Introduction

In this project, we aim to simulate the process of discovering the Higgs particle since Higgs particles are essential for explaining why other particles have mass. Our goal is to train a binary classifier by using given training set and then use the obtained model to predict whether an event was signal (a Higgs boson) or a background (something else). The methods we used to process the data, and the models we chose to train are shown in Section II. The result is written in Section III. Also, we analyze the advances and the weakness of our approach in Section IV. Section V is the summary of the project.

Models and Methods

First, we began with observing the features of data based on the original CSV file, and we found a large amount of '-999' appeared dispersedly in the data set. Then we found that the column 'PRI\_jet\_num' which is consisted only with the integer number from 0 to 3, has a strong connection with the appearance of the value '-999'. '-999' appears when the value of 'PRI\_jet\_num' is equal to 0 or 1 and disappears when the value is 2 or 3. So, in our first test, we decided to drop all the rows whose 'PRI\_jet\_num' is 0. However, there are still many invalid data left as we chose to reserve the data with their ‘PRI\_jet\_num’ equal to 1. We hence plotted the distribution of value for every feature. According to these pictures, we dropped the column which more than half of their data is '-999'. Because they have too little useful data to analyze (ex: DER\_deltaeta\_jet\_jet). We also dropped the features which have the uniform distribution and Gaussian distribution for that we couldn’t distinguish two types of labels from their distribution (ex: PRI\_lep\_phi, PRI\_lep\_eta). As for the rest of the features, there are some process strong polarization which is quite valuable for the binary classifier.

Furthermore, some features peak at one value and have little invalid data we thus decided to replace their '-999' by their median. Then we used python to perform the data processing: beginning, we removed the label 's' and 'b' in the original file and replaced them by '1' and '-1' respectively. After that, we trained the logistic regression model with the processed data and using the obtained model to predict the test data set. The result of the first attempt is 74%.

The second time, we normalized our data (used in the first experiment) utilizing the standard normalization method, trained a new model and tested the model on the test set. This time, we noticed that even though the accuracy of the prediction did not improve, the computation efficiency and space efficiency improved dramatically. Thus, we decided to keep normalizing the data in every following experiment.

The third time, we classified all samples into four categories according to the value of 'PRI\_jet\_num', and each type of samples corresponds to the value of 'PRI\_jet\_num of' 0, 1, 2, and 3. Furthermore, for each group, we divided them into two subgroups according to their labels and drew the distribution of the feature. So we obtained, for each feature, eight pictures. This time, we tried to study the dependence between two features. We randomly picked two features and calculated their coefficient of correlation. Then we listed all pairs of features which have a coefficient higher than 0.8 which means they have a strong association between them. So for each pair, we saved only one feature out of two and dropped the other one. In our case, we left the column 'DER\_mass\_vis', 'DER\_pt\_h' and 'DER\_sum\_pt'. Furthermore, unlike the first attempt (dropped the column which has too many invalid data or has the uniform distribution), this time, we kept the features with the Gaussian distribution. Then we trained four models separately based on the data with distinct values of 'PRI\_jet\_num' and developed four logistic regression models which have different parameters. We used these models later to predict the labels of our test data. The final score improved to 75%.

For the last experiment, we continued to analyze our features. First, we calculated the log of the features whose values are all positive or calculated the normalized values otherwise. These preprocesses significantly saved our computed loss and storage loss. Then, we used the polynomial with a degree two to augment the four categories of features that we prepared during our third test, and each term in the polynomial has a coefficient one. Next, we fed these data to four models perspectively and used the obtained models to make the prediction. This time, the accuracy has significantly improved to about 82%.

Result

In the last experiment, the four types of samples were processed differently, and then they were trained separately to obtain four logistic regression models with different parameters. According to the training results, the accuracy of the model trained with the samples with the PRI\_jet\_num value of 0 is 0.8426130733738353, the accuracy of the model trained with the samples with the PRI\_jet\_num value of 1 is

0.8014030743835758, the accuracy of the model trained with the samples with the PRI\_jet\_num value of 2 is 0.8288572619543858, and the accuracy of the model trained with the samples with the PRI\_jet\_num value of 3 is 0.8223696083739397. The correct rates of the classifications of these four models are more than 80%. Thus, we thought these models are useful. Finally, we performed the same processing on the test data as the training data and predicted the test data. We obtained an accuracy of 0.82394.

Discussion

In our project, we used nonlinear logistic regression to perform a fit, which is more accurate than using a linear least square model. For samples, we used a polynomial with a degree of 2 to augment features, which improved the accuracy of the model. However, feature augmentation increases the amount of training data, which increases computational complexity and hence dramatically lengthens the time required to train the model.