Credit Card Financial Fraud Analytics

Wang Weishi 3036033699 Mak Lam 3036036500 Gao Peiyan 2013518343
Xie Ning 3036033924 Li Dongheng 3036034875

1. Case Background Profiles

1.1 background

Digital payments are evolving, but so are cybercriminals. According to SMALL bank's report, more than 80 thousand credit card digital payments records are reported of being stolen in a monthly basis, a concerning statistic shows that it is still very common both for Card-Present and Card-not-Present type of payments. To protect the rights and interests of customers and the bank itself, we must build an effective model for detection of fraud in both Card-Present and Card-not-Present scenarios.

We are the Financial Fraud Analysis team in SMALL bank. SMALL bank recently released its new credit card to its customers. However, we have received a lot of fraudulent transaction reports from our customers a month later. To protect SMALL1. bank's reputation and their customer's assets, we are assigned to develop a fraud detection model to prevent similar fraud cases happening in the future.

We retrieve the transaction records of these credit cards from the database of SMALL bank. After removing the confidential information of our customers, we get a table of eight columns that can be used to develop a fraud detection model.

1.2 Scope of Fraud Data Analytics

Our Scope of this credit card fraud data analytics is to detect frauds according to transaction records and evaluate the correlation between fraudulent behavior and provided transaction information. Moreover, we can use the developed model for credit card fraud detection to identify future fraudulent behaviors related to SMALL bank credit cards using corresponding transaction information in each transaction entry.

1.3 Fraud Scenario Identification

The committing person in this specific case is people with intent to defraud and the Entities related to this specific case are credit cards and leaked accounts tied to credit cards.

There are three major fraudulent actions that are addressed in the transaction records: card-present offline fraud, card-present online fraud, and card-not-present fraud.

The Card-Present offline fraud is related to the scenario where copied or stolen credit cards are used in offline transaction situations to conduct fraudulent action. One example of an offline transaction is making a payment when the POS machine system is set to offline mode by the merchant.

The Card-Present online fraud is related to the scenario where copied or stolen credit cards are used in online transaction situations to conduct fraudulent action. Examples of online transactions are making payments through Electronic Funds Transfer (EFT).

Card-not-Present fraud usually happens when the person committing the fraud acquires credit card numbers with personal information used in online orders or leaked user accounts tied to credit cards by attacking vulnerable merchant websites.

Figure 1.1 is the Permutation of Fraud Scenario for SMALL bank credit card fraud case:

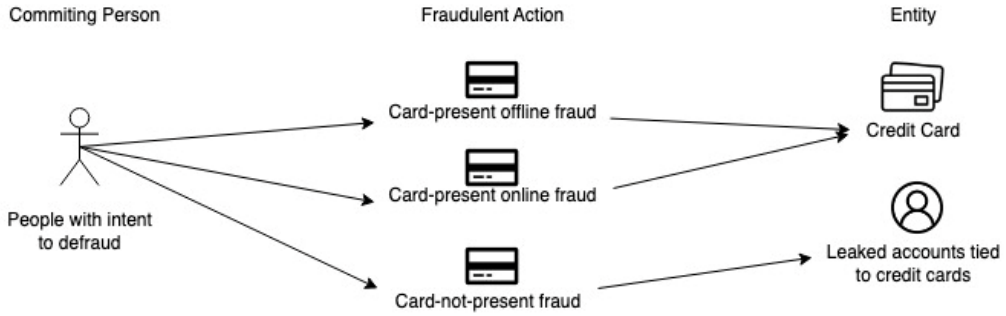


Figure 1.1 the Permutation of Fraud Scenario for small bank credit card fraud

1.4 Data Analytics Strategies

The Specific Identification Strategies that are used for our Credit Card Fraud project are identifying stolen or copied Credit Cards that are used in both online and offline transactions as well as leaked user accounts tied to Credit Cards that are used in online transactions. There is no internal control factor in our Credit Card Fraud project. In the dataset, we are looking for any fraud transaction using PIN numbers, any fraud transactions that are online orders, any fraud transactions that are from repeat retailers and fraud transactions that has high ratio of purchased price transactions to median purchase price. In the EDA analysis below, we are also going to check whether there's any anomalies or outliers in the dataset and process them if necessary.

2. EDA

```
> #print out the summary of each attribute
> summary(credit_card_data)
 distance_from_home  distance_from_last_transaction ratio_to_median_purchase_price repeat_retailer
 Min.   :    0.005   Min.   :    0.000              Min.   :  0.0044               Min.   :0.0000
 1st Qu.:    3.878   1st Qu.:    0.297              1st Qu.:  0.4757               1st Qu.:1.0000
 Median :    9.968   Median :    0.999              Median :  0.9977               Median :1.0000
 Mean   :   26.629   Mean   :    5.037              Mean   :  1.8242               Mean   :0.8815
 3rd Qu.:   25.744   3rd Qu.:    3.356              3rd Qu.:  2.0964               3rd Qu.:1.0000
 Max.   :10632.724   Max.   :11851.105              Max.   :267.8029               Max.   :1.0000
   used_chip      used_pin_number   online_order       fraud
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
 Median :0.0000   Median :0.0000   Median :1.0000   Median :0.0000
 Mean   :0.3504   Mean   :0.1006   Mean   :0.6506   Mean   :0.0874
 3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:0.0000
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
>
```

Figure 2.1

As shown in the Figure 2.1, there are 8 attributes and 1,000,000 observations in our dataset.

2.1 Outlier Analysis

```
> null_values_table <- sapply(credit_card_data, function(x) sum(is.na(x)))
> print("the number of null values of each attribute")
[1] "the number of null values of each attribute"
> #The number of null values in each attribute
> print(null_values_table)
          distance_from_home distance_from_last_transaction ratio_to_median_purchase_price
                           0                              0                               0
             repeat_retailer                      used_chip                 used_pin_number
                           0                              0                               0
               online_order                          fraud
                           0                              0
>
```

Figure 2.2

There are no missing values in our dataset, and no negative value in every attribute. Although there are some extreme cases shown in Table 2.1, we can't verify that they are mistakes or just some special cases, so we can't remove them from the dataset.

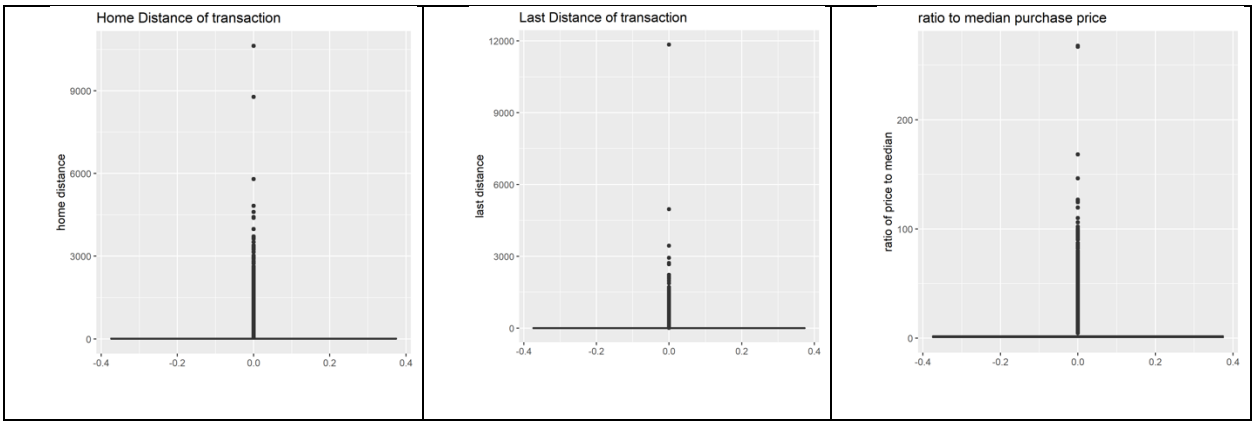


Table 2.1

2.2 Univariate Analysis

As shown in Figure 2.3, this is an unbalanced dataset as normal cases takes up to more than 90 percent of all cases.

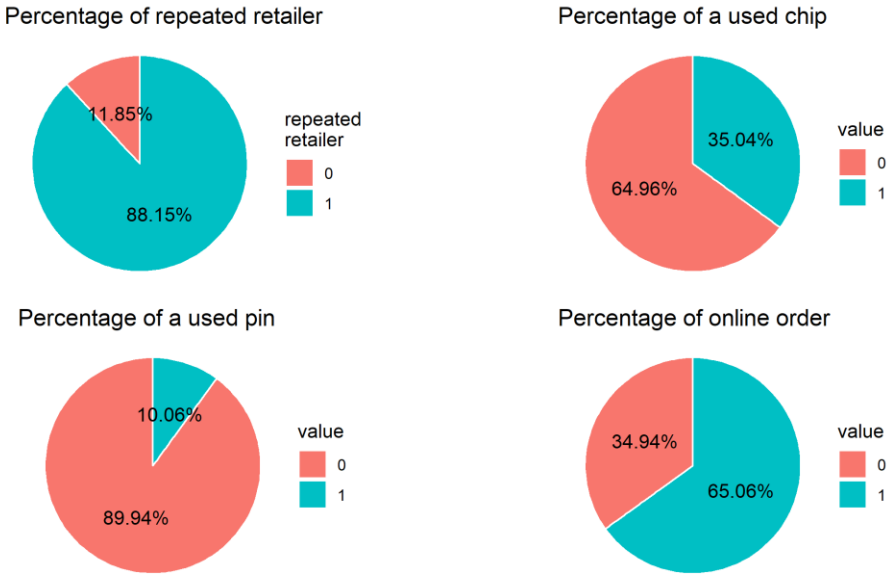


Figure 2.4

In the Figure 2.4, we can see that the majority transactions are online orders and literally people do not use pin number when they do a payment. Moreover, it indicates that repeated retailer cases are the majority in the dataset.

2.3 Bivariate analysis

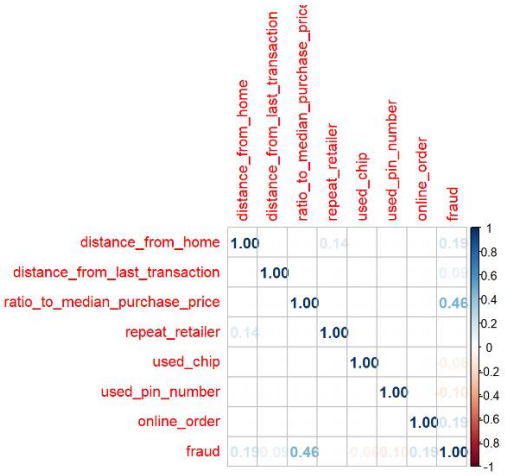


Figure 2.5

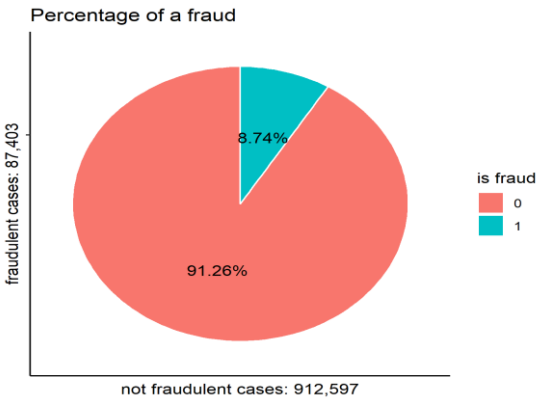


Figure 2.3 Unbalanced dataset

We can see that the relationship between attribute ratio to median purchase price and fraud cases are much higher than others. There are also some weak relations between the fraud attribute and attribute distance_from_home and online_order.

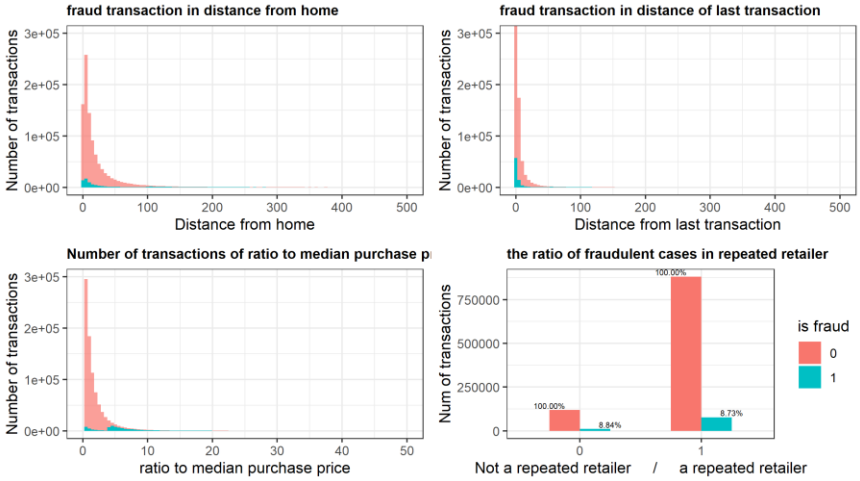


Figure 2.6

The Figure 2.6 plotted the number of transactions of fraudulent cases and all cases in dataset. The red area indicates the number of all transactions in its corresponding attribute. The blue area shows the fraud cases in that attribute.

Most of the distance_from_home and distance_of_last_transaction have a small value. However, there also some fraud cases distribute widely in large values.

The distribution of ratio to median purchase in shown in bottom left graph. We can also divide the fraud cases into two parts. The part with small ratio has fewer cases than the higher ratio part. The fraud percentage is increases when the ratio to median purchase price goes up. The fraud percentage of repeated retailer cases are almost distributed evenly, which means this attribute is not relevant to fraud attribute while the others more less has some relation with fraud.

3. Data Analytics

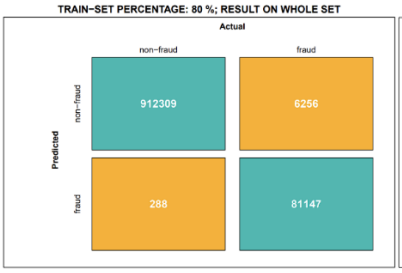
For this part, we will deploy various fraud detection models to predict whether a given transaction is fraud or non-fraud with the 7 attributes (all the attributes provided by the dataset minus the Fraud label) as inputs. We will comment on the strengths & weaknesses of each classifier we use and evaluate their suitableness to our dataset based on the prediction performance. We will also make inferences on each variable's importance and give detailed graphical descriptions on how they play the role in deciding the outcome.

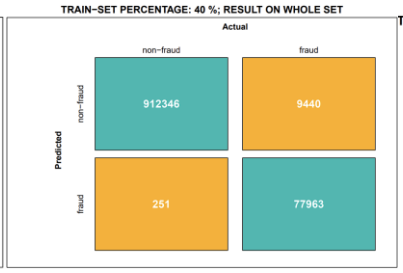
3.1. SVM

The first model we consider using is Support vector machine model (SVM). It has the merit of being able to capture the non-linear structure of the dataset by kernel transformation, as well as being resilient to outliers and correlated features, which are particularly preferable under this case as the EDA showed there're certain extreme observations existing in those continuous attributes, and the distance from home and whether the retailer is a repeated tend to be related to one another.

Before building the SVM we rescale each attribute to have a mean 0 and standard deviation 1. Besides, we noticed that the time required to build the SVM model is proportional to the cost parameter C, hence we chose a small value of C (0.01) due to time complexity concerns. We use the gaussian kernel to do the transformation and the gamma is set to 0.5 by cross validation.

We build the SVM under two scenarios: the first one is to use the whole dataset, corresponding to the Figure3.1 and Figure 3.2 below:





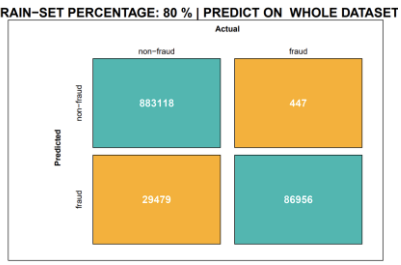


Figure 3.1 ***Figure 3.2*** ***Figure 3.3***

In Figure 3.1: Precision:0.996 Recall:0.928 Accuracy:0.993 F1:0.961

In Figure 3.2: Precision:0.997 Recall:0.892 Accuracy:0.99 F1:0.941

In Figure 3.3: Precison:0.747 Recall:0.995 Accuracy:0.97 F1:0.853

And the second is to use the down-sampled dataset after balancing the fraud and non-fraud cases, corresponding to the figure below:

For each scenario we consider two different training set proportions: 0.8 vs 0.4, in order to see whether the model performance is sensitive to the number of trainings instances.

From the former two figures, we can see that the SVM performs well on the whole dataset with training-set percentage of 80%, whereas the recall still seems to have space for improvement. As the training percentage is tuned down to 40%, its performance also gets slightly worse. From the above figure, we can see the trade-off between label balancing and down-sampling: Since the proportion of fraud cases is elevated, the model does achieve a higher

recall rate meaning it becomes more capable of capturing those fraud transactions. On the other hand, as the effective sample size gets smaller after down-sampling, the overall performance gets poorer. There is a sharp drop in precision value, meaning that the model now generates too many false positive predictions.

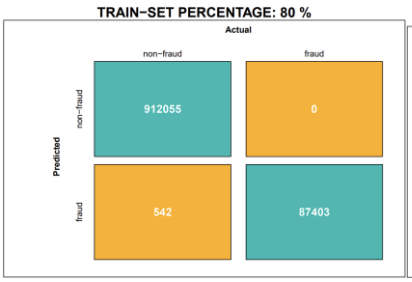
A significant drawback of SVM, which has also been mentioned above, is the heavy computational cost, which limits the scopes of fine-tuning hence makes it hard for us the find the optimal value of hyperparameters. Besides, SVM assumes that the two transaction classes can be divided by a hyperplane, which heavily depends on the structure of the dataset and may not hold true.

3.2 KNN

In our case, the dataset is of large-scale with only a few attributes, and if the structure tends to be clustered instead of being separable, the k-nearest-Neighbor (KNN) is supposed to be a more suitable approach.

To build KNN, we also do the rescaling first. We consider balancing to be a must for KNN to achieve desired performance, so we only build our model on the down-sampled dataset. By repeatedly try all acceptable k values and check the recall rates, we find the best k as 5 for this dataset, and the figures below shows the results of KNN evaluated on the test set:

The KNN model performs almost perfectly on the test set with only 13 cases misclassified. Its performance drops down a little bit while evaluating on the whole dataset, but still obtain full recall and all the other indicators close to 1. Even if we drag down the training set proportion to 40%, the performance still maintains at roughly the same quality except only a few more false positives, which indicates that KNN can learn the task quite well even with a small percentage of training instances.



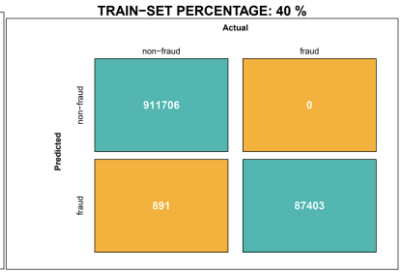


Figure 3.4 ***Figure 3.5*** ***Figure 3.6***

In Figure 3.4: Precision: 0.999 Recall: 1 Accuracy:1 F1:1

In Figure 3.5: Precison:0.994 Recall:1 Accuracy:0.999 F1:0.997

In Figure 3.6: Precision: 0.994 Recall:1 Accuracy:0.999 F1:0.997

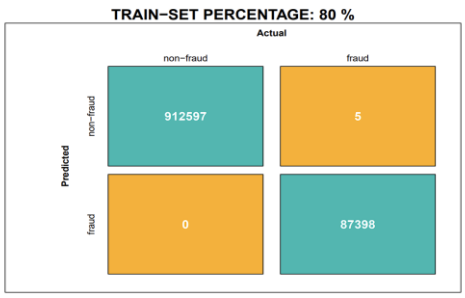
The goal of detecting fraud cases has been achieved, we then need to consider detailed attributes analytics and some further situations like getting new records containing new features.

3.3 Tree-based Models

As we know, fraudulent will never stop even though we have finished a good job at fraud detection, and there will always be new fraud techniques created as fraudulent prevention methods developing. In order to extract features from the new fraud records rapidly, we need to

deploy some models that don't rely on the size of the dataset. Thus, one widely used tree-based algorithm, random forest, is good to use.

According to the characteristics of random forest, we don't need to do data pre-processing and even don't need to do many configurations to the model. To validate our idea, we still trained this model on different proportions of the dataset, and the results are amazing as well.



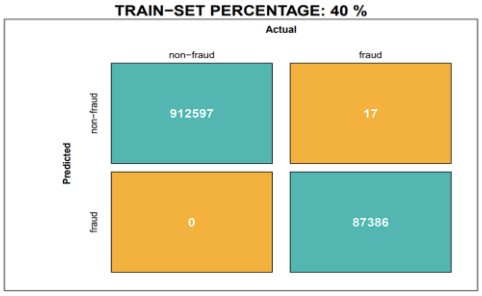


Figure 3.7 ***Figure 3.8***

In Figure 3.7: Precision: 0.999 Recall: 1 Accuracy:1 F1:1

In Figure 3.8: Precison:0.994 Recall:1 Accuracy:0.999 F1:0.997

Although we cannot take the place of KNN model, the advantage of fast training speed of random forest makes it be able to be used as an auxiliary model to help us improve the prediction effect. The other purpose of using random forest is to check the attributes importance to verify the information we get from EDA process and to make further analysis.

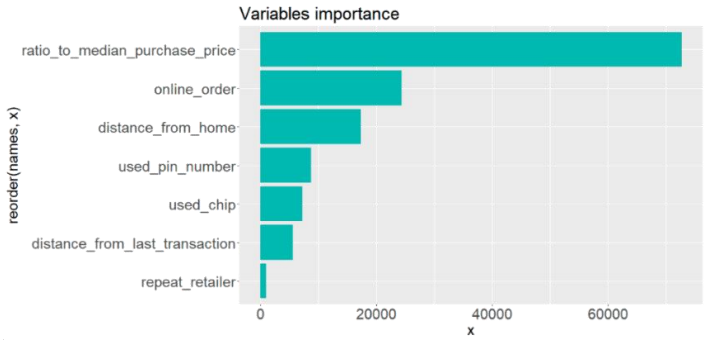


Figure 3.9

Another model we used to help us analyze the results is decision tree, which is highly interpretable. We can simplify the whole model to specific rules by deleting the nodes leading to non-fraud records and constructing paths from root to fraud nodes. The five rules we get are shown below:

Summarize the five rules to fraud cases, they should be merchant abuse, stealing, deception, leaked account and leaked pin number. Then we can use these rules to give our customers some advices and to add some restrictions to our system to decrease the fraud rate.

ratio_to_median_purchase_price	online_order	distance_from_home	used_pin_number	used_chip	distance_from_last_transaction
≥4	0	<1.9	X	0	X
≥4	0	≥100	X	X	X
≥4	1	X	0	X	X
<4	1	≥100	0	0	X
<4	1	<100	X	0	≥50

Figure 3.10

4. Non-data analytic element

4.1 red flags

The red flags in the dataset for this analysis are (in order of importance):

- The ratio of the current transaction amount to the median is too high
- The transaction type is online shopping
- The location of the transaction is too far from the user's address
- No payment password was used during the transaction
- Both transactions occurred at the same merchant i.e. repeated retailer

4.2 non-data analytic elements

a) Firstly, we should check each link within the bank to confirm whether any link has the risk of information leakage. In information management, we should clarify the responsibilities of each department and person so that they can restrict each other and avoid the disclosure of information.

b) Next, we should improve the information management system to ensure the reliability of the technology and leave traces of every step of information modification so that there is evidence to trace when problems occur in the future.

c) Improve the professionalism and ethics of merchants to prevent them from being able to obtain other people's information from daily purchases and commit credit card theft.

4.3 suggestions

a) For the credit card service in the bank, strictly check the applicant's information when opening an account to ensure that it is true and valid, verify the applicant’s credit record to avoid fraudulent behavior such as malicious overdraft.

b) detect the amount of each transaction, warn of larger transaction amount, and increase the audit of such transactions.

c) Since the fraud rate of online payment is higher than that of other transaction methods, it is necessary to improve the security verification of online payment by requiring the trader to enter a password and perform information identification for identity verification when making online payment. And for the payer, raise awareness of the safety precautions is very important. When they are paying the bill, they need to confirm the safety of the website and the information of the retailer.

d) Add the function of detecting the location of the transaction to warn of transactions with a large difference between the location of the transaction and the address of the account holder. Strengthen the verification of the identity of the transaction to prevent credit card theft abroad.

e) For transactions that can be authorized to merchants for no-pin payment debit, the amount of debit that can occur in the future can be limited to reduce the degree of damage when fraud occurs.

f) Cardholders who discover that their credit cards have been lost or stolen should immediately report the loss to their banks to prevent further losses.