CISC 372- Advanced Data Analytics

Assignment 1

Author: Runze Yi

First Trial:

In this trial there wasn't a lot of stuff changed from the original version of the script. After browsing through the data sets I discovered that one of the selected features, is_business_travel_ready, has basically all f's in it. This feature appeared to be the same for all ratings, therefore it should appear to be not important to the final evaluation. However, after changing that feature to a feature that ranges more among the ratings, the Leaderboard score appeared to be less than before.

Second Trial:

Here I started to play with more features.

Third Trial:

I just submitted the original to see what the score was, since I believed that I was making improvements on the model but actually I wasn't. The result came out to be as expected, it matches with the given score on the Leaderboard.

Fourth Trial:

I changed the feature 'is_business_travel_ready' to 'host_location', then started to tune the hyper-parameters. Increase the max-depth from 20 to 50, increase number of estimators from 100 to 150, change strategy of imputer from mean to median, etc. Performance finally started to increase.

Fifth Trial:

In this submission I added a new feature of 'street', and further tuned the hyper-parameters, including trying out with learning rate with different sizes of model.

Here I found out that in the hyper-parameters(param_grid), the imputer strategy would change the strategy defined in the numeric_transformer. Therefore no matter what you use for the transformer, it would change it to what you set in the param_grid. Originally it was mean in the grid and median in the transformer. I have made them the same for clearness and to avoid confusion. What I also discovered is that since there are only less than 10 features, the tree won't expand to the number of max depth as defined in the grid. I've set the max_depth to 10 and will consider raising it if needed.

To my surprise, performance lowered as the best-score used in the grid search increased. This appeared similarly as my first trial where features that appeared to have no contribution to the model would increase performance.

Sixth Trial:

To prove my prediction made in the fifth trial, I did another try with only the hyper-parameters changed comparing to trial 4, and got the same result.

Seventh Trial:

Made small adjustments to the fifth trial on different features. What I used was adding another feature of 'state' which did nothing since the Leaderboard scores appeared to be the same.

Eighth Trial:

I changed the features back to what they were originally and wanted to see the impact of only the hyper parameters I had. The result came out to be disappointing as it lowered the performance comparing to the original version.

This thought appeared as when I only changed the features, the performance lowered. Therefore I wondered would the performance increase if I change then back with my tuned hyper-parameters.

Ninth Trial:

Change the feature of which I though invaluable(is_business_travel_ready) to 'street', just wanted to see how it goes.

Tenth Trial:

Explored the new model LogisticRegression. Tuned the hyper-parameters to achieve the highest score. Explored different parameters like C(controls level of regularization), mas_iter(max iterations before it converges), random_state(the random generator for shuffling data. The np(numpy) in then code would take place if random_state is set to none, just wanted to see influence of this parameter. Then I explored it with the set of features I gathered from previous trials. The pattern appeared to be the same as the features that achieves best score(grid search score) in XGBoosting would still do best with LogisticRegression. Results of LogisticRegression appeared to be better than ones from XGBoosting.

Questions to be answered:

1. In my opinion, the Leaderboard is designed to accept 3 submission per day for 2 reasons. First of all, this encourages students to try out this assignment constantly and therefore come up with new ideas and thoughts everyday. If it is allowed unlimited number of trials per day, then students might not have the chance to think more about their ideas and let then grow into bigger and more valuable ones. The second reason is to punish students that tend to start working on the assignment only on the last day or two. With limited number of submissions, it is really hard to get the idea of how the model performed and how to improve it.

2. The private Leaderboard with another set of testing data is used for evaluation instead of the public Leaderboard because as we try further and further on the public Leaderboard to achieve higher performance on it, out models might overfit only to that. Such problems should be avoided. Therefore 2 different testing data sets were used for implementing model and evaluation.

3. Eventually I chose to use Logistic Regression as my model. I did comparison with my results with Logistic Regression and the original XGBoosting, and found the former to have better performance. The flexibility was controlled through hyper parameters like max_iter, C, solver, etc. I tried them all at first by only changing one of them at a time, see what the influence would be, then adjust them to have best performance. The best solution I came up with was only to change the iterations it takes towards converging before it stops(max_iter), and to leave other hyper-parameters as default.