Spot the Boson

Mladen Korunoski, Ljupche Milosheski, Manjot Singh
e-mail: mladen.korunoski@epfl.ch, ljupche.milosheski@epfl.ch, manjot.singh@epfl.ch

***Abstract*—The Higgs boson is an elementary particle in the Standard Model of physics which explains why other particles have mass. Particle collisions at high speeds can produce the Higgs boson but, due to its rapid decay, it is difficult to observe its existence directly. That is why we have to estimate whether a given event signature was a result of a Higgs boson. The purpose of this paper is to showcase how machine learning can help with this task by using effective classification methods for a vector of features that represent the decay signature of a collision effect.**

***Index Terms*—Machine Learning, Linear Regression, Logistic Regression, Classification, Higgs Boson.**

I. Introduction

The ATLAS and CMS experiments recently claimed the discovery of the Higgs boson. This particle is produced by collision between protons at high speeds. The observance of this particle is through its decay signature i.e. the given event signature is either a result of a Higgs boson signal or other processes (background). The ATLAS experiment at CERN provided simulated data used by physicists to optimize the analysis of the Higgs boson [1].

The Higgs boson has many different processes through which it can decay. When it decays, it produces other particles. In physics, a decay into specific particles is called a *channel*. The Higgs boson has been seen first in three distinct decay channels which are all boson pairs. The subject of this study is to try and improve on the analysis using machine learning.

The rest of the paper is organized as follows. Section II explains the data and preprocessing steps, and also discusses the models used and the model selection process. Section III presents the classification accuracy of the best models. Section IV contains discussion of the results. The paper is concluded in Section V.

II. Methods and Models

A. Data

The data was obtained from EPFL Machine Learning Higgs 2019 competition at AIcrowd[1]. The train and test datasets consist of $250,000$ and $568,238$ entries respectively. They have a unique integer identifier column, 17 primitive columns prefixed with *PRI* which are raw quantities about the bunch collision as measured by the detector, 13 derived columns prefixed with *DER* which are quantities computed from the primitive features, and a binary target variable *Prediction* with two possible values: s indicating the event is a result of a Higgs Boson and b otherwise. The only other column with discrete values is *PRI_jet_num* which has integer values between 0 and 3 inclusive. The rest of the columns are ordinary variables of real numbers. There are missing values for some columns indicated with -999. These values are either meaningless or could not have been computed.

[1]www.aicrowd.com/challenges/epfl-machine-learning-higgs-2019/

B. Data Preprocessing

One observation from analysing the data was that certain columns have missing values depending on the value of *PRI_jet_num* column. The data was split by the value of *PRI_jet_num* into 4 subsets. Columns that have missing values in all entries were dropped from the subset. The only column with partially missing values in every subset was *DER_mass_MMC* and, with respect to that, each subset was further split into 2 subsets, resulting in 8 subsets in total.

It was observed that some of the features are highly linearly correlated. This makes sense since the derived features are calculated from the primitive ones. It is essentially a feature augmentation, which was already given. If the calculation is linear, then their correlation would also be linear. The features were pruned so that the absolute value of the correlation between the remaining features is less than 0.88. Correlation of 0.9 was initially chosen, but for more convenience, features with a correlation of slightly less than 0.9 were dropped. This is important because of the feature selection process using L_1 regularization. If the features were not removed, then their coefficients in the trained models with L_1 regularization would have been about twice smaller, thus giving them a disadvantage in the feature selection process.

C. Feature Standardization

The distribution of the data was visualized using histograms. Most of the features in each subset follow Gaussian or skewed Gaussian distribution. Before passing the data to the model, it was standardized using

$$x_{new} = \frac{x_{old} - \mu}{s}, \tag{1}$$

where μ is the mean and s is the standard deviation of the feature column x. Each subset was standardized separately, with the mean and standard deviation of its corresponding training set. The testing subsets were not standardized separately with their parameters because it would result in a different transformation from the training subsets. This will help the models converge faster and also lead to stable coefficients convergence.

Another technique that was considered was min-max normalization however, the results were more accurate using the standardization technique.

D. Feature Engineering

The subsets were augmented using polynomial expansion with a degree 2 or 3 for every feature. The degree was chosen such that, on average, it had the best classification accuracy calculated with 5-fold cross-validation. Usually, the subsets with a small number of entries were augmented with a degree of 2. In some cases, the degree 3 was already overfitting the data due to the higher number of features. Because of that, low degree was preferred.

The complexity of the model was reduced using feature selection with L_1 regularization. It produces sparse weights and can be robust to irrelevant features [2]. For every linear or logistic regression model, a similar model with L_1 regularization was trained. Then, the percentage of features with the highest weight values were chosen.

E. Models and Model Selection

Linear and logistic regression models with L_2 regularization were tested and gradient descent was used to learn the parameters. The models were completely characterized by only two hyperparameters: the step size and the regularization parameter. However, two more hyperparameters were used for the input data for these models: the degree for polynomial expansion and the percentage of features after the feature selection. Furthermore, 4 or 8 models were trained depending on the number of available subsets.

Each model was evaluated with 5-fold cross-validation. The model that gave the best average classification accuracy with cross-validation was chosen. The expected classification accuracy calculated by cross-validation across all subsets was almost equal to the accuracy obtained on AIcrowd.

TABLE I
CLASSIFICATION ACCURACY (IN %) FOR DIFFERENT MODELS OBTAINED USING 5-FOLD CROSS-VALIDATION AND USING AICROWD'S EVALUATION SYSTEM. THE MODELS WERE TRAINED ON 8 SUBSETS. VALUES WITH * REPRESENT THE ACCURACY OBTAINED FOR INVERTED PREDICTIONS.

Model	5-fold cross-validation	AIcrowd
Baseline	67.0	67.1
Linear	47.7 (52.3*)	48.6 (51.4*)
Linear + FS	47.4 (52.6*)	48.3 (51.7*)
Logistic	**80.3**	**80.3**
Logistic + FS	80.1	79.9

III. RESULTS

It was discovered that splitting the data into 8 subsets is always more accurate than splitting it into 4 subsets. Due to simplicity, only the classification accuracy of the models trained on 8 subsets is presented. The baseline is defined as the obtained accuracy if the majority class in every subset is predicted. The models with feature selection (FS) in their names were trained with L_1 regularization. The step size and regularization parameters of all models were chosen such that the average classification accuracy with 5-fold cross-validation is maximized. The results are presented in Table I. The step size and regularization parameters are not provided in this paper because providing 16 constants would make it unreadable. They can be found in the source code of the project.

IV. DISCUSSION

It was not expected linear regression models to perform much worse than the baseline – predicting the majority class for each subset. The reason probably is that they predict real values instead of probabilities and are sensitive to imbalanced data. In fact, both linear regression models had an accuracy of bellow 50%, which is unacceptable for binary classification tasks. The reason being that one can always invert the predictions and get higher accuracy. In our case, even if the predictions were inverted, the accuracy would still be much lower than the baseline's. Therefore, it does not make sense to use any of the linear regression models at all.

The logistic regression model did significantly better than the majority of the models. This is not surprising since it is more robust to imbalanced classes. The optimal degree for polynomial expansion was different across subsets. In general, smaller subsets use a degree 2, whereas larger subsets use degree 3. Using a higher degree for the smaller subsets leads to overfitting. Therefore, it is better to use a simpler model.

The feature selection with L_1 regularization had worse accuracy for both models than without it. The reason for trying feature selection was to reduce variance by making the models simpler. We expected that using polynomial expansion with a higher degree and then selecting only a few of the best features would improve the performance. However, it made the results worse. It could be that the models found the polynomial expansions of only a few features important and they overfitted the data as they were more complex.

V. CONCLUSION

In this paper, a thorough study for spotting the Higgs boson was conducted. Different machine learning approaches i.e. regression and classification with an extension of hyperparameter tuning and feature engineering were used. It was found that some feature engineering techniques, such as using correlation to remove linearly dependent features and polynomial expansion, are essential in order to achieve better results. The logistic regression without feature selection with L_1 regularization outperformed the other models with an accuracy of 80.3%. For future study, it will be useful to include some insights from physics experts so as to do some feature engineering which will lead to better predictions. Other machine learning approaches, for instance, neural networks or support vector machines, may produce even more accurate results.

REFERENCES

[1] CERN, "Higgs documentation," 2014, Accessed: 10.10.2019. [Online]. Available: https://higgsml.lal.in2p3.fr/

[2] A. Y. Ng, "Feature selection, L1 vs. L2 regularization, and rotational invariance," in *International Conference on Machine Learning*. ACM, 2004, p. 78.