Individual Report

——Zixuan Huang

1. Introduction

We used "The Instacart Online Grocery Shopping Dataset 2017" as the data set for our project. This data set contains six sub sets, including orders.csv, products.csv, order_products_*(train/prior).csv, departments.csv and aisles.csv. All the features in original dataset covered the basic information about the ordering history of customers. What we will do in this project is to predict whether the customers will buy specific products in their next orders. So the first thing we need to do is doing the exploratory analysis and also cleaning the data sets. Next, base on the results of the EDA, we need to create some features that we think might related to our question, since the feature in the original datasets is not suitable for modeling. Then we can build different models using the features we created and also make evaluation of those models. The last thing we will do is use the best model to get our predictions.

Therefore, the overall division of coding part is divided into three parts, which are EDA/data preprocessing, feature engineering and model and summary; the ways of presentation are summary report, slide and GUI, and each of us is responsible for one part. Although it seems each of our task is independent with each other, actually we discussed all the time and some of the works are intersected.

2. My work

1. Data preprocessing

I merged product.csv, departments.csv and aisles.csv and generate a new data frame named 'product', about which I checked the shape, columns, data types and missing value. This data frame is used for the exploration of product information.
I also vertically joined train.csv and prior.csv and then merged it to orders.csv and product data frame. This new data frame I just created is called 'order_flow'. I will explore the ordering information in this data frame. I also checked the structure and summary of this data frame.
These newly merged data frames mentioned above are used for EDA analysis. I mainly used the 'group by' function in Pandas to get the results and used Seaborn for visualization.

2. Part of EDA

I am mainly responsible for the second part of EDA. I first merged product.csv, aisles.csv and departments.csv into a new data frame called 'product' and also explored basic information about products. In this section, I found the how many products are in each department and each aisle. Next, I merged 'product', train.csv and prior.csv. In this part, I calculated the sales in each department and aisle and the reordered rate of products. I basically used group by to calculate these values and used seaborn to present the plots.

3. Naïve Bayes model

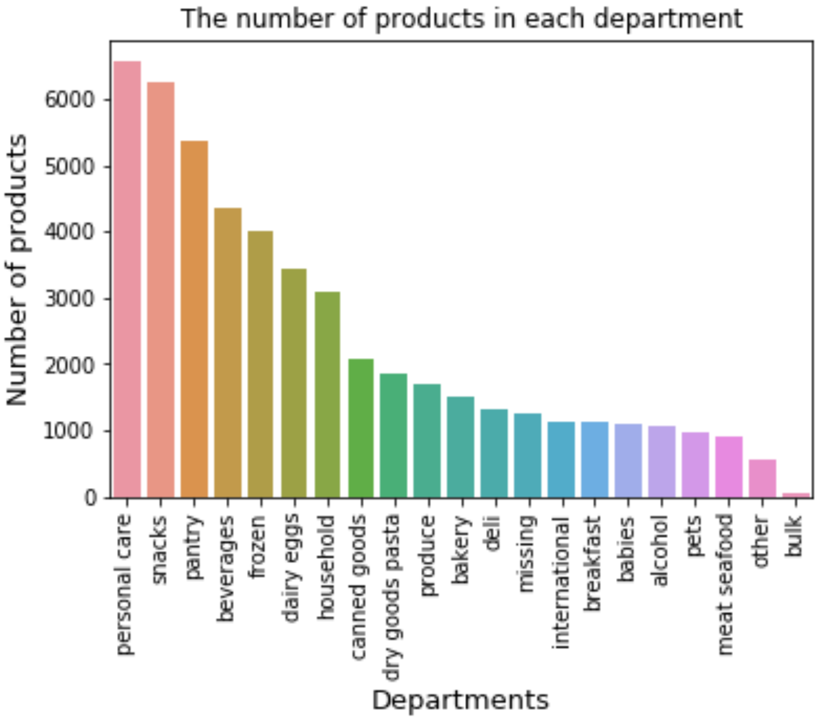
For the modeling part, I tried Naïve Bayes model twice. I also compared the results of before oversampling model and after oversampling. Unfortunately, both the results are not as good as we expected. So we decided not to use this model.

4. GUI

Thanks for my friend, KK, who helped me to finish the coding for GUI. He taught me how to implement GUI by using Tkinter package. The code part of GUI contain several classes and functions which are used to run the windows. And then I merged the code of our projects into those classes and functions.

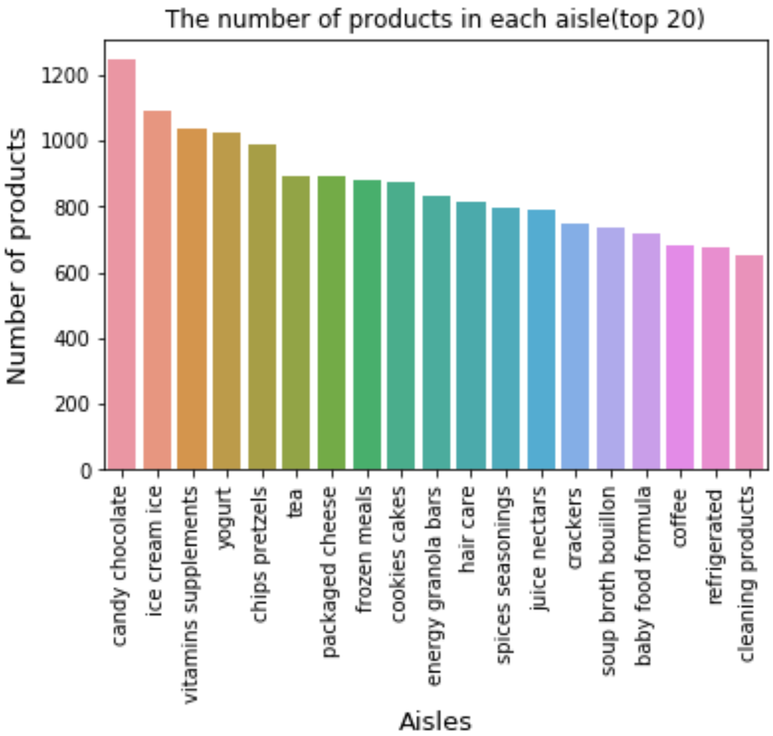
3. Results

1. EDA(second part)



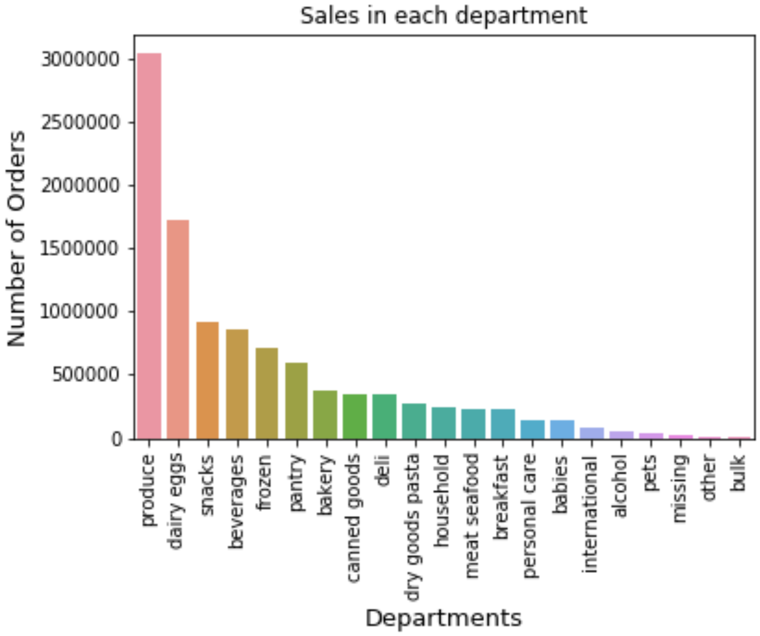
This plot presents the number of products in each department. It is calculated by grouping 'department_id' and then counting the number of products in each department.

The most five important departments are personal care, snacks, pantry, beverages and frozen. The number of items from these departments were more than 4,000 times.



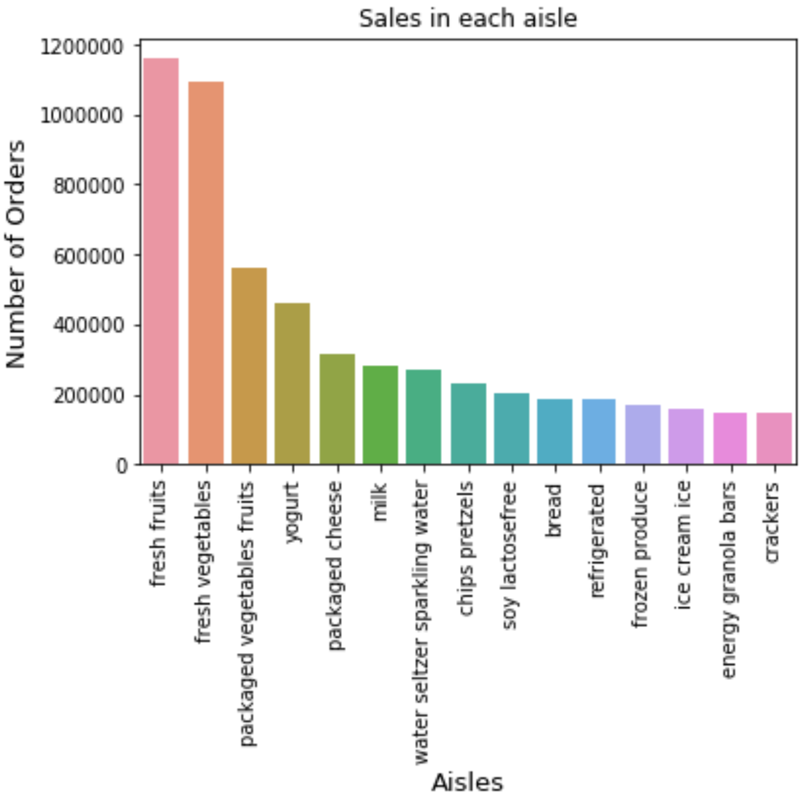
This plot shows the number of products in each aisle and only the first 20 aisles with the most number of products are presented. The way to calculate the result is the same as above, but group by 'aisle_id'.

The most three important aisles are candy chocolate, ice cream and vitamins supplements.

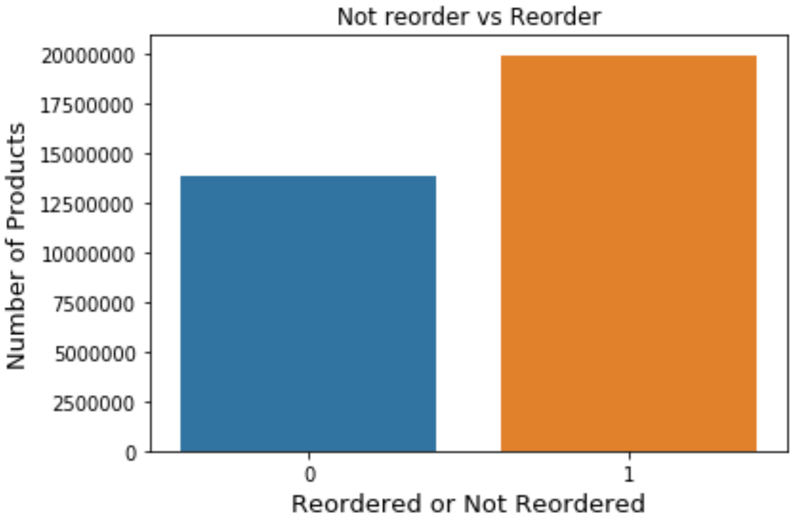


It shows the sales in each department. I first grouped 'department_id' and then calculated the number of orders in each department to get this result.

The most three popular departments are produce, dairy eggs and snacks.

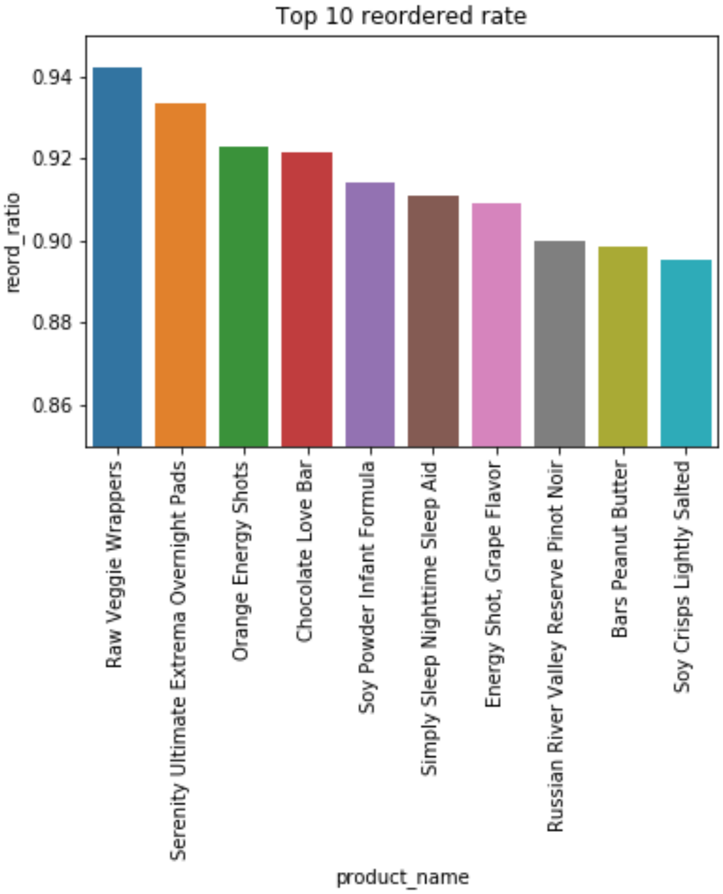


Here is the plot showing top 15 popular aisles. It is the result of total number of orders in each aisle.
The top three best-selling aisles are fresh fruits, fresh vegetables and packaged vegetables fruits.

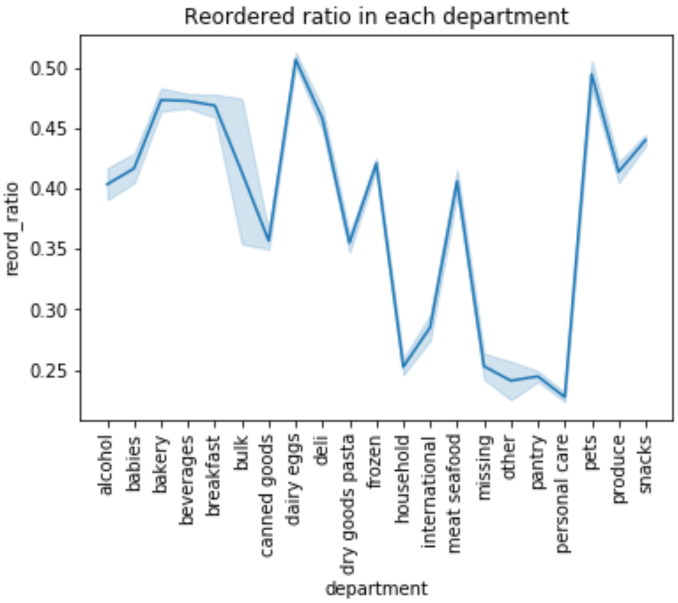


The ‘reordered’ variable has two levels, 0 and 1, which represent this product has only been bought once, this product has been reordered by customers before, respectively. So by counting the number of 0 and 1, we can see how many products have been reordered.

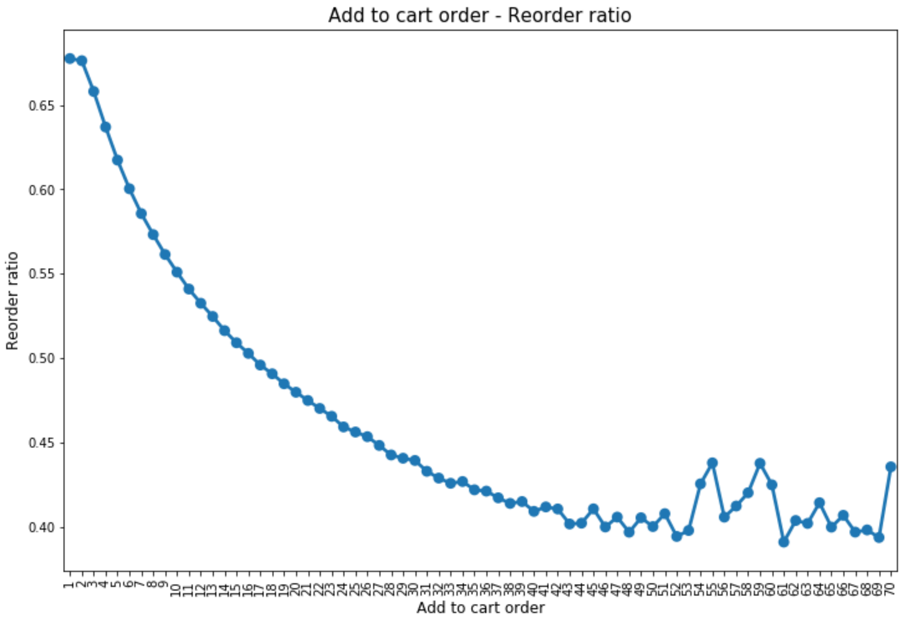
There are 19,126,536 products previously reordered by customers, reordered products take 0.59 % of ordered products and 13,307,953 products are not ordered by customers before, non-reordered products take 0.41 % of ordered products.



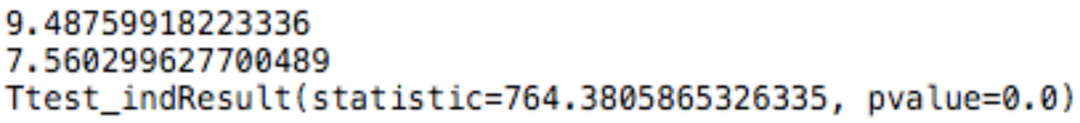
I also calculated the reordered rate of each product, which is the total number of times the product was reordered by the same customer divided by the total number of times the product was purchased.
The top three products with the highest reordered rate are Raw Veggie Wrappers, Serenity Ultimate Extrema Overnight Pads and Orange Energy Shots.



This plot shows the reordered rate in each department. Personal care has lowest reorder ratio and dairy eggs have highest reorder ratio.



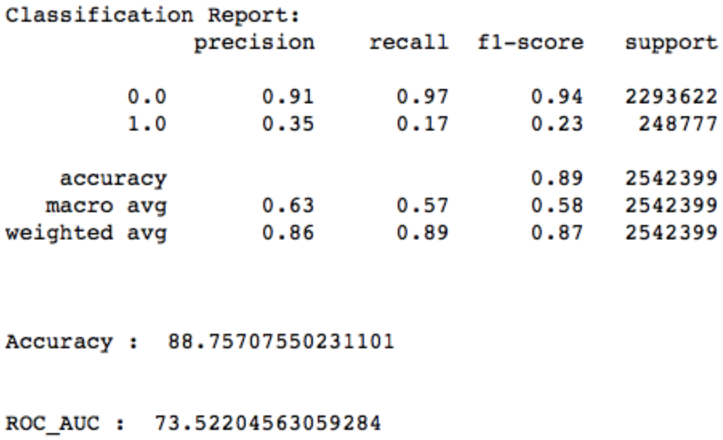
‘add_to_cart’ in the data set means the order of this product added to cart by each customer. This graph tells us orders placed initially in the cart are more likely to be reordered than the one placed later in the cart.



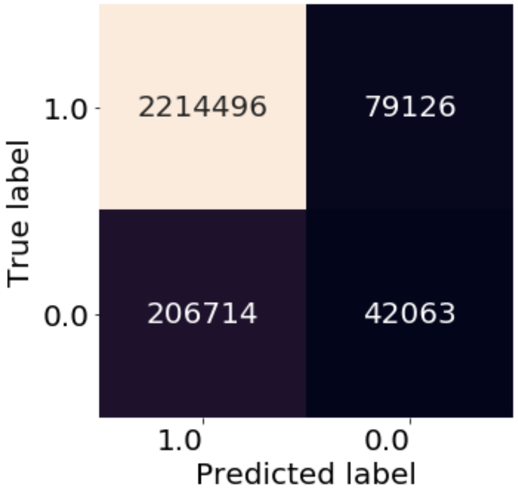
```
9.48759918223336
7.560299627700489
Ttest_indResult(statistic=764.3805865326335, pvalue=0.0)
```

And I did t-test to verify whether the sequence of adding to cart are significantly different between reordered products and not reordered products. The results show that the p-value is smaller than 0.05. Therefore, the sequence of adding to cart significantly influence whether the products being reordered.

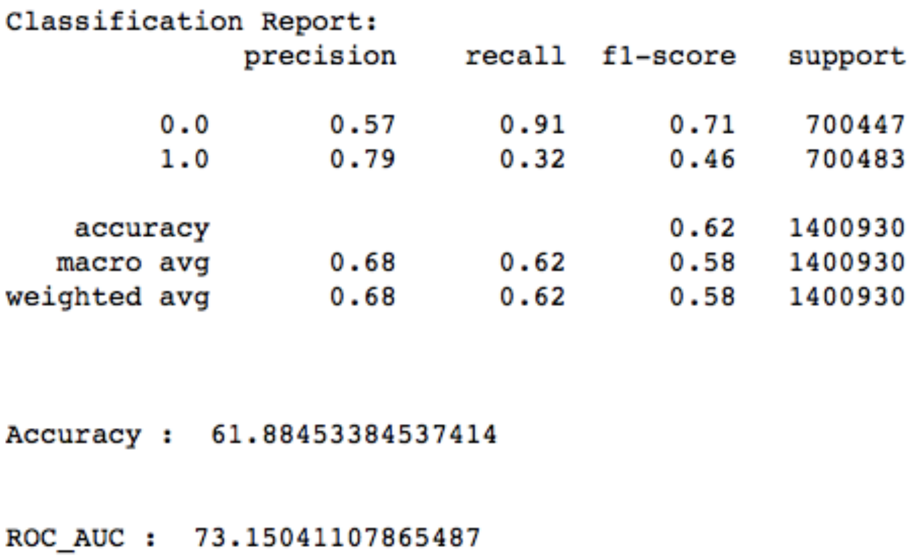
2. Naïve Bayes model (before/after oversampling)
   Before oversampling, the results of naïve bayes is like the following:



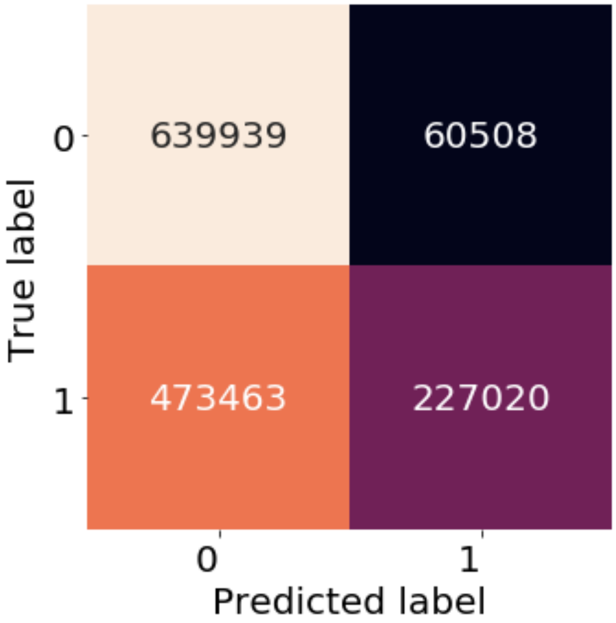
```
Classification Report:
              precision    recall  f1-score   support

         0.0       0.91      0.97      0.94   2293622
         1.0       0.35      0.17      0.23    248777

    accuracy                           0.89   2542399
   macro avg       0.63      0.57      0.58   2542399
weighted avg       0.86      0.89      0.87   2542399


Accuracy :  88.75707550231101


ROC_AUC :  73.52204563059284
```



We can see from the classification report, the accuracy score is 88.75 which is very high. It seems to be a good result. However, when we dig deeper, we found that the recall value of 0 is much bigger than that of 1. We also notice from the confusion matrix that the type 2 error is very large. It is not a good model as we expected. The reason is the that there are 2293622 of level 0 but only 24877 of level 1 in the target. Therefore, we used oversampling to modify our data and run this model again. Here are the result of evaluation of naïve bayes model after oversampling data.



```
Classification Report:
              precision    recall  f1-score   support

         0.0       0.57      0.91      0.71    700447
         1.0       0.79      0.32      0.46    700483

    accuracy                           0.62   1400930
   macro avg       0.68      0.62      0.58   1400930
weighted avg       0.68      0.62      0.58   1400930


Accuracy :  61.88453384537414


ROC_AUC :  73.15041107865487
```



After oversampling, the frequency of 0 and that of 1 are basically the same. However, the accuracy value decreased to 61.88 and the type 2 error still very large. Therefore, naïve bayes may not suitable for this case. So we decided not use this model.

3. GUI

Finally, the window is divided into three parts, including EDA, features and models. By clicking the EDA, we can see all the plots. By clicking features, we can see all the features and the first 15 records. By clicking the model, we can get the confusion matrix and classification report of the three models.

4. Summary and Conclusions

1. Our goal of this project is to predict whether a user will buy a product in the next order. First, we get familiar with the data through exploratory data analysis. Through the preliminary analysis of the data, we have a general understanding of the data set, thus raising our main question which is what product will the customer buy in their next orders. In this part, I learned how to deal with missing value and outliers. I am very familiar with how to merge data and group data through the practicing.
2. We tried several models, but also found our data set has some limitation. To improve that, we decided to use oversampling to balance the target and finally got a relative good result. However, the naïve bayes model is not ideal for this case, so we just not explain the result of this model. In this part, I learned how to run models and also improve models by using oversampling method.
3. The data set we choose is very large, so it really took time to run the outputs of models. We find that SVM may not be a good classifier for big data because it would take too long to see the result. But I also learned how to adjust the parameters to change the ratio of the training set to the test set, thus speeding things up.

5. Calculation

（550-220） / （550+140） = 50.7%

6. References

https://www.kaggle.com/serigne/instacart-simple-data-exploration
https://www.kaggle.com/philippsp/exploratory-analysis-instacart