Binary Classification of Imbalanced data from Bosch Production Line

P-1

1 Data Set

The data source is an ongoing Kaggle competition: Bosch Production Line Performance (`https://www.kaggle.com/c/bosch-production-line-performance/data`). The data represents measurements of parts as they move through Bosch's production lines. The goal is to predict which observations will fail quality control ('Response' = 1). The raw data is separated by the type of feature they contain: numerical, categorical, and date. The date features provide a timestamp for when each measurement was taken.

2 Project Idea

To solve this binary classification problem, we will follow the general machine learning pipeline. The first step is to generate new features from raw data(including stratifying and sampling). Then select machine models and do hyperparameter selection (grid search, bayes search, etc.). Last, ensemble several models based on train data and make a prediction of test data.

We will focus on the following areas: feature engineering, sampling and machine learning model. For feature engineering, we will do data exploration and feature reduction, the size of raw data is large(15.4GB) and doesn't fit into a single machine without feature reduction. The goal is to fit important features into a single memory with 16GB memory. For sampling, we will testify different methods such as upsampling, downsampling and smote to process imbalanced data. The samples are highly imbalanced(6879:1176868).For machine learning model, gradient boosting tree is popular in Kaggle competitions for structured data. We will firstly use gradient boosting tree and then compare its performance with other machine learning models. Moreover, the shortcoming of gradient boosting tree is speed, it is kind of slow compared to random forest. We will testify three implementations of speeding up gradient boosting tree: Xgboost, FastBDT and SAS Viya. We want to get some insights how to improve the performance of gradient boosting tree for imbalanced data.

3 Software

python open source libraries: pandas, scikit-learn, matplotlib

gradient boosting tree libraries: xgboost and FastBDT

cloud platform: SAS Viya

4 Papers to read

[1] Chawla, Nitesh V., et al.(2002) SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, , pp. 321–357.

[2] Tianqi Chen, & Carlos Guestrin. (2016) XGBoost: A Scalable Tree Boosting System. *KDD*.

[3] Keck T. (2016) A speed-optimized and cache-friendly implementation of stochastic gradient-boosted decision trees for multivariate classification *arXiv preprint arXiv*: 1609.06119.

5 Teammate and work division

Xi: Data exploration and feature reduction for categorical data, SMOTE

Liang: SAS Viya machine learning pipeline, Tree building for imbalanced data

Weijie: Employ and compare xgboost and FastBDT, hyperparameter optimization

Yeojin: Data exploration and feature reduction for date data, upsampling and downsampling

6 Midterm milestone

The project can be divided into three phases. The first phase is to do feature engineering and get first submission in Kaggle by SAS Viya. The second phase is generate submission by xgboost and FastBDT in single machine with new features. The last phase is to try different sampling methods and get insights how to develop gradient boosting tree for imbalanced data. The goals of the first two phases are midterm milestone.