CS506 Midterm - Kaggle Competition

My name	Xi Chen
BU id	U23766637

1. Introduction

This problem based on a bank-transfer dataset, the goal is to predict whether the record is fraud.

The dataset include 23 features as belows.

Id	trans_date_trans_time	cc_num	merchant	category	amt	first	last	gender	street	city	state	zip	lat	long	city_pop	job	dob	trans_num	unix_time	merch_lat	merch_long	is_fraud

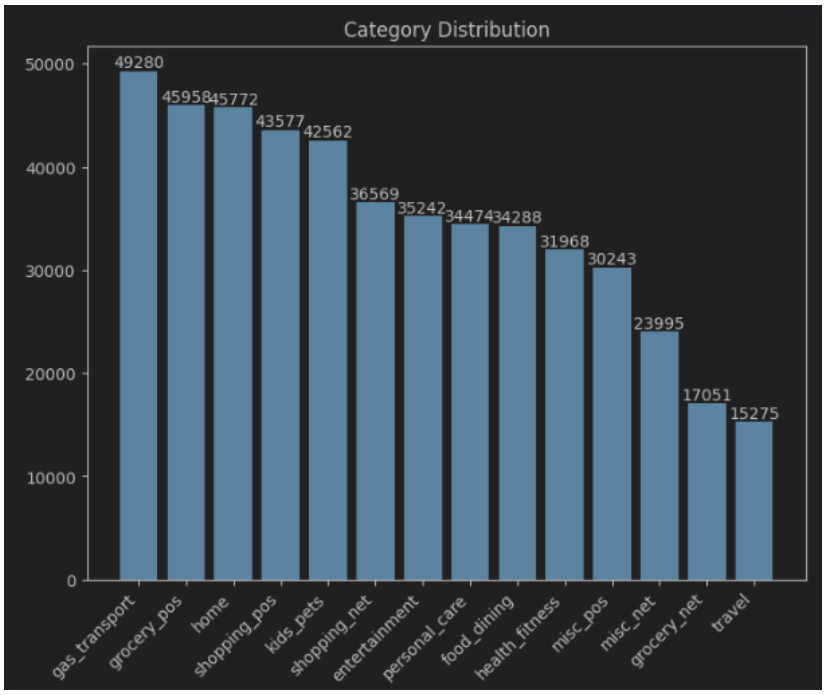
There are a total of 555719 training data and 69465 test data.

Since deep learning is not allowed, I mainly used KNN method to segment the data set well, reaching an F1-score of 94.

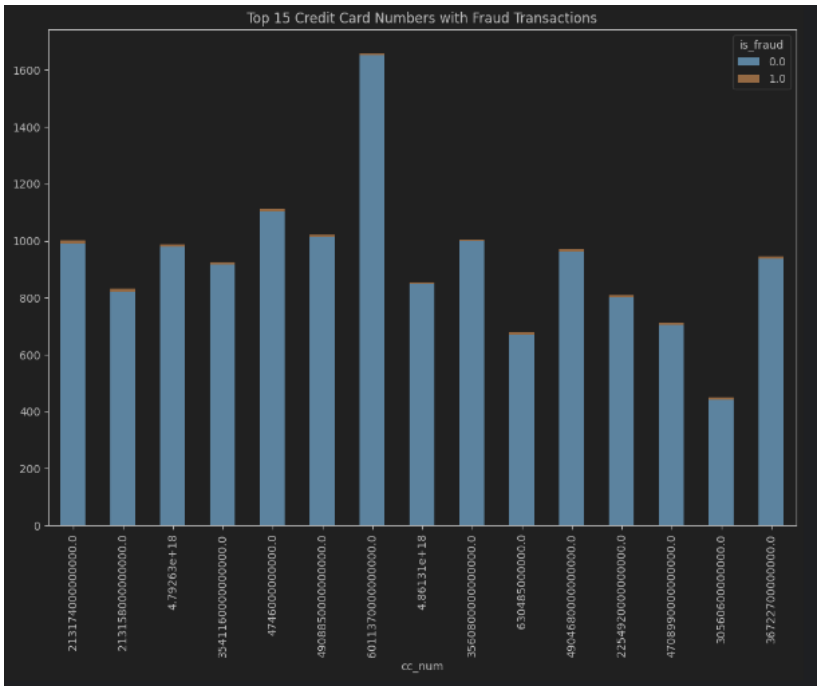
2. Data Analyse

Initially, I calculate the data distribution as belows.

There are approximately 0.4% data points are "Fraud", which means that the data distribution is very uneven. Therefore, using probability models like Naive Bayes are seem to difficult to reach a satisfying result. Then, I tried to split the data and look for some relavent. Here are some examples.



This is the distrubution of transfer type, I tried to find some relationship between fraud record and category, but no obvious different.



	cc_num	date	non_fraud_count	fraud_count
106715	4.599290e+15	2020-10-25	0.0	14.0
87415	3.577790e+15	2020-08-02	0.0	11.0
78201	3.545110e+15	2020-12-08	0.0	11.0
89783	3.588000e+15	2020-12-02	0.0	10.0
127662	6.538440e+15	2020-11-20	0.0	10.0
17327	4.586260e+12	2020-09-14	0.0	10.0
106927	4.607070e+15	2020-12-12	0.0	10.0
68035	3.501940e+15	2020-10-15	0.0	10.0
31143	3.054650e+13	2020-11-25	0.0	10.0

Then, I tried to find the relationship between card_number , trans_date and fraud. As the figure shown, there is a great deal of similarity in the performance of the same bank card on the same day.

3. implementation

According to the data-analyse above, I tried to use KNN(k-nearest-neighbor) as the main method.

The main feature is cc_num, the second feature is trans_date, and other are other factor.

To unify the data, I transfer trans_date into 8-digits integer and drops out {Hours:Minute}, and directly transfer cc_num into float, so that the weight of cc_num is larger than trans-date. I also tried to assign smaller values to other factors like categories, but they almost have no influence in my experiments.

```
merged_data['date'] = pd.to_datetime(merged_data['trans_date_trans_time'],
format="%d/%m/%Y %H:%M")

merged_data['date_str'] = merged_data['date'].dt.strftime('%Y%m%d')

merged_data['date_decimal'] = merged_data['date_str'].astype(int)
```

After that, setting the number of neighbors to 3 and run the KNN models.

```
X = Data_train[['date_decimal', 'cc_num']]
Y = Data_train['is_fraud']
knn = KNeighborsClassifier(n_neighbors=3)
knn.fit(X,Y)
```

The number of neighbors cannot be too large, I've tried to set k = 3 but the f1-score is worse. That's probably because most of the data is not_fraud, so the number of neighbor are easy to influence by the noises.

4. Result

When setting k = 3, my code can reach a optimal result, where the f1-score is 0.94395.