A Machine Learning Application in Higgs Boson Prediction

Zhechen Su, Harshdeep Harshdeep Futong Liu

***Abstract*—With the explosion of data in numerous scientific and experimental disciplines, rational data interpretation and prediction requires machine leaning techniques to create 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 article 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, possible methods introduced during the lectures so far are implemented and tested as baselines to further analysis. Table **??** shows the performance of the six methods without pre-processing except standardization, of which the parameters λ and γ are optimized by grid search. The data set is split with 80% as training set and 20% as test set to compute the prediction accuracy. It can be concluded that all methods except SGD achieve similar prediction accuracy (72% - 74%). However, the prediction of Higgs boson is essentially a binary classification problem. Therefore, regularized logistic regression fits the scenario most suitably, as it works best for binary classification whether the data is linearly separable or not, where the penalty term λ regulates the balance between bias and variance. Hence, regularized logistic regression is selected as the baseline for further improvement and development.

Methods	γ	λ	iterations	Accuracy
Gradient Descent	0.05	-	500	74.38%
Stochastic Gradient Descent	0.005	-	500	69.69%
Least Squares	-	-	-	74.47%
Ridge Regression	-	0.001	-	74.36%
Logistic Regression	0.01	-	10000	72.55%
Reg Logistic Regression	0.01	1	10000	72.81%

TABLE I
Summary of six mandatory methods

B. Exploratory Data Analysis

As observed from the raw data set, the prediction result y is a binary value (either s or b), while the 30 features have numeric values, where there are numerous values of 999.0 separating among valid values. These -999.0s are outside the normal range of all variables (outliers) and hence should be substituted with NaN.

In order to discover the dependencies between the features, the correlation of the features of the data set is computed and compared. Table III 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, according the Table k, five features can be deleted: DER_lep_eta_centrality, DER_prodeta_jet_jet, PRI_jet_subleading_eta, PRI_jet_subleading_phi, PRI_jet_subleading_pt.

```
DER_lep_eta_centrality,
DER_prodeta_jet_jet,
PRI_jet_subleading_eta,
PRI_jet_subleading_phi,
PRI_jet_subleading_pt
```

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. Further observation after sorting the data set by PRI_jet_num, as depicted in table II, shows that the NaN values congregates

Advisor: Dipl.–Ing. Firstname Lastname, Lehrstuhl für Nachrichtentechnik, TUM, WS 2050/2051.

into blocks within category 0 and 1, such that several features become 100% NaN. These features should be deleted as they are arguably irrelevant with the prediction result.

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

Therefore, it is rational to partition the whole data set into four categories depending on PRI_jet_num, since redundant features with NaN can be removed in this way.

For category 0, since its last feature PRI_jet_all_pt also becomes all zero, this feature is also removed.

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. By using 10-fold cross validation, Table IV shows the prediction accuracy of regularized logistic regression with categorization. Apparently categorization refines the model and improves the prediction result.

Category	total features	γ	λ	iterations	accuracy
Cat0	18	0.001	5	5000	72.94%
Cat1	22	0.001	5	5000	68.17%
Cat2	25	0.0001	5	5000	61.65%
Cat3	25	0.001	5	5000	64.98%

TABLE IV
FINAL MODEL WITH FEATURE ENGINEERING

C. Feature Engineering

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. On the other hand, it is highly possible that there are non-linear relationships between features. Therefore, for each feature pair in the present data set, three non-linear operations are undertaken: multiplication, square root and logarithmic, and hence three new combination features are generated and added to the augmented feature array. Table V shows the final result after feature engineering

Category	total features	γ	λ	Degree	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

D. Cross validation for Hyper-Parameter Optimization

The number of features in the data set increases significantly with cross-terms and hence is predisposed to over-fit the data set. In order to identify whether the current prediction model with hyper-parameter **p** is over-fitting or not, a testing data set is required to substantiate the model. Cross validation makes good use of the existing data set where every datum is used for testing for once and training for other times. It is scientifically experimented that 10-fold cross validation generates convincing measuring result [Reference!!] and hence is utilized in this project.

The bias-variance decomposition graph (Fig. 1.), as a use case of cross validation for polynomial degree M adjustment, depicts the effect of M on prediction accuracy. When M is low, the model is too simple to describe the complexities of the data and hence has high bias. When M is high, the model is so complex that it even fits the noise of the data and hence has a high variance. In this case, degree of two achieves satisfactory results.

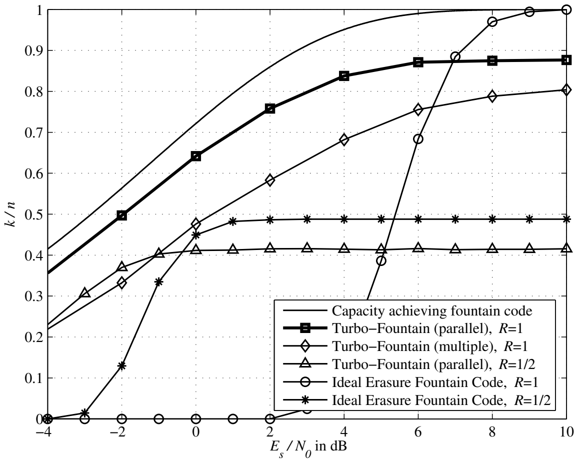

Fig. 1. Bias-variance decomposition for tuning of polynomial degree M

III. RESULT AND DISCUSSION

So far, four models are established and optimized to describe the data set. When the fresh test data set x comes in, it is classified regarding its 22nd feature PRI_jet_num, and hence for each row of x 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 solution, 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 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 is 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] J. Hagenauer, E. Offer, and L. Papke. Iterative decoding of binary block and convolutional codes. *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 429-445, Mar. 1996.
Fushiki, T. Estimation of prediction error by using K-fold cross-validation. Stat. Comput. 21, 137146 (2011).

[2] T. Mayer, H. Jenkac, and J. Hagenauer. Turbo base-station cooperation for intercell interference cancellation. *IEEE Int. Conf. Commun. (ICC)*, Istanbul, Turkey, pp. 356–361, June 2006.

[3] J. G. Proakis. *Digital Communications*. McGraw-Hill Book Co., New York, USA, 3rd edition, 1995.

[4] F. R. Kschischang. Giving a talk: Guidelines for the Preparation and Presentation of Technical Seminars. http://www.comm.toronto.edu/frank/guide/guide.pdf.

[5] IEEE Transactions LaTeXand Microsoft Word Style Files. http://www.ieee.org/web/publications/authors/transjnl/index.html

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