We used different models to predict the data we processed from week 7. Since there is no separate test set in our data, we used a train\_test\_split on the training dataset just like the validation process at the beginning. Below are all the attributes that we are used to predicting if one customer is a repeat buyer. The target is a label which 0 means non-repeat buyer and 1 means repeat buyer, and we tried Decision Tree classification, Logistic regression, Naive Bayes, and NN classification.

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable.

A decision tree is built by splitting the source set based on splitting rules by attributes. This process is repeated on each derived subset recursively and completed when the subset at a node has all the same values of the target variable.

Naïve Bayes is calculating the conditional probability of a class label given a data sample. For our model, we have the Probability of attributes given the label value and want to predict the Probability of the label given the attribute values, which we are predicting if the user will become a repeat buyer given information in our attributes data.

Surprisingly, the accuracy rates for all the models are very high, some are even above 90%. Then we checked the dataset and found that the number of non-repeat buyers is more than 15 times the number of repeat buyers, which means that if our model predicts every user as a non-repeat buyer, we can also get an accurate baseline of 90%.

To deal with our imbalanced data, we tried several data augmentation methods including increasing the size of the number of repeat buyers by copying the data or manually creating a test dataset that contains half repeated buyers and half non-repetitive buyers. However, if we use the stratified splitting on the train\_test\_split, the test dataset is still biased and cannot solve the overfitting issue.

In the 9th week, Scott found an approach called Inverse Probability Weighting which can solve the imbalance problem. Inverse probability weighting is a statistical technique for calculating statistics standardized to a pseudo-population different from that in which the data was collected. To apply this method, we added class-weights according to the class distribution. Class-weights is the extent to which the algorithm is punished for any wrong prediction of that class. After trying several models with the weight, we got much lower but more meaningful accuracies around 55% on average. It turns out that the weighted approach avoids the overfitting problem and shows the real accuracy rate when predicting new users.

After trying several models with weighted prediction, we found that Naïve Bayes and Support vector machine cannot apply to our model because after we did the standardization, some attributes values are negative and MultinomialNB assumes that features have a multinomial distribution which is a generalization of the binomial distribution. Moreover, to change our data into categorical, we need to bin the data, which will cause the information loss. (For example, 5.5 and 5 are the same)

Finally, we made our test data by stratified simple random sampling which we had 2000 samples for each kind of target, and trained our data with logistic regression, decision tree, and random forest. Now let me show you what we got.

As you can see, this is the logistic regression comparison result.

The unweighted accuracy is about 0.5.

The weighted accuracy is about 0.6.

This is a decision tree comparison result.

The unweighted accuracy is about 0.5.

The weighted accuracy is about 0.56.

This is the random forest comparison result.

The unweighted accuracy is about 0.5.

The weighted accuracy is also about 0.5.

From the bottom two pictures, we can see that Action count 0 which refers to user clicking and seller item count are the most important features for random forest classification.

We also got a comparison chart for the predictions.

It is worth mentioning that the best accuracy score provided by Tmall official team is 70%, while the highest accuracy rate we got with weighted logistic regression is only 10% lower than the baseline. In general, the weighted method increases our accuracy rate.