Lab 1 Data Pre-processing

Duruo Li

April 15, 2023

1 Introduction

This report is about data pre-processing. I divide the report into two parts: Part 1. Data Cleaning and Part 2. Data Exploration and Findings.

2 Data Cleaning

2.1 Format: missing and duplicated

Questions: 1, 3

1. Duplicated: first, I read the dataset description, and find out the seemingly similar variables; second, I compare the "suspiciously-duplicated" columns, and then drop one of the duplicated columns.
2. Entirely missing: if we can find extra information sources, we can use them to impute the columns, otherwise, drop them.
3. Partly missing: I first summarise the missing percentage of each variable. To handle this problem, I decide to drop rows with at least one missing value, since the percentage of data being dropped is only 1.232%, which is not very influential, i.e., the remaining amount of data is large enough. Moreover, if we are considering more specific questions w.r.t less variables, this percentage will be even smaller.

2.2 Value: outliers and special variables

Question: 2, 4, 5

1. Outliers: to detect outliers in numerical variables, I use boxplot to visualize the distribution of values, see Figure 1; and use Z-Score and LQR as metrics to do further detection. The number of outliers see table 1.

To handle outliers, since the pattern is reasonable, i.e., these numerical characteristics vary among different people, it's inappropriate to simply drop them. Thus, I will handle these outliers depending on certain problems. E.g., if we want to build a model concerning creditLimit for normal people (majority of people), it's practical and valid to drop the outliers, since those values are far beyond the range of normal credit limit; as for the other 3 variables, since the number of outliers are considerable, we might break them into two parts and build different models respectively.

As for potential impacts, there are both positive and negative ones.

- Disadvantages: distorted models, i.e., unmatched with the reality/ or at least with the situations we care about, e.g., credit limit for normal people; influential points, i.e., make the model unrobust, since the extreme values are rare and random, which aren't always the case, however, they are influential as training data in modeling.
- Advantages: provide unexpected insights. In this data, the outliers aren't so "rare", especially for "availableMoney" and "currentBalance", thus, they might reveal some certain truths underlying the usual situations, i.e., if we them into several different models, we might find out that those "outliers" show different patterns w.r.t fraud transactions.

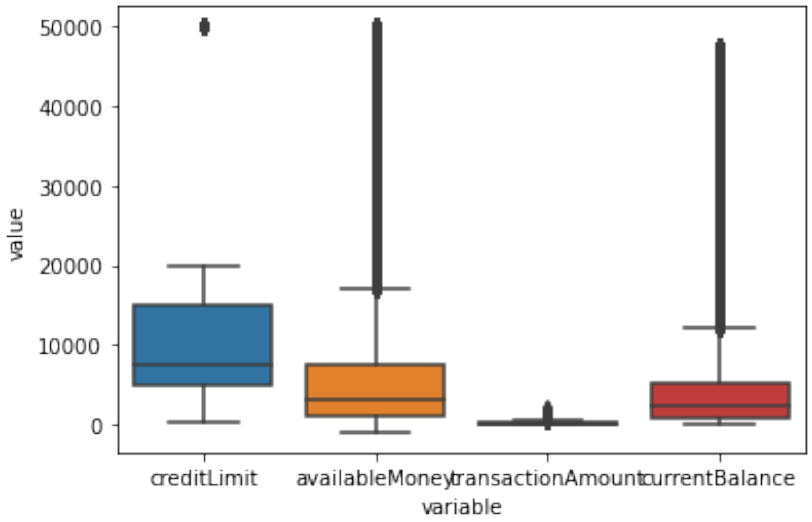

Figure 1
Numerical variables boxplots

Variable	Z-Score	LQR
creditLimit	48781	48781
availableMoney	25524	58834
transactionAmount	14933	40700
currentBalance	18183	69603

Table 1: Number of outliers (Z-Score and LQR)

2. Special variable-time variable: there are 2 potential issues regarding time variables and I deal with them as follows:

- Missing value: I summarise the number of missing values. There is no missing value for each of the time variables
- Inconsistent format: I separate the "transactionDateTime" into "transactionDate" and "transactionTime", and convert "currentExpDate" to standard date format. In this way, all the time variables will be consistent, i.e., become comparable. There are many benefits, e.g., the plots w.r.t different time variables are easier to read; transaction time can be used in calculations as I do in later parts

3. Special variable-sensitive variables: first, I convert cardCVV, enteredCVV, cardLast4Digits to string type, since they couldn't be regarded as integer. Second, since they contain sensitive private information, in real life situations, they shouldn't be analyzed carelessly. Although comparing cardCVV with enteredCVV might be a practical way to detect fraud transaction, we should ensure that these parts of information won't be leaked, or they should be processed beforehand, e.g., encrypt them, such as mapping the real codes to virtual ones
(BTW, can data analysts really get these kinds of data during work?)

3 Data Exploration & Findings

This part includes exploration on special variables, e.g., time variables, informative variable 'transactionAmount', and outcome variable 'isFraud'; and the relationships between 'isFraud' and categorical variables, numeric variables and sensitive variables. Moreover, it also includes a construction of a new characteristic "multi-swipe transaction" based on existing variables.

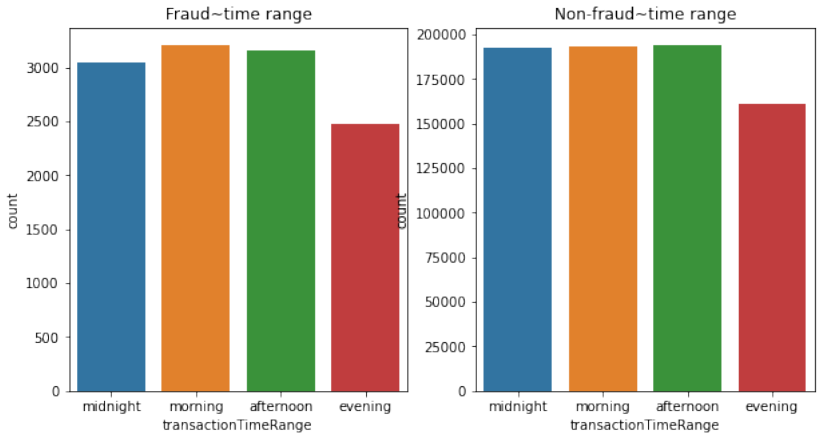

Figure 2
Time range patterns

Card and Entered	Fraud Rate
matched	0.015672
unmatched	0.028938

Table 2: compareCVV and isFraud

3.1 Time variables

Question: 4

I extract a new feature of 'transactionTimeRange' by splitting "transcationTime" into rougher time range, i.e., midnight(hour 0-6), morning(hour 6-12), afternoon(hour 12-18), evening(hour 18-24). And it will be interesting to figure out at which range, fraud transactions are more likely to happen. But from Figure 2, the patterns of fraudulent and non-fraudulent don't differ much. But in later exploration (multi-swipe transactions), this feature presents interesting differences.

3.2 Relationship: fraud & sensitive variables

Question: 6

1. cardCVV, enteredCVV

To find the relationship, I combine cardCVV and enteredCVV together and construct a new variable 'compareCVV' which indicates whether these 2 variables matched or not. Then I would like to know is there a relationship between 'compareCVV' and 'isFraud', see Table 2, unmatched CVV transactions have a fraud rate almost twice as high as matched CVV transactions. To visualize this relationship, I use barcharts, see Figure 3

2. cardLast4Digits

After simple explorations, I find that there are considerable number of values in cardLast4Digits aren't 4-digit strings. Thus, I combine construct a new variable 'is4Digit' which indicates whether cardLast4Digit is a 4-digit string or not.
Then I also compare the fraud rate according to 'is4Digit', see Table 3. Since the fraud rates in two groups are similar. Thus, it seems that whether the cardLast4Digits is a regular value, i.e., a 4-digit string, isn't related to fraud transactions.
Moreover, there are certain cardLastDigits which occur in fraud transactions much more often than others, e.g., '593' is involved in fraud 32946 times, '2194' is involved 10867 times, which imply that the corresponding cards might be especially used for crime.

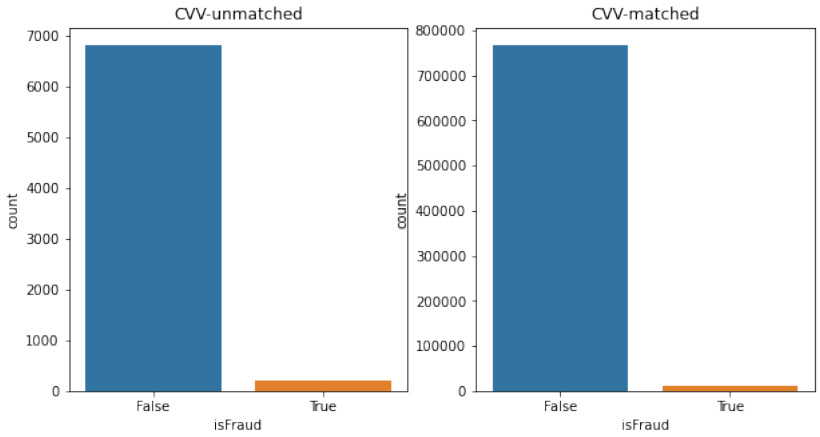

Figure 3
cardCVV and enteredCVV isFraud

cardLast4Digit	Fraud Rate
4-digit	0.015699
Not 4-digit	0.016351

Table 3: is4Digit and isFraud

3.3 Transaction Amount

Question: 7

I first use histogram combined with kernel density plot and boxplot to visualize the distribution of 'transactionAmount', see Figure 4. It's obvious that the distribution has a long tail, i.e., although most of transaction amounts are below 500, there exist a number of transactions amount are much higher than 500. Since there are many details in the "long tail", see Figure 5, which may be neglected in the full-data model, we could split the data into several parts, i.e., set certain cut points, e.g., 0 and 500, and explore the patterns (e.g., relationships with fraud) respectively.
Moreover, the distribution has many zeros (2.83%), thus, it might follow ZIP (zero inflated poisson) distribution.

3.4 Relationship: fraud & categorical variables

Quesion: 8

For each categorical variables, I create a bar chart to display the fraud rate for each category, see Figure 6 and Figure 7, fill NA with 'None' and drop NA respectively. As for patterns observed:

1. Fill NA

If we re-encode the missing value as 'None', we can see that the fraud rate of category 'None' is the highest among all categories, except for merchantCategory, which doesn't have missing values.
It implies that we should take a further look into the missing values, since they are strongly correlated with fraud. E.g., what has caused the missing; did it happen during the transaction or afterwards, i.e., does 'missing' itself implies fraud possibility (e.g., criminals hide their tracks on purpose). If it's true, we might add "categorical variables exist missing or not" as an indicator predictor into the model.
As for merchantCategory, some certain categories weren't involved in fraud transactions, e.g., food delivery, fuel, etc., thus, we might exclude them from the models if we use one-hot encoding.

2. Drop NA

If we drop the missing values, the fraud rate of different categories still differ, not so much, though, which means that there might be correlative relationships between these categorical variables and outcome variable, isFraud. Thus, we might take these categorical variables as predictors.

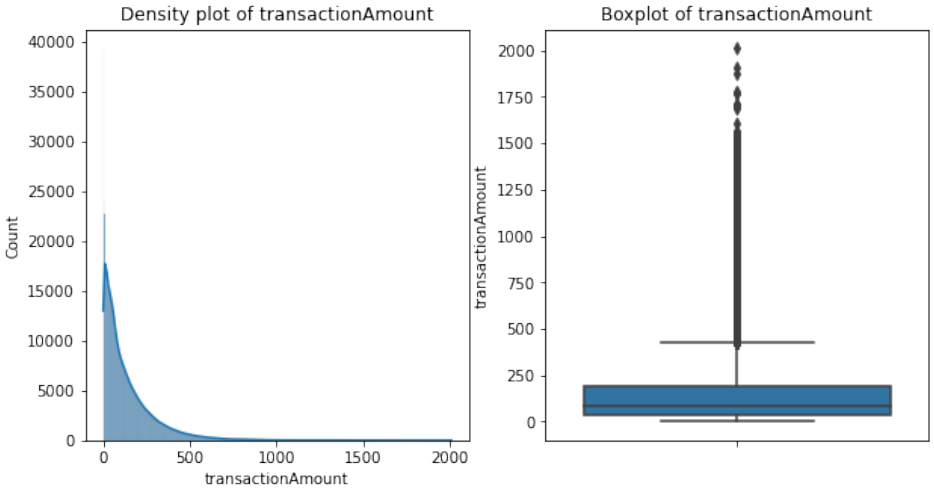

Figure 4
Distribution of transaction amount

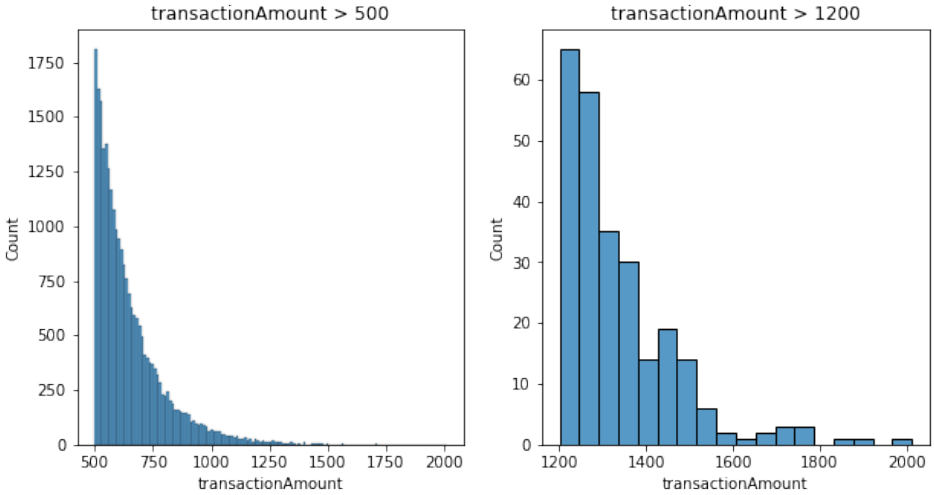

Figure 5
Distribution of transaction amount (split details)

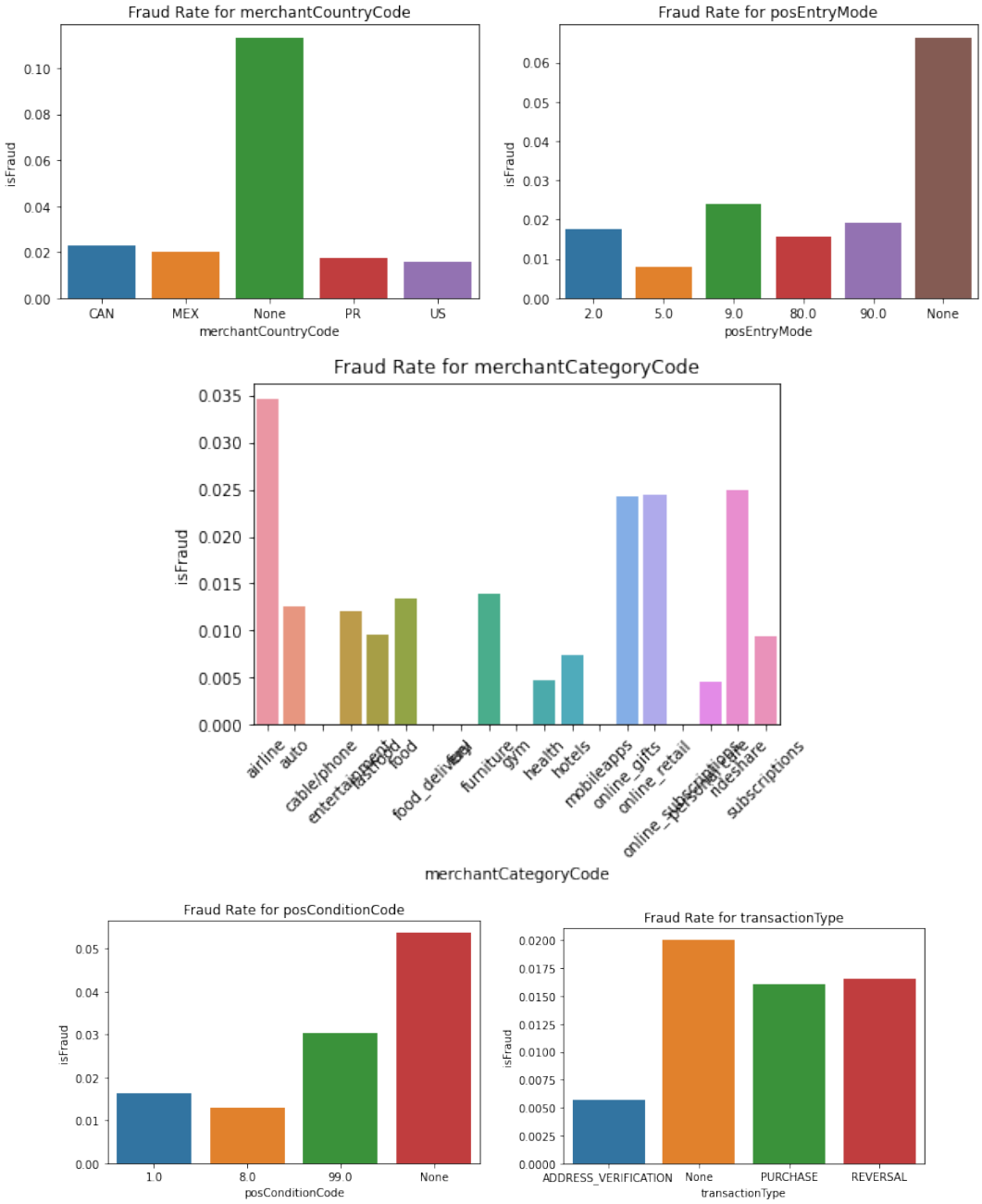

Figure 6: Fraud rate for categorical variables (fill na)

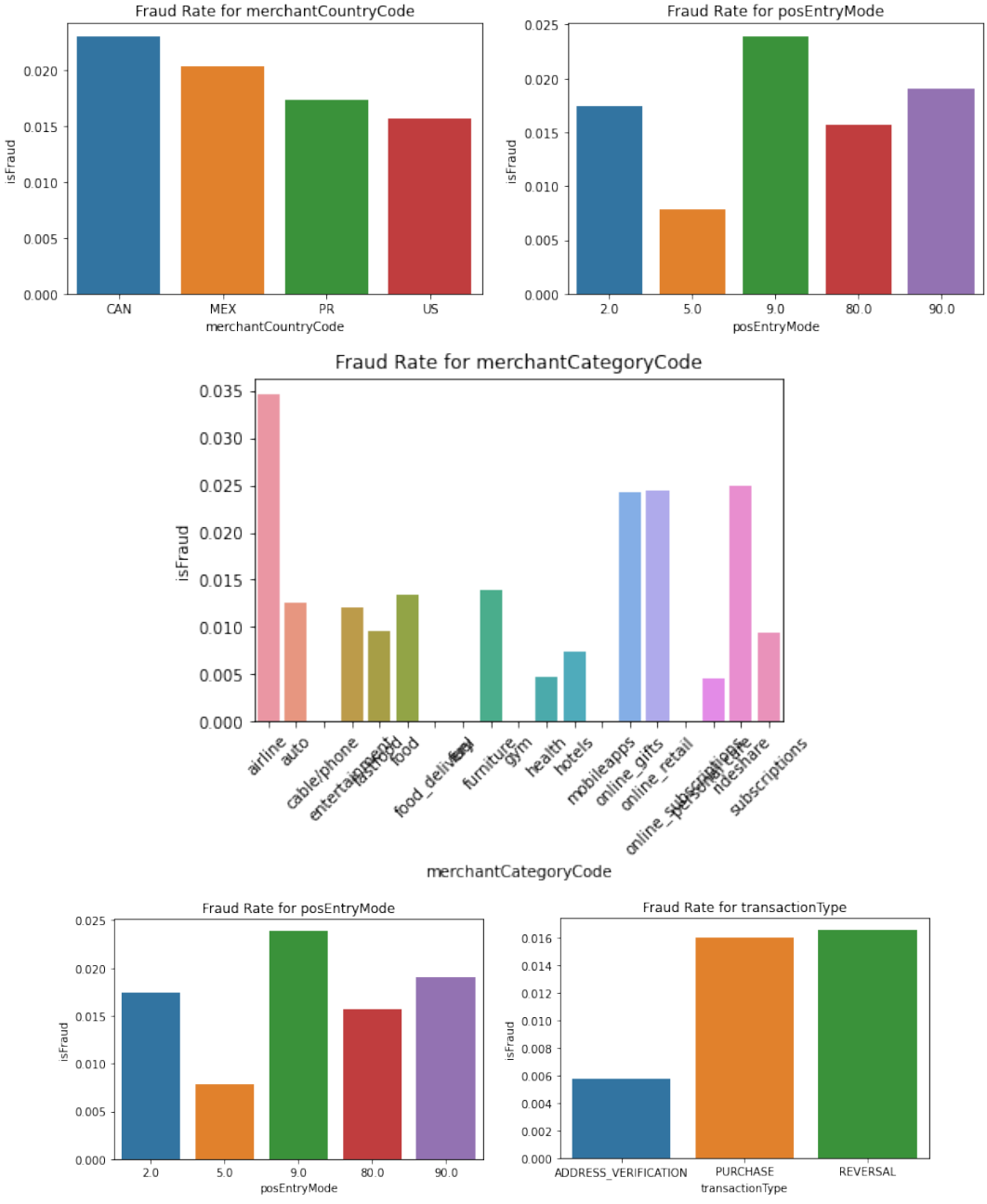

Figure 7: Fraud rate for categorical variables (drop na)

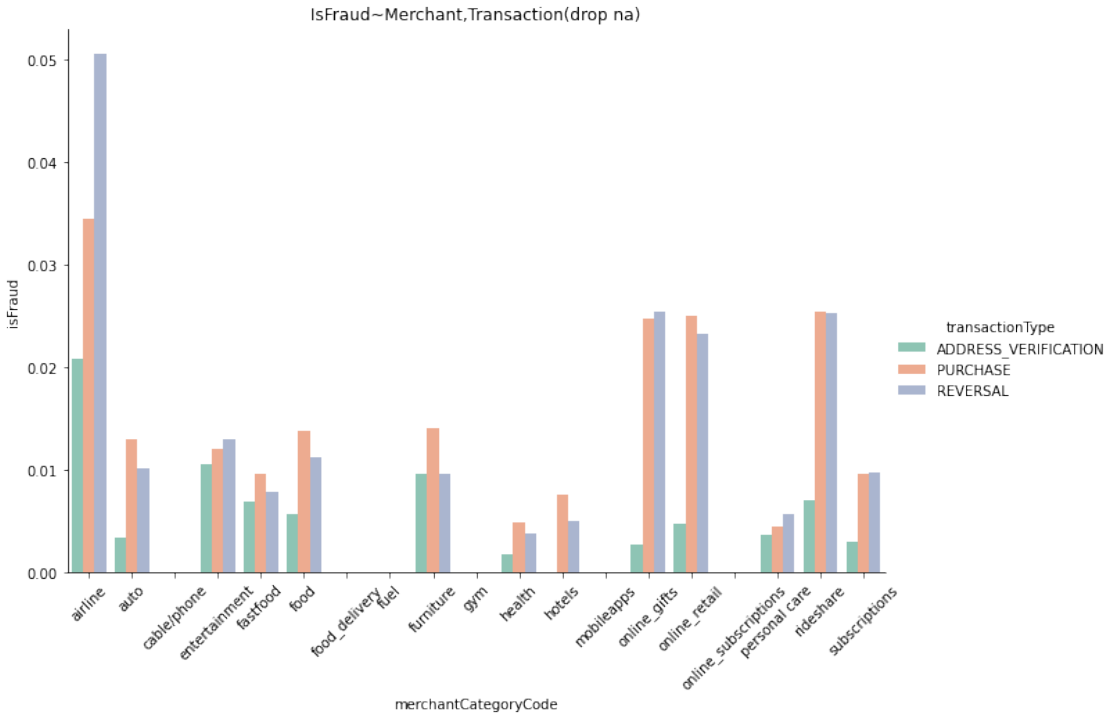

Figure 8
Fraud rate of transactionType conditioned on merchantCategoryCode (drop na)

3.5 Relationship: fraud & transactionType, merchantCategoryCode

Question: 9

To explore the relationship between isFraud and transactionType conditioned on merchantCategoryCode, I create two grouped bar chart, fill na, see Figure 9 and drop na, see Figure 8. To interpret:

1. Missing transactionType only occurs in certain merchant category groups, e.g., airline, entertainment, fastfood, online gifts, online retail, and ride share.

2. For the merchant category group which contains missing transaction type values, the fraud rate of this subgroup, i.e., with missing transaction type values, can be very high—much higher than that in other subgroups. Especially for the airline-missing subgroup, the fraud rate has been higher than 10

3. If we drop missing values: generally, the fraud rate of each group follows the whole pattern, i.e., address verification is least likely involved in fraud, while purchase and reversal constitute the majority of fraud, and the rates are similar.
To be specific, airline group is special, since its reversal and purchase subgroups have the highest and second-highest fraud rate. In previous plot, we already know airline group has the highest fraud rate, but after splitting it into different transaction type subgroups, the difference has become more significant—airline-reversal subgroup's fraud rate is around 5%, which is much higher than any other subgroups.

3.6 Relationship: fraud & numerical variables

Question: 10

I use conditional probability density plots to explore the relationships between numerical variables, i.e., 'creditLimit', 'availableMoney', 'transactionAmount', 'currentBalance', and the target variable, 'isFraud'. See Figure 10.

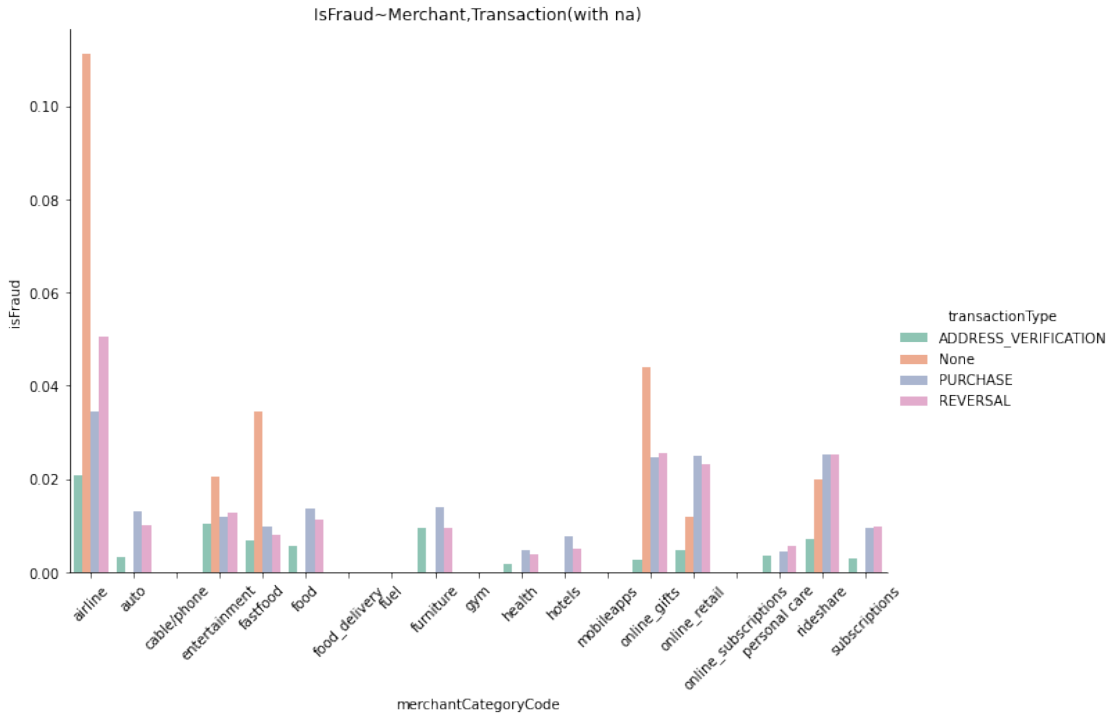

Figure 9
Fraud rate of transactionType conditioned on merchantCategoryCode (fill na)

As for trends and patterns observed from the distributions:

1. The overall distribution patterns are similar in fraud group and non-fraud group, e.g., location of modes, which implies that more transactions, more fraud.

2. As for specific variables:

- Credit limit: there are 3 modes, the first two are around 5000 and 15000, and the third is around 50000. It implies that fraud transactions happened "fairly" among different credit limit groups, i.e., both normal limit and extremely high limit credit cards suffered from fraud.
- Available money: it's interesting to see that despite the actual limit of cards, the current available credits are mostly around 0 to 1000. Fraud group distribution has one single peak around 0, while for non-fraud group, there still exist 3 "¿10000" peaks, not so significant as in credit limit distribution, though.

3.7 Define and explore multi-swipe transactions

Question: 11

The conditions for defining "multi-swipe transactions" is:

- same accountNumber
- same merchantName
- transactionDateTime difference less than 5 minutes
- transactionAmount difference less than 0.01 dollar (eliminate the interference of floating-point precision)

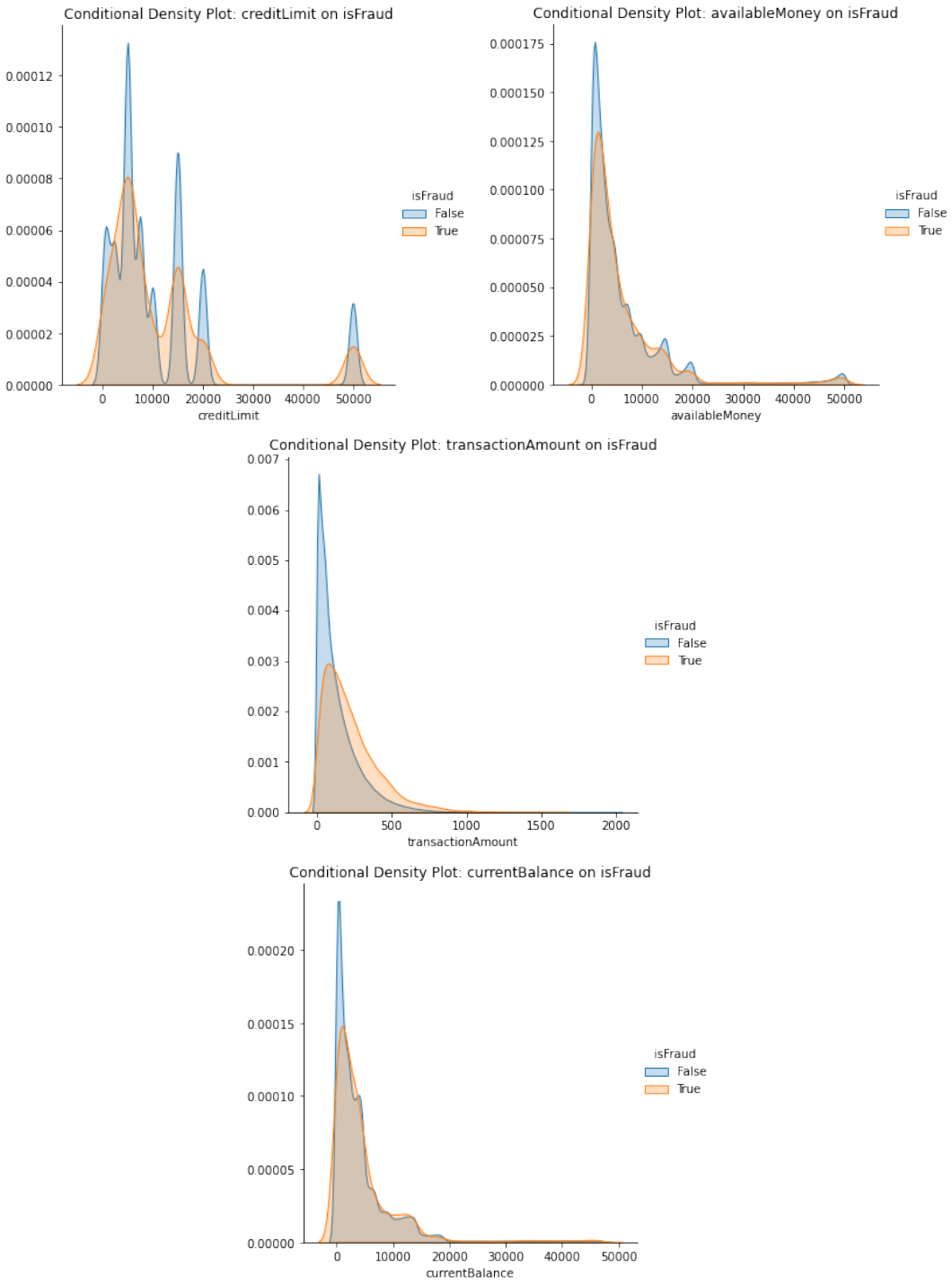

Figure 10: Conditional distribution of numerical variables based on fraud

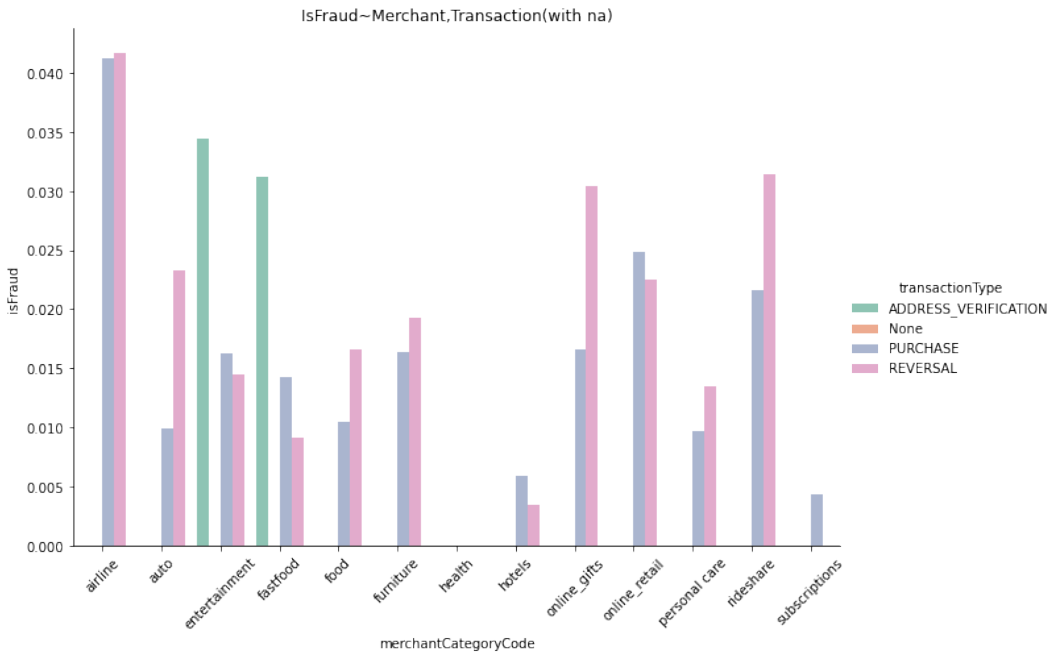

Figure 11
Fraud rate condition on merchantCategory-transactionType (multi-swipe subgroup)

After defining the "multi-swipe transactions", we can estimate the percentage of multi-swipe transactions, which is 1.70%; and the percentage of the total dollar amount for these transactions, which is 1.79%.

I find something interesting w.r.t. this subgroup:

- The fraud rate of multi-swipe transactions is 1.74%, which isn't much higher than the overall fraud rate, 1.58%.
- When considering the fraud rate comparisons w.r.t. categorical variables, except for posEntryMode, the fraud rate of missing category is 0 in other categorical variables, which is quite different from overall transactions, i.e., the fraud rate of "missing" category is the highest among all categories.
- When considering fraud rate of nested groups of merchantCategory-transactionType, the pattern is also quite different from overall pattern, see Figure 11. Address verification fraud only occurs in entertainment and fast food purchase, and the fraud rate of address verification is the highest, much higher than other transaction types. It's abnormal, since address verification fraud is rare compared with other types according to overall summary.
- The average amount of fraud multi-swipe transactions, 226.83, is greater than that of real multi-swipe transactions, 142.86.
- The fraudulent and non-fraudulent transaction time range patterns are interesting in the multi-swipe subgroup, see Figure 12. Real multi-swipe transactions happen almost equally in the midnight, morning, afternoon and evening, while fraud multi-swipe transactions are more likely to happen in the midnight and in the morning.

3.8 Oucome variable: imbalance problem

Quesions: 12, 13

Figure 12
Transaction time range patterns (multi-swipe subgroup)

By calculation, we find that the overall fraudulent transaction percentage is only 1.579%, while non-fraudulent rate is 98.421%. Thus, there exist an "imbalance" problem in 'isFraud' variable.
I think there are two potential negative impacts on building predictive models:

- The failure of commonly used metrics: the false prediction of a minority of the data won't influence the overall model performance. However, what we care most about is the accuracy of predicting fraudulent transactions, thus, some commonly used metrics, e.g., the prediction accuracy, aren't reliable anymore, i.e., even the metrics show that the model performs perfectly, it doesn't.

- Biased model: if this data isn't a good representation of the real world, i.e., the real world fraud rate, or in other words, the fraud rate under certain situations which we are interested in, is much higher or even lower, the model won't perform well on future data/real-world data.

To eliminate the potential effects, I would like to try two different methods:

- Method 1: Re-define the metric for model performance
  Re-weight the loss function, i.e., we give a higher weight to the fraudulent-transaction data points in the loss function. Or we can simply use the Area Under the Precision-Recall Curve (AUPRC) metric, which also focuses on the positive class.
  Effects: Model performs better in detecting fraudulent transactions: since the "fraud" part of the training data plays an important role in minimizing the loss function, i.e., the loss function "cares" a lot about the prediction accuracy of the fraudulent-transaction. Moreover, AUPRC also balances the precision and recall, i.e., unlikely to detect non-fraudulent to be fraudulent and be able to detect true-fraudulent targets.

- Method 2: Build a two-step model
  Since most of the transactions are non-fraudulent, we might first build a model to "detect" obviously non-fraudulent objects, i.e., the first prediction is about whether the transaction is non-fraudulent or unsure.
  The remaining data will be more balanced than before. Thus, we can use it to build a second-step model based on traditional metrics, which now focuses on detecting fraudulent transactions.
  Effects: Performs better in predicting fraud, and we don't need to worry about whether reweighting or oversampling will change the characteristics of data, which might lead to bias due to its inconsistency with the real world.

p.s. what does it really mean to be imbalance? In the real world, the percentage of fraudulent transactions is indeed very low, so if we try to oversample them, will it cause any potential problem? Since oversampling might result in biased training data in terms of whether the training data is a good representative of the real world.

4 Conclusion

Data pre-processing is time and efforts consuming, however, it can provide us with a lot of information about the data and is necessary for building a well-performed model.

References