**Models Selection**

After FE, we must check the effect of FE. For the dual consideration of efficiency and accuracy, we first need to screen out a fast and accurate model as the prior model for classification.

Due to the excellent historical performance of GBDT algorithms on binary classification problems, our prior model mainly chooses among various tree algorithms.

The following is the performance of each model that we use to simply use label encoder and standard scaler to process the original data.

The top five models with the highest accuracy and MCC are all GBDT algorithms. Among them, the best performing algorithms are marked as yellow, ACC, AUC, and Precision of the cat boost model perform best, while F1, Kappa, and MCC of light GBM perform the best.

Given the excellent MCC performance of Light GBM and its efficient training speed, we choose Light GBM to show the results of FE. I seldom talk about algorithms here, because algorithms of light GBM will have a lot of classmates willing to share.

**Choose Light GBM**

The accuracy of the model after FE has been reduced and MCC has been extremely improved. Why? I will explain later.

**Hyperparameters**

These are the hyperparameters of the model. There are not many hyperparameter changes, so in the case of controlling variables, the comparison of our models intuitively reflects the role of FE.

**Confusion Matrix**

Look at our confusion matrix and precision error. Here our confusion matrix is ​​reversed. Due to the large gap in the number of two-category labels in the original data, over-sampling yes-label and under-sampling no-label at the same time to ensure the effectiveness of the training for the two labels. This makes our training results more balanced.

**Evaluation**

Consider the real business scenario, as a marketing analyst for a bank; after that, my partner will consider the identity of the client's agent analyst; you want to identify users who may subscribe to a term deposit but have not yet subscribed. Such classes will belong to False Positives users who are expected to make a subscription but do not make it. Therefore, in addition to the non-zero entries in TP and TN, we also prefer non-zero entries in FP and work on reducing them. Therefore, the accuracy of the model depends on the goal of the prediction exercise.

**Recall** is calculated as **TP divided by TP plus FN**. It provides us with the percentage of correctly predicted target customers.

**Specificity** is calculated as **TN divided by TN plus FP**. It gives us the percentage of non-target customers that are correctly predicted.

**Precision** is calculated as **TP divided by TP plus FP**. It provides us with the efficiency of prediction because we can get as much TP with as few resources as possible.

**ROC**

The Receiver Operating Characteristic curve is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters: **Precision & Recall**. And AUC stands for "Area under the ROC Curve." That is, AUC measures the entire two-dimensional area underneath the entire ROC curve (think integral calculus) from (0,0) to (1,1).

After doing FE, the performance of yes-yes as TP's ROC curve and the precision-recall curve has greatly increased. Observing the ROC curve, the AUC of class 1 increases from 0.93 to 0.96.

***The following is a cluster analysis of Boundary, Dimension & Manifold.***

A manifold classification is divided into several mutually exclusive classes according to some given characteristic and then each class is divided by reference to some second, third, etc. characteristic. After FE, the main feature dropped from 5 to 2, and the boundary between the two categories became obvious, which also confirmed the success of our FE.

**Threshold Plot.**

Previously, when using the ROC curve and precise recall curve, the best or optimal threshold of the classifier can be directly calculated. To be more precise, it is possible to use a grid search to tune the threshold and locate the optimal value. The result clearly shows the optimal threshold **tr** = 0.38 adjusted when converting the probability into a clear class label for imbalanced classification. After FE balanced the number of labels, **tr** = 0.46.

**Performance**

From Calibration Plots, Validation & Learning curves, the performance of the model has also been improved.

**Interpretability**

The interpretability and feature selection of the model is explained by my teammates.

**Prediction**

This is a comparison of our prediction results before and after FE. It seems that although MCC has decreased, TP has increased, and FN has decreased. Our key **Precision** and **Recall** have increased a lot, that is, our work performance and efficiency have been obtained a great improvement.

The analysis of features can let us know more about our customers. Let me ask my partner for an analysis.