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

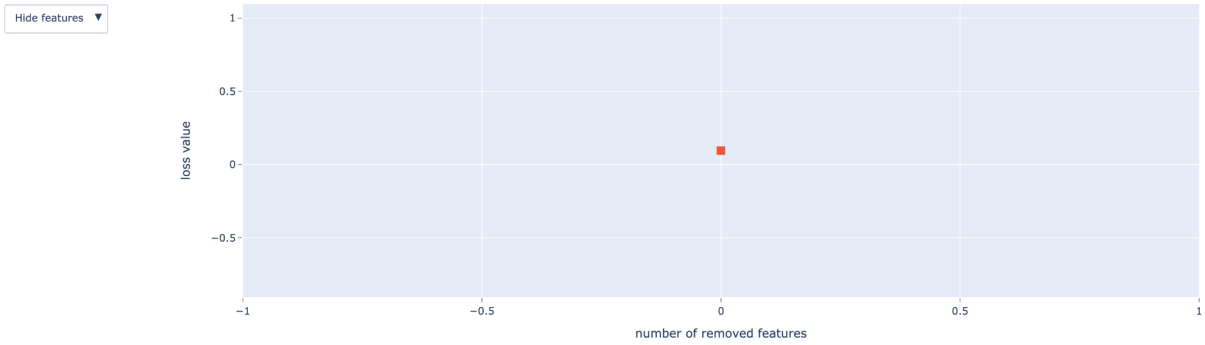
Internal Evaluation

The figure below the plot shows the progress of model training

```
The number of features selection steps (1) is greater than the number of features to eliminate (0). The number of steps was reduced to 0.
Learning rate set to 0.049518
Train final model
0:      test: 0.9868458 best: 0.9868458 (0)     total: 16.5ms   remaining: 16.5s
100:    test: 1.0000000 best: 1.0000000 (10)    total: 5.12s    remaining: 45.6s
200:    test: 1.0000000 best: 1.0000000 (10)    total: 11.4s    remaining: 45.3s
Stopped by overfitting detector  (200 iterations wait)

bestTest = 1
bestIteration = 10

Shrink model to first 11 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