Submission to ACM Research Application Coding Challenge

Approach

I start this project on Jan 29. Because I have previous experience on Kaggle, I know that This Coding Challenge is more or less like a Kaggle Classification Competition. And I know usually Gradient Boosting Algorithms perform extremely well on handling missing values, Preventing overfit, and usually have better Accuracy score than Naive Bayes Classifier or K-NN, etc. So I chose Catboost (a gradient boosting decision tree) as my ML models to classify the Mushroom.csv file.

If you have a interest on how I implement Catboost to train a ML model that can Classify the whether a Mushroom is Edible or Poisonous please click the link below!

Colab Links:

https://colab.research.google.com/drive/1MFnD6XHkVPQ3mDCMtfm55dPanHmqynCv?usp=sharing

Internal Evaluation

The figure below the plot shows the progress of model training

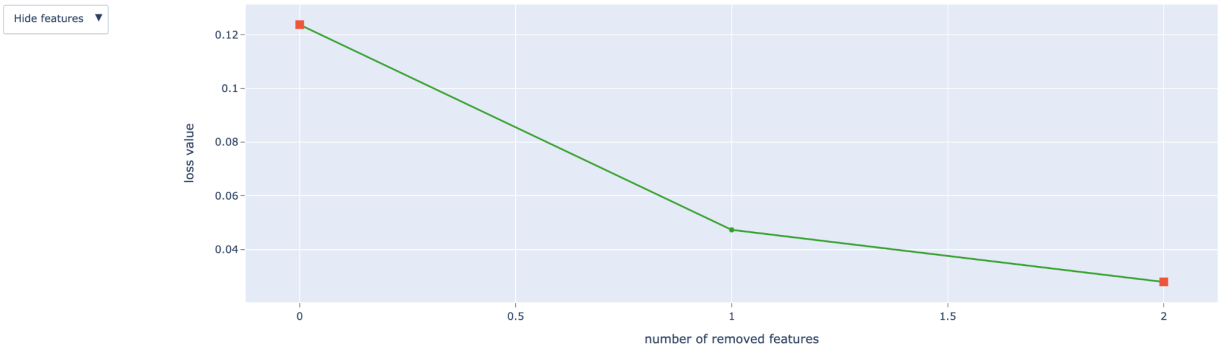
```
Learning rate set to 0.049518
Step #1 out of 1
0:      test: 0.9834197 best: 0.9834197 (0)     total: 15.9ms   remaining: 15.9s
200:    test: 1.0000000 best: 1.0000000 (8)     total: 10.3s    remaining: 40.8s
400:    test: 1.0000000 best: 1.0000000 (8)     total: 30s      remaining: 44.9s
Stopped by overfitting detector  (400 iterations wait)

bestTest = 1
bestIteration = 8

Shrink model to first 9 iterations.
Feature #3 eliminated
Feature #5 eliminated
Train final model
0:      test: 0.9875434 best: 0.9875434 (0)     total: 75.9ms   remaining: 1m 15s
200:    test: 1.0000000 best: 1.0000000 (15)    total: 11.6s    remaining: 46.3s
400:    test: 1.0000000 best: 1.0000000 (15)    total: 23.2s    remaining: 34.7s
Stopped by overfitting detector  (400 iterations wait)

bestTest = 1
bestIteration = 15

Shrink model to first 16 iterations.
```



I actually use a very simple method to evaluate the model. I use the build-in parameter of Catboost called “eval_set= ‘X_valid, Y_valid’”, to evaluate the models.

Author: Frank Gao QXG190000