Applying Machine Learning to Higgs Boson Prediction

Zhechen Su, Harshdeep Harshdeep & Futong Liu

***Abstract*—With the explosion of data in numerous scientific and experimental disciplines, rational data interpretation requires machine learning techniques to create a robust model to depict the data set. Higgs Boson prediction is such a typical problem of high data complexity and dimensions. This report proposes a methodology that achieves 83.28% prediction accuracy by regularized logistic regression with appropriate data pre-processing and feature engineering. This methodology can be further refined by feeding more data or by adding more non-linear cross-terms.**

I. Introduction

Higgs boson is an elementary particle that gives mass to other elementary particles. However, Higgs boson cannot be observed directly as it decays rapidly through many different processes and produces a decay signature. By learning from the actual dataset provided by CERN, this project aims to establish a machine learning model to distinguish a Higgs boson (signal) from other process or particle (background). Thereby six basic machine learning algorithms are implemented and compared, among which regularized logistic regression is selected to predict the Higgs boson detection.

II. Models and Methods

A. Mandatory Methods

In order to find the most suitable model to depict this prediction problem, six methods are implemented and tested as baselines. Table I shows the performance of the six methods with standardization only, of which the parameters λ and γ are optimized by grid search. The prediction accuracy is computed by taking the average accuracy of 10-fold cross validation.

Methods	γ	λ	iterations	Accuracy
Gradient Descent	0.05	-	1000	71.40%
Stochastic Gradient Descent	0.005	-	2000	64.50%
Least Squares	-	-	-	72.26%
Ridge Regression	-	0.001	-	71.80%
Logistic Regression	0.01	-	5000	72.18%
Reg Logistic Regression	0.01	5	3000	72.47%

TABLE I
SUMMARY OF SIX MANDATORY METHODS

It can be concluded that all methods except SGD achieve similar prediction accuracy, among which regularized logistic regression works best. Besides, regularized logistic regression fits the scenario most suitably, as the prediction of Higgs boson is essentially a binary classification problem. Its penalty term λ regulates the balance between bias and variance, and handles the situation when the data is linearly separable. Hence, regularized logistic regression is selected as the baseline candidate for further improvement and development.

B. Exploratory Data Analysis

As observed from the raw data set, there are numerous 999.0s separating among valid values of features. These -999.0s are outside the normal range of all variables (outliers) and hence are substituted with NaN.

In order to discover the dependencies between the features, the correlation between features is computed and compared. Table II shows the feature pairs that possess strong correlation (more than 0.999). Strong correlation indicates that the information stored in one feature is highly relevant and similar to the other, and hence deleting one feature in a feature pair with strong correlation scarcely erases any information of the data set. Therefore, the five features in the second column of Table II can be deleted.

Feature 1	Feature 2	Correlation
DER_deltaetaet_jet	DER_lep_eta_centrality	0.999998
DER_deltaetaet_jet	DER_prodeta_jet_jet	0.999981
DER_deltaetaet_jet	PRI_jet_subleading_eta	0.999995
DER_deltaetaet_jet	PRI_jet_subleading_phi	0.999996
DER_deltaetaet_jet	PRI_jet_subleading_pt	0.999346

TABLE II
FEATURE PAIRS WITH STRONG CORRELATION

Moreover, it is noticed that the 22nd feature PRI_jet_num contains in fact categorical rather than numerical information, as it can only be a value of four numbers: 0, 1, 2, 3. Additionally, Table III shows that the NaN values congregate after categorizing with respect to the value of PRI_jet_num, such that several features become pure NaN and hence should be removed. For category 0, since its last feature PRI_jet_all_pt also becomes all zero, this feature is also removed.

Feature	Cat0	Cat01	Cat2	Cat3
DER_mass_MMC	26	9	5	6
DER_deltaeta_jet_jet	100	100	-	-
DER_mass_jet_jet	100	100	-	-
DER_lep_eta_centrality	100	100	-	-
PRI_jet_leading_pt	100	-	-	-
PRI_jet_leading_eta	100	-	-	-
PRI_jet_leading_phi	100	-	-	-
PRI_jet_subleading_pt	100	100	-	-
PRI_jet_subleading_eta	100	100	-	-
PRI_jet_subleading_phi	100	100	-	-

TABLE III
PERCENTAGE OF NAN IN EACH FEATURE AFTER CATEGORIZING

Regarding the first feature DER_mass_MMC, where the NaN values do not exhibit orderly distribution, it is feasible to substitute the NaN values with the mean of valid values, which consequently causes little influence to its overall statistic property. So far, there are 12 features removed from category0, 8 from category1, 1 from category2 and category3 respectively.

C. Augmented Feature Vectors with Polynomials

Since a linear model per se is not very rich, the input features are augmented by adding a polynomial basis of degree M to enhance its representational power. Table IV shows the prediction accuracy of regularized logistic regression with categorization, of which the average accuracy is 81.53%, while by contrast its prediction accuracy without categorization is 78.84%. Apparently categorization refines the model and improves the prediction result.

Category	Total features	γ	λ	M	Iterations	Accuracy
Cat0	18	0.001	5	2	5000	82.91%
Cat1	22	0.001	5	2	10000	78.59%
Cat2	25	0.0001	5	2	25000	79.70%
Cat3	25	0.001	5	2	6000	81.44%

TABLE IV
FINAL MODEL WITH FEATURE ENGINEERING

D. Feature Engineering

Additionally, it is highly possible that there are non-linear relationships between features, and hence for each feature pair three operations are undertaken: multiplication, square root and logarithmic. Table V shows the final optimized parameters of the complete model after feature engineering.

Category	Total features	γ	λ	M	iters	accuracy
Cat0	97	0.001	5	2	8000	84.41%
Cat1	903	0.001	5	2	8000	81.48%
Cat2	1281	0.0001	5	2	45000	84.77%
Cat3	1281	0.001	5	2	10000	84.96%

TABLE V
FINAL MODEL WITH FEATURE ENGINEERING

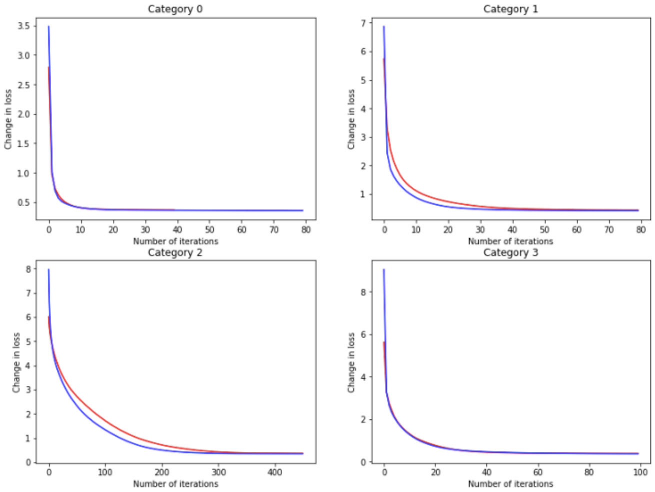

Fig. 1. Effect of non-linear features on training loss. (Red line: with polynomial and multiplication. Blue line: with polynomial, multiplication, logarithmic and square root)

The effect of these cross terms are Fig. 1 shows that adding non-linear features introduces higher precision on describing the data set, as the training loss decreases with more features.

E. Cross validation and Adam

The number of features in the data set increases significantly after feature engineering and hence is predisposed to overfit the data set. Hence, 10-fold cross validation, which is scientifically proved reliable [1], is used to substantiate the model and tune the hyper-parameter.

Adam (Adaptive Moment Estimation) is an extension to Stochastic Gradient Descent to update weights of the model, which computes adaptive or different learning rates for different parameters using the first and the second moments of the gradients (the mean and the variance) [2]. Adam calculates the moving average of the gradient, of which the decay is controlled by parameters (beta1, beta2 ~1), and the square of the gradients. This is beneficial in the project, either when the polynomial degree is high or when cross-terms are complex, to avoid error oscillating on its convergence to the minimum.

III. RESULT AND DISCUSSION

So far, four models are established and optimized to describe the data set. When the fresh test data set comes in, it is classified regarding its 22nd feature PRI_jet_num, and hence for each row there is a corresponding model to predict whether it is a Higgs boson. According to the Leaderboard of Kaggle, the prediction accuracy reaches 83.28% with this methodology.

The initial baseline solution exploits the data set given directly, which engenders the model to fit the irrelevant or redundant data. Further improvement is obtained by categorizing the data and removing unnecessary features on the basis of rational manual interpretation of the data set. Finally, the model is refined by augmenting the feature vector with polynomial terms and cross-terms, assuming there are non-linear relationships between features.

More data does not necessarily refine the machine learning model. In this project, though the original data set is large in scale, it does not produce satisfactory prediction result when directly fed to the machine learning model. On the contrary, appropriate deletion of specific features and generates better solutions, as manual modification of irrelevant data facilitates the machine learning model to describe the underlying distribution.

IV. CONCLUSION AND FUTURE WORK

It can be concluded that the prediction result is a comprehensive outcome of the model selected, the data available and the features prepared. This project implemented six basic methods, of which regularized logistic regression is improved and refined as much as possible. Analysis and comparison are made for each step that reinforces the model. However, this report does not include parallel comparison between methods given same data processing. Further directions include finding better machine learning methods or exploiting the data set with more complex manipulation.

REFERENCES

[1] Fushiki, T. Estimation of prediction error by using K-fold cross-validation. *Stat. Comput.* 21, 137146 (2011). Available at: https://doi.org/10.1007/s11222-009-9153-8

[2] Kingma, D. and Ba, J. (2018). Adam: A Method for Stochastic Optimization. [online] Arxiv.org. Available at: https://arxiv.org/abs/1412.6980