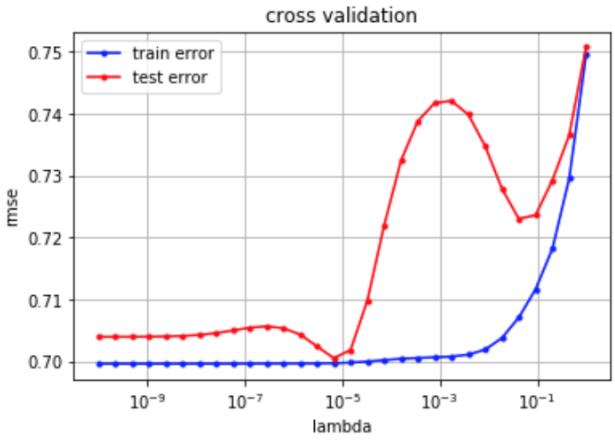

Figure 3. Cross Validation for jet0 with degree 3 trying to locate best lambda).

D. Adding features

The above transformations helped us to clean the data-set but were not enough to get the best predictions. And that is why we have decided to add more features of our own:

- Apply the sine function.
- Apply the cosine function.
- Apply the exponential function.

We believe that these operations gave us more than enough features for our algorithm to find a linear relationship between the input and the output.

E. Training

Cross validation outputs the degrees and the λs that guarantee the best loss in training without over-fitting. We compute the new X matrix using all the operations mentioned above along with the best degree. Then we apply the Ridge Regression algorithm with the best penalty λ to find the best values for W

F. Testing

Once we get the optimal values for W for each subset, We test our algorithm on all of them. the results are shown in Table II

	Subset by jet number				
	0	1	2	3	Overall
Accuracy (%)	84.48	80.42	83.00	83.07	82.75

Table II
RESULTS OF RIDGE REGRESSION.

III. Results and Further Discussion

The predictive ability of the algorithms we were asked to implement was in accordance with what we have studied during this first half of the semester. However, due to a lack of time we couldn't perfect the parameters and the features to fully benefit from what Logistic Regression has to offer as the optimal method for binary classification. Nevertheless, the results that we managed to get show that Ridge Regression is as good.

With our way of feature engineering, Ridge Regression yielded an accuracy of 83% on the Kaggle leaderboard dedicated to this class.

In hindsight, another way of cleaning the data could have been by splitting the data-set even more in order to get rid of the columns that had missing values. We could have even tried to predict them. However using the latter method could have been dangerous since we would be using values that are not 100% correct in our models.

IV. Conclusion

Through the course of this project, we implemented 6 methods of machine learning and we experienced hands-on the power and the limits of each one of them.

We learned how to explore the data, look for creative ways to clean it and get rid of all the missing values either by deleting the column if all it has are missing values or replacing them with the mean of the values of that feature.

References

[1] P. Onyisi, *Higgs boson FAQ*. University of Texas ATLAS group, 2012.

[2] "CS433 epfl, machine learning course," https://mlo.epfl.ch/page-146520-en-html/, accessed: 2018-10-21.

[3] "Machine Learning Mastery logistic regression for machine learning," https://machinelearningmastery.com/logistic-regression-for-machine-learning/, accessed: 2018-10-23.