Machine Learning: Looking for the Higgs Boson

Y. Saied, L. Syvis, V. Crettenand
School of Computer and Communication Sciences, EPFL, Switzerland

***Abstract*—The standard model was one of the most ambitious scientific achievements in the 20th century culminating in the identification and classification of all the basic building blocks of our physical reality. In 2013 the last elementary particle of the standard model was discovered the Higgs boson by analysing the decay signatures of collisions at CERN. This paper aims to mimic that discovery with simple machine learning models.**

I. Introduction

The Higgs boson was the last elementary particle in the standard model to be experimentally discovered. Its discovery was announced by CERN in 2013 and won Higgs, the Belgian physicist who predicted its existence 50 years earlier, the Nobel prize in physics. As the Higgs boson decays rapidly into other particles, normally it's not directly observed, in reality it's its "decay signature", or the products that result from its decay process that are measured. This is rather challenging as numerous other decay signatures look similar. The aim of this paper is to make use of simple binary classification tools to analyse data from the Large Hadron Collider at CERN to recreate the process of "discovering" the Higgs particle. This paper arrives at a method that correctly identifies the Higgs boson with 81% accuracy.

II. Models and Methods

Six methods are used to distinguish datapoints from collisions that had the Higgs boson from others that did not.

A. Feature analysis

Number of features: The data had 30 features (columns) and our testing data had 250000 data points (rows).

B. Data cleaning and pre-processing

Data cleaning: As multiple columns in the data-set contained the value -999, which could potentially make our models fail, it was elected to replace them by the neutral value 0.

Data standardisation: The data was standardised by subtracting the columns' means from the respective columns and dividing by their respective standard deviation.

Feature augmentation: The data sets were augmented by appending them with increasing powers of the columns. Various degrees were tested for every method.

youssef.saied@epfl.ch

C. Classification techniques

This paper made use of 6 standard binary classification tools, all implemented using the Python numpy library; namely:

1) **Linear regression using gradient descent(LR_GD):** This was one of simplest methods used.
2) **Linear regression using stochastic gradient descent(LR_SGD):** This method is similar to the **LR_GD** with the difference being that instead of calculating the gradient using the entire data it instead uses an estimate calculated from a randomly selected subset of the data. This reduces the computational burden, achieving faster iterations in trade for a slightly lower convergence rate.
3) **Least squares regression using normal equations(LS):** This is a non-iterative fitting method that calculates the hyperplan that has the minimal average distance with all the data points.
4) **Ridge regression using normal equations(RR):** Similar to **LS** but with penalisation for large weights.
5) **Logistic regression using gradient descent or SGD(Log_GD/ Log_SGD):** The only method that doesn't return a hyperplan instead returning a logistic function.
6) **Regularized logistic regression using gradient descent or SGD(RegLog_GD/ RegLog_SGD):** Similar to **Log_GD** but with penalisation for large weights.

As this a binary classification problem the Logistic regression methods are normally better suited to address it. Because such logistic methods are less susceptible to outlier/ 'extreme' values. Furthermore the error or loss function associated with logistic regression is a more reliable measure of the fitness of the predictions.

III. Results and Discussion

In this section we compare and discuss the accuracy obtained using the six different models mentioned earlier. The errors and accuracy obtained through the six models as well as the parameters used are provided in Table I.

A. Results

For all the following results the hyperparameters were obtained by doing a grid search over a logspace from 10^{-9} to 1. The best results were obtained through the least squares method with augmented features which had an accuracy over the test data of 81%

Method	LR_GD	LR_SGD	LS	RR	Log_GD	RegLog_GD
Accuracy	0.745	0.738	0.746	0.738	0.712	0.712
Parameters	$\gamma = 0.1$	$\gamma = 0.01$, batch_size =10	$\gamma = 0.1$	$\gamma = 0.1$, $\lambda = 0.1$	$\gamma = 0.1$	$\gamma = 0.1$, $\lambda = 10^{-7}$

Table I
THE ACCURACY OBTAINED USING THE 6 DIFFERENT METHODS

1) No feature augmentation: The optimal parameters were similar throughout so the same parameters were used for all the models, namely $\lambda = 0.1$ and $\gamma = 0.1$(except **SGD** with $\gamma = 0.01$).
Linear regression through gradient descent provided an error similar to ridge regression and least squares. What is surprising is that even though Logistic regression using gradient descent methods were supposed to be the most optimised for the binary this is probably explained by the fact that;

2) Feature augmentation and best model: To increase the accuracy it was decided to use feature augmentation as the linear models were potentially under-fitting, there variance was low and they were noticeably biased. With all the models the results increased with feature augmentation, confirming the under-fitting hypothesis, but the model that had the best accuracy remained **LS** with features augmented to a polynomial of degree 9. This degree was arrived at by doing a grid search over degrees from 1 to 20. To verify that this model was indeed the best model. A 8-fold cross validation giving a similar accuracy which confirmed the un-biased-ness of the result.

B. Discussion

Even though the **LS** method with feature augmentation presented in the paper gave the best accuracy further statistical investigation is required to improve on the 81% accuracy figure as for particle detection applications this accuracy level seems unsatisfactory. The logistic regression methods were also inadequate due to the low variance of the features. Other more advanced tools such as neural networks could potentially improve on this accuracy level.

IV. SUMMARY

The aim of this paper, to preform binary classification on datasets from CERN to mimic the Higgs boson discovery, was achieved by implementing six different basic statistical methods. While all six methods were able to provide prediction of 70+% level of accuracy, the best results were obtained through the use of the simplest of them, the Least square technique (**LS**). The **LS**, when implemented with feature augmentation and the optimal parameters, gave an accuracy of 81 %.