CatBoost

Aaryan
CO21BTECH11001

CatBoost (Categorical Boosting) is an implementation of gradient boosting, which uses binary decision trees as base predictors.
Here are the two main aspects considered in this algorithm:

1. Ordered boosting, a modification of standard gradient boosting algorithm.

2. Categorical features, a new algorithm for processing categorical features.
However, it is not restricted to only categorical data but also supports Numerical and Text features, but it has an effective handling technique for categorical data.

When the distribution of the estimated model is different from the distribution of the testing samples, the model is biased and could lead to overfitting in some cases. This is called prediction shift problem which is often faced by other gradient boosting algorithms.

To avoid prediction shift, ordered boosting is introduced.

Ordered Boosting:

In ordered boosting, a new training dataset is obtained in each step of boosting. This means that the model is trained such that the model we obtained previously is applied to this set of new training samples. This guarantees that the model in which we have obtained previously has not seen the labels in the new training set.

In ordered boosting, the algorithm starts with generating $s + 1$ independent random permutations $\sigma_0, \sigma_1, \dots, \sigma_s$ of the training dataset.

$\sigma_1, \ldots, \sigma_s$ contributes to the internal nodes of a tree where the evaluation of the splits takes place, whereas σ_0 contributes to the terminal nodes where the leaf value is determined.

The trees built under this model are symmetric. Define $M_{r,j}(i)$ as our current model for the i^{th} sample based on the first j samples. A random permutation σ_r is sampled from the $s+1$ independent random permutations of the training dataset.

Let gradient of loss function $= grad_{r,j}(i) = \frac{\delta L\left(y_i, M_{r,j}(i)\right)}{\delta s}$

The leaf value for the tree constructed based on the i^{th} sample will then be equals to the average of the gradients computed= avg $grad_{r,\sigma_r(i)-1}$

When a tree, T_t is constructed, the tree itself is used to boost all the existing models $M_{r',j}$. In our final model, F, the leaf values are computed using the standard gradient boosting procedure. σ_0 is used to determine the final leaf values. Therefore, once we obtain our final model, we match the training samples, says i^{th} sample, to $leaf_0(i)$ and so on.

Questions:

1. CatBoost is
   (a) Gradient Boosting Algorithm
   (b) Bagging Algorithm
   (c) Supervised learning algorithm
   (d) Unsupervised learning algorithm
   Ans. (a), (c)

2. When did prediction shift happens?
   Ans. It happens when the distribution function of the training set is not the same as the testing set.

3. How does CatBoost *cures* prediction shift?
   Ans. With every iteration in CatBoost, it ensures that the model is

now being trained on a dataset whose labels are not seen yet by the model. It achieves so by random permutations of the training dataset. This whole procedure is called ordered boosting.

4. What is the advantage of symmetric decision trees in CatBoost?
   Ans. This helps to improvise the algorithm in terms of its runtime.

5. What are the disadvantages of using CatBoost?
   Ans. There is a difficulty in finding the correct set of parameters.