Dongyang Wang
Stat 535: Statistical Learning
Professor Meila
12/6/2022

Classification of Higgs Boson: A Report on Bagging and Logistic Regression

Part I: Introduction

For the final project of the class, the topic concerns Higgs Boson, "an elementary particle in the Standard Model of particle physics produced by the quantum excitation of the Higgs field, one of the fields in particle physics theory."[1] In particular, the goal of this project is to classify whether an observation is Higgs Boson or not based on the features in the dataset. More specifically, I will train two predictors for this task, and try my best to obtain a low expected classification error. The classification error is defined as "valued at 1 for misclassifying a true negative example and 100 for misclassifying a true positive example".[2]

The dataset is obtained from the UCI Machine Learning Repository, titled *HIGGS Data Set*. "The data has been produced using Monte Carlo simulations. The first 21 features (columns 2-22) are kinematic properties measured by the particle detectors in the accelerator. The last seven features are functions of the first 21 features; these are high-level features derived by physicists to help discriminate between the two classes."[3] For the project, some data preprocessing has been done. By observation, one feature has been removed and the dataset contains 100,000 observations in the training dataset and 10,000 observations in the testing dataset.

To tackle the project, I have chosen two machine learning models: the bagged decision trees and the logistic regression. I will introduce the models in detail in future parts of this report. The

[1] Unknown. "Higgs Boson." *Wikipedia*. Retrieve from https://en.wikipedia.org/wiki/Higgs_boson
[2] Meila, Marina. "Foundations of Machine Learning." *STAT 535 Autumn Quarter 2022*. Retrieve from https://sites.stat.washington.edu/mmp/courses/535/fall22/projects.html
[3] Unknown. "HIGGS Data Set." *UCI Machine Learning Repository*. Retrieve from https://archive.ics.uci.edu/ml/datasets/higgs

structure of this report is as follows. I will first introduce and explain the data cleaning and EDA I have done in Part II, as well as some pre-modeling preparation. I will then detail the work I have done with the bagging model in Part III, and briefly talk about the performance. After that, in Part IV, I will discuss logistic regression and its training process. In Part V, I will briefly conclude the report by talking about the test set prediction and performance, as well as some insights the models have provided for the project.

Part II: EDA and Modeling Preparation

To begin with, I started looking at the structure of the data and the interpretation of the variables. Despite the physics context of the dataset, it is still manageable to understand its structure. For example, there are 100,000 observations with 28 variables in the training set. All of the variables have a data type of float. Different variables have different distributions, in terms of their range, variance, and median. There is no null value in the dataset.

Among all the variables, the first one is the dependent variable, namely whether the Higgs Boson is present. The distribution of this variable is unbalanced, since there are only 1004 positive cases out of 100,000 observations, and that gives us a 1 in 100 chance.

Moreover, the correlation between the variables is as shown in the plot in the appendix of this report. There are only a few pairs of variables that are highly correlated, and notably there is no single variable that is highly correlated with the dependent variable. And the correlation between the last few variables is potentially due to the fact that these variables are engineered by combining information from different raw variables, whereby they can contain the same information as each other.

After the EDA, I have started working on preparing for the modeling afterwards. First, I separated the Y and X from the datasets to ease the use of the *sklearn* package. To accommodate the difference in the distribution of the features, I have scaled all variables in the X subset of data with a standard scaler, thereby "removing the mean and scaling to unit variance."[4]

[4] Unknown. "Standard Scaler." *Scikit Learn*. Retrieve from https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

So the general approach I planned to take was a validation set approach. Basically, I split the training set into two datasets at a proportion of 80% vs 20%, the first being the training set and the second being the validation set for training the models. While splitting the datasets, I also applied a stratified approach in terms of the number of Higgs Boson being present. Since there are a very small portion in the dataset (1% as described earlier) with a positive value in the dependent variable, it's crucial that after splitting the dataset, the dependent variables are distributed evenly to allow for modeling with less bias.

Part III: Bagged Decision Tree

To construct the bagged decision tree, I have attempted a few methods by adjusting different parameters and testing out the performance on the validation set. First, I modeled with different tree depths for the decision tree classifier that I was about to use for the bagged trees. It turns out that after several attempts, not restricting the tree depths would be optimal.

Therefore, I have kept the decision tree classifier intact, which I would then use to construct a bag of trees. I have set the bootstrapping with replacement in the bagging algorithm, and started tuning the model. I first tuned the number of samples drawn from the training set to train each base estimator, in this case, each individual tree. It seems that when that number is around 0.8, i.e, the sample is 80% of the original data size, the performance in terms of both the out of bag score and the classification error are optimal.

One other parameter I tuned was the number of base estimators in the ensemble. I tried different values from 10 to 1,000, and it turns out that it's not the more the better. So an optimal point is approximately 200, which I later used in the final bagging model. Admittedly, it was not the best number in magnitude, 500 would be slightly better. But accounting for the very small increase in prediction performance, possibility for variance with new datasets, and the great increase in running time, I decided to use 200 in the final model.

After finalizing the parameters, I started testing out the model on the validation set. I constructed the bagging model with 200 trees within the ensemble, using 80% of the training data size for

bootstrapping with replacement, and no specification of tree depth in the decision tree classifier. With this model, I was able to obtain the out of bag score at 0.936 and a classification error of 0.3717. Moreover, I constructed a confusion matrix to visualize the prediction, and obtained a false negative rate of 0.06 and false positive rate of 0.31. These numbers are what I put in the submission for guessing the test set performance.

Part IV: Logistic Regression

The fitting of the logistic regression is rather simple. I constructed five models in total, including different types of solvers and different penalties. The solver-penalty pairs include: lbfgs optimizer with L2 penalty, liblinear optimizer with L1 penalty, liblinear optimize with L2 penalty, saga optimizer with elastic net penalty with L1 ratios from 0.1 to 0.9, and lbfgs optimizer with no penalty. Out of these five models, the first four models have been constructed with a cross validation estimator, which is "an estimator that has built-in cross-validation capabilities to automatically select the best hyper-parameters."[5]

While fitting the models, I set the balanced argument in each model to account for the fact that the distribution of the dependent variables is unbalanced. After fitting these five models, I have generated the confusion matrices to compare the performances. With different results, I found that the L1 penalty and the elastic net penalty models classify everything into the negative category. This would not be ideal for our project. Therefore, I removed these two models and compared the other models. The rest of the models have very similar performance in terms of the classification error.

So, I have then generated the probabilities from each model regarding their predicted values. With the probabilities, I tested out each possible threshold for classifying the results and chose the best one in terms of the classification error. After doing this for all three models, the results are surprisingly the same. The three models generated exactly the same threshold, as well as the predictions and the confusion matrix. Therefore, using which model is irrelevant as long as the

[5] Unknown. "Glossary of Common Terms and API Elements." *Scikit Learn*. Retrieve from https://scikit-learn.org/stable/glossary.html#term-cross-validation-estimator

threshold is used for prediction. For convenience, I used the model with the lbfgs optimizer and L2 penalty, and chose the threshold as 0.476 for a positive class.

Fitting the final logistic model on the validation set renders a classification error of 0.7549. Moreover, I constructed a confusion matrix to visualize the prediction, and obtained a false negative rate of 0.46 and false positive rate of 0.29. These numbers are what I put in the submission for guessing the test set performance.

Part V: Test Set Results and Conclusion

The test set contains 10,000 observations with 28 variables, the exact same structure as the training dataset. Based on previous validation set results, I have quite accurately guessed the performance of both models as shown in Steve's presentation poster. Overall, the bagged decision tree has good performance and the logistic regression performs just fine. But both models are better than random guessing because their False Positive Rate and False Negative Rate are all below 0.5.

Therefore, with these two models, we can predict the existence of Higgs Boson particles given the features.

References

Meila, Marina. “Foundations of Machine Learning.” *STAT 535 Autumn Quarter 2022*. Retrieve from https://sites.stat.washington.edu/mmp/courses/535/fall22/projects.html

Unknown. “Higgs Boson.” *Wikipedia*. Retrieve from https://en.wikipedia.org/wiki/Higgs_boson

Unknown. “HIGGS Data Set.” *UCI Machine Learning Repository*. Retrieve from https://archive.ics.uci.edu/ml/datasets/higgs

Unknown. “Glossary of Common Terms and API Elements.” *Scikit Learn*. Retrieve from https://scikit-learn.org/stable/glossary.html#term-cross-validation-estimator

Unknown. “Standard Scaler.” *Scikit Learn*. Retrieve from https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

Appendix

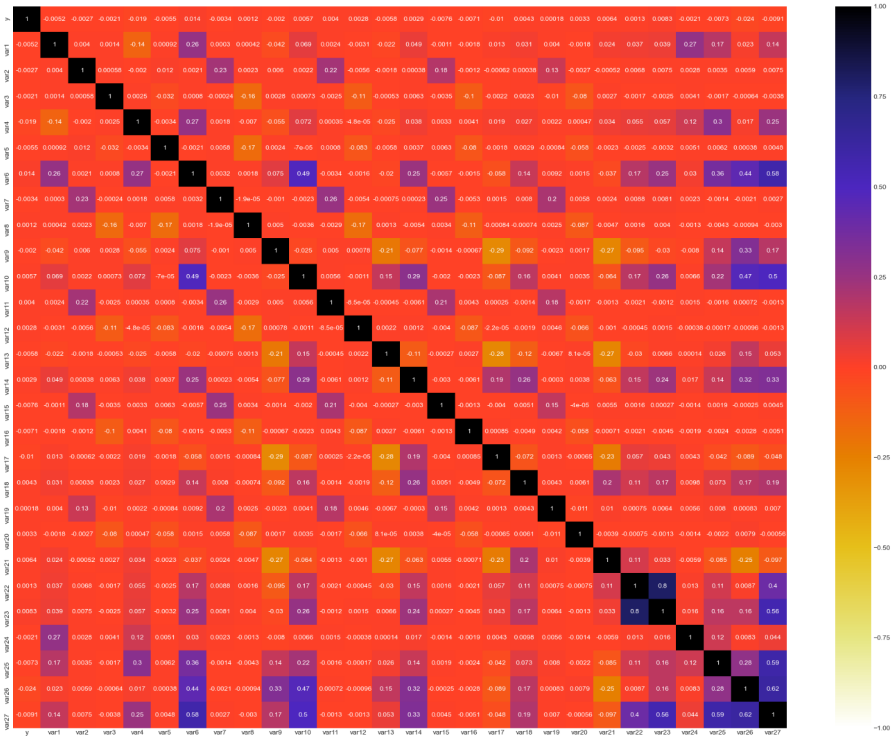

Figure 1: Correlation Plot