Finding the Higgs Boson

Anna Vera Linnea Fristedt Andersson, Erik Agaton Sjöberg & Theodor Tveit Husefest
Machine Learning, CS-433, EPFL, Switzerland

***Abstract*—This report covers the implementation and prediction accuracy of six different machine learning algorithms, along with data analysis and feature engineering of a large dataset from CERN. The algorithms are trying to identify the Higgs Boson from background noise and the report shows that, given the used feature engineering method and algorithm implementations, the best suited algorithm for this is ridge regression.**

I. Introduction

The Higgs mechanism is said to give mass to elementary particles and consists of the Higgs boson and the Higgs field [2]. In this project our aim is to find the Higgs boson in a dataset from CERN, using exploratory data analysis and machine learning algorithms. The prediction accuracy of six different machine learning algorithms will be tested, after finding the optimal hyper parameters for the cleaned dataset.

II. Data Analysis and Feature Processing

A. Exploratory Data Analysis

By first exploring the dataset, it was discovered that a lot of values in the set contained -999. As these values differed greatly from all other values in the set, they were considered to be undefined values, which implies great need of data handling. These undefined values were temporarily set to $np.nan$ and the distribution of each feature was plotted. From this, the following observations were made:

1) The only categorical feature is PRI_jet_num.
2) Some features look log-normally distributed, which can cause issues when using least-squares method.
3) Some features look uniformly distributed in the interval $[-\pi, \pi]$. It is therefore natural to assume these are angles.

To examine the relationships of features, the covariance matrix including the predictions y was plotted. This gave an indicator of the importance of each feature, which was used in the feature engineering.

B. Feature Engineering

Based on the observations obtained from the analysis of the data, the following modifications of the data were made in order to improve the model:

1) All undefined values were set to the median value of the feature to avoid influence of outliers.
2) If the column only had non-negative values, the feature was assumed to be log-normally distributed. Therefore, the natural logarithm was applied on all those columns.
3) On all the columns that were assumed to be angels, a cosine-basis was applied.
4) After augmenting the features, the distributions were once again studied. The features that had low covariance with the prediction, and that had poor distribution, were then removed.
5) We tried splitting the data into the different jets, but as this did not give better results, we did not use this going forward.

III. Method and Models

The six different machine learning algorithms that were tested are: linear regression using gradient descent, linear regression using stochastic gradient descent, least squares regression using normal equations, ridge regression using normal equations and both regularized and simple logistic regression using gradient descent.

In order to obtain accurate algorithms, the hyper parameters for each algorithm were calculated using cross-validation. These hyper parameters are: γ, λ, $degree$.

A. Gradient Descent and Stochastic Gradient Descent

For gradient descent, the hyper parameter to be determined was γ. By cross-validating γ in the interval $[10^{-4}, 10^{-2}]$, it was possible to determine which γ that gave the lowest error, see Figure 1. In the cross-validation step, the number of iterations was set to 50. After finding γ that gave the lowest error, the gradient descent was once again tested with this value and an increased number of iterations set to 3000, in order to make sure that the chosen γ indeed gave the lowest error.

For stochastic gradient descent, the cross-validation of γ was done the same way as for gradient descent. However, the interval was $[10^{-5}, 10^{-2}]$ and the algorithm different, which gave another lowest error, and therefore another optimal $\gamma = 0.0003$.

B. Least Squares

For gradient descent, the hyper parameter to be determined was the polynomial degree. By cross-validating $degree$ in the interval $[1, 11]$, it was possible to determine the $degree$ where the error was its lowest, see Figure 2.

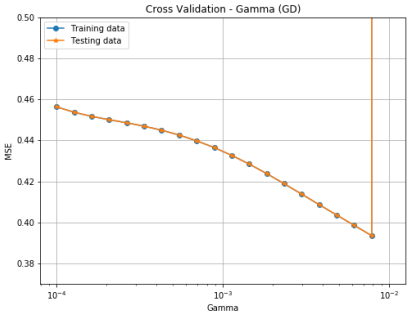

Figure 1. Cross-validation for γ, with gradient descent

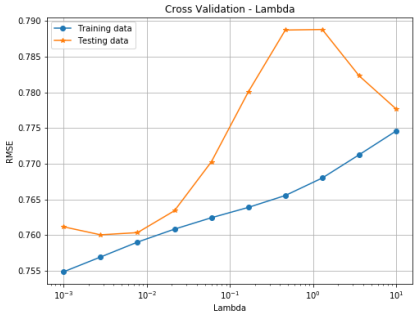

Figure 3. Cross-validation for λ, with ridge regression

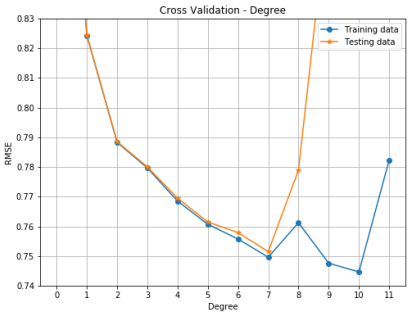

Figure 2. Cross-validation for $degree$, with least squares

C. Ridge Regression

Ridge regression has two hyper parameters, which are $degree$ and λ. Cross-validation was done on λ using different degrees. To begin with, the $degree$ that gave the best result in the least squares algorithm was tried. However, this didn't give a good result for ridge regression. Higher and lower degrees were then tried, along with the cross-validation, in order to obtain a plot that showed a good looking error curve. The degree that gave the best plot was 10 and this plot is shown in Figure 3. The resulting λ was then chosen according to the global minimum of the test error curve.

D. Logistic Regression

The regularized and simple logistic regression algorithms were both optimized by trying different hyper parameters and choosing the values that resulted in the best prediction. Since the logistic algorithms want the prediction to be either 1 or 0, all of the y values that were equal to -1 were changed to 0. The prediction was evaluated by using the $sigmoid$-function with the model prediction as input. Following this, all values that were bigger than 0.5 were set to 1 and all values underneath 0.5 were set to 0. By then, it was possible to calculate the accuracy of the model. The regularization term for the regularized logistic regression was set to be the sum of the absolute of the weight-values, also known as the Manhattan distance.

IV. Results

The hyper parameters and number of iterations that were used for the different algorithms can be seen in Table I, along with the different models' prediction accuracy. From this table it is visible that the ridge regression algorithm gave the best prediction result.

Table I
HYPER PARAMETERS AND NUMBER OF ITERATIONS USED

Algorithm	γ	λ	Max. iter.	Deg.	Pred.
Gradient descent	0.008	-	3000	1	73.3%
Stoch. grad. descent	0.0003	-	3000	1	69.6%
Least squares	-	-	-	7	81.3%
Ridge regression	-	0.0027	-	10	80.7%
Logistic regression	10^{-6}	-	2000	-	73.5%
Reg. log. regression	10^{-6}	1	2000	-	69.5%

V. Summary

In this project various ways of creating a machine learning model were studied. Six different algorithms were used to try and solve a binary classification problem. In the end, regardless of the algorithm used, it became clear that cleaning and handling the data is one of the most important aspects of a machine learning project. One can tackle the data issue in different ways and the way it was done in this project is an example of how it could be done. The NaN values were set to the median of the specific feature. There are certainly many other ways which would result in a better end result, for example one could study the distribution of the feature and change the NaN values according to the given distribution. Furthermore this project gave the creators insight on the issues and problems with data handling, that one has to face when dealing with machine learning.

References

[1] Urbanke R., Jaggi M. and Khan M. (2019). Machine Learning Course - CS-433.

[2] ATLAS. (2018). The Higgs boson: the hunt, the discovery, the study and some future perspectives. [online] Available at: https://atlas.cern/updates/atlas-feature/higgs-boson [Accessed 23 Oct. 2019].