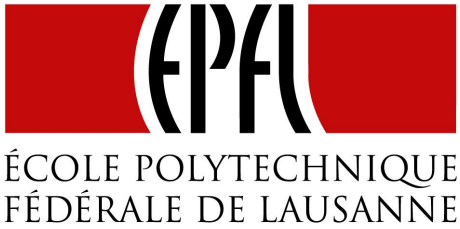


DATA SCIENCE IN PRACTICE
(MGT-415)

Fraud Analysis and Detection in KiWi Transactions

Authors:
Luis Emmanuel MEDINA RÍOS
Xueyan ZHAO
Dan CHAI
Jingwen WANG

Supervisor:
Ph.D. Christopher BRUFFAERTS

5th April 2018

Contents

1 Introduction

1.1 Background

KiWi is a swiss startup founded in December 2013. The headquarter of KiWi is based in Lausanne. As a FinTech company, it aims to build a digital platform and provide financial services for micro-merchants. The main serving objects of KiWi are those rapidly developing foreign countries. As the first implementation of KiWi, it started to penetrate in Mexican market from 2015. KiWi has made good performance in microfinance industry since it was launched into Mexico. According to statistics, the total transaction amounts based on KiWi devices & app is shown to increase exponentially within three years, reaching almost 200,000 at the end of 2017. KiWi's objective is to serve more than 1 million merchants in Mexico by 2021 [1].

As for the stakeholders, we will focus on investors, employees, collaborators and merchants respectively. KiWi Mexico is a subsidiary of eBOP SA, Switzerland. The investors are Foundation for Technological Innovation as well as winners of Swiss Fintech Venture Leaders 2017 [1]. There are eight full time equivalent employees currently in Mexico, working on business strategy and service maintenance. As a new entrance in Mexican market, KiWi largely relies on local bank Bankaool, which is the first branchless bank in Mexico and key financial partner of KiWi. Besides, KiWi also collaborates with some micro-finance corporates, like CAME and FINCA, who share KiWi's payment solution with their clients. The merchants that use KiWi payment method are basically retailers, saleswomen and door-to-door promoters, running small-sized business, for instance, female hairdresser, boutique owner. Their customers would rather use card to transfer money than pay with cash and get small change. In this case, KiWi payment is an ideal choice. By connecting KiWi device to a merchant's smartphone and inserting credit or debit card into card reader, customer can easily close the deal. Moreover, the launch of micro-credit service as a new strategy could bring more customers since the tailored pre-payment way help them reach purchase goals with small amount of money [2]. According to an interview of clothes saleswoman, who benefits from KiWi micro-credit products, she doesn't even feel like paying the loan back by herself, her customers help her finish it instead.

Figure 1: KiWi device attached to a smartphone

1.2 Problem statement in business

As a financial services solution provider, KiWi inevitably faces with fraud sometimes. A thorough and detailed description of fraud provided by Van Vlasselaer is that "Fraud is an uncommon, well-considered, imperceptibly concealed, time-evolving and often carefully organized crime which appears in many types of forms." Fraud exists in all kinds of social sectors and cannot be eliminated as long as there are rules constraining people. Especially, financial industry is a high-incidence area of fraud since most fraudsters' purpose is to gain personal profitability.

So far, KiWi Mexico has completed more than 200 thousand transactions. Among them, around 260 transactions have been labeled as fraud, taking negligible percentage of the entity. Nevertheless, we believe that there are many undetected frauds hiding behind the mountain of total transaction. There are different types of fraud existing in KIWI. In terms of object, the fraudsters could be merchants, customers, or even fraud gangs. In terms of fraud types, it can be debit card or credit card fraud, consecutive transactions at different places for a short period, and money laundering as well. The rules with which KiWi measures a transaction is fraud or not is nowhere to know since it is confidential information of the company. As far as we know from KiWi managers, they currently detect fraud manually by raising a flag to suspicious transaction in the dataset. The transaction money will be hold-up in the bank of KiWi until the card-holder comes to complain the money loss and claims it. Things get complicated when KiWi could not detect real fraud. If a card-holder complains to the bank that he has unauthorized charges, the bank of card-holder will contact with KiWi bank. If KiWi has approved this transaction, KiWi has to be responsible for the economic loss of card-holder. Typically, the scam amount is enormous. Thus, fraud can be detrimental to KiWi's profit as well as reputation, and should be treated seriously.

1.3 Problem statement in analysis

As mentioned in the last section, the main purpose of our project is to leverage the historical data to detect cases of fraud through the analytical method. There are several challenges related to the detection of fraud, like low percentage of fraudsters, limited operational efficiency and velocity. To reach the goal and bring the value, there are several steps to follow in this throughout the project. Firstly, we need to specify the problems the stakeholders want to solve and their requirements. Then, Data scientists, we in this project, needs to follow the requirements, translate the business problems into a data problem and perform the analysis with the dataset provided by the company. In usual cases, fraud experts are very helpful to analyze the behaviors or techniques of fraudsters.

In this project, we leverage the complete dataset which is merged by the dataset of each chargeback provided by the company, KiWi Bop, and the dataset of detected fraud cases in these transactions. The dataset contains four sections, including the information about the shop, the merchant, the transactions and the class of fraud. The whole dataset is composed of 169,981 transactions with 80 features. Among these

169,981 transactions, there are 260 cases judged as fraud. Our main strategy is to analyze the effect of the each feature, such as transaction time, on the fraud performance through the feature analysis and train the models to detect fraud through the machine learning approach. We determine to analyze the fraud performance in a binary classification. As the dataset is labeled, we will take several approaches of the supervised methods to detect our target variable, fraud.

As for the data management section, we prepare our dataset through two steps, data tables merging and data cleaning. In the part of dataset cleaning, we found that there are several columns with large amount of data with no values by visualization tools. After going back to and checking the original data source, we take several measures and treat the missing data seriously. Afterwards, through the detailed study on the dataset, we found that the relationship between shop and merchant is a one-to-one correspondence so that the data is filtered by omitting repeated information and reducing the columns. In addition, we also pay attention to transforming data, including data normalization as well as transforming several complicated features into easy-operated columns. For instance, the specific transaction time is classified into several time periods.

In the next section, data analysis & modeling, we divided it into two parts, exploratory data analysis and machine learning modeling. We analysis the importance of each features with the help of visualization tools and take advantage of some statistical methods. As mentioned before, the dataset is labeled so that we employed the supervised modeling methods for this classification problem. The main problem for the modeling part is to deal with the extremely imbalanced dataset, since fraud cases only comprised 0.1% of the total cases. Several measures have been already proved to be efficient to solve problems with the imbalanced dataset, from the following three dimensions, sampling methods, skew-insensitive classifiers and evaluation metrics[3]. Although over-sampling and under-sampling techniques are easy to operate, to achieve a balanced class distribution in our case where the dataset is really large and the fraud cases are rare, will make the dataset too large resulting in reduction of the efficiency of modeling or too small leading to the loss of credibility of the model. Therefore, considering time and cost, we select the solutions based on the skew-insensitive classifiers and effective evaluation methods. As for the skew-insensitive learners, decision trees are one of the good choices, which mitigate the need for sampling. Except decision trees, we also adopted the ensemble-based methods, which leverages the classification power and improves performance for the classification problems, like Random Forests, AdaBoost and VotingClassify.

Moreover, in the evaluation sector, to modify the assessment of performance of a classify, since the accuracy ceases to be an effective evaluation metric, the receiver operating characteristic (ROC) curve and F_1-Measure are employed to reflect the performance of a learner when rare class exists. In addition, in order to validate the stability of our models, cross-validation is also involved in the section.

1.4 Expectations

By going through all the parts described above, we expect to bring some values for company. First, we intend to come up with a data-driven approach to detect fraud to upgrade the existing manual detecting method. Therefore, we can help both company and its clients to reduce economic loss. At the same time, we want to improve the precision of fraud detection so that the false positive cases can be minimized. Otherwise, the company may face customer churn problem. With our model, we expect the company can not only retain loyalty of present clients, but also may have acquisition in new clients.

2 Data Analysis

2.1 Data Fetching and cleaning

Since we want to train models to predict whether a transaction could be a fraud or not, we need a lot of data. For this, and because KiWi is starting acquiring the philosophy of the data-driven decisions, we had to study the whole KiWi database to collect those fields that would help us with the analysis and at the end, with the help of the back-end software engineer we could extract around 48 columns containing the information about the shops (id, affiliation date, type of industry, etc.), the merchants (id, phone, CLABE number, email, etc.) and the transactions (id, date of transaction, amount, location, etc.). However, the resulted dataset was not enough, since we didn't have the information regarding the card numbers (masked pan), as well as the information regarding the KiWi device and the cellphone model, so for this we had to go deeper in the transaction level and extract, again with the help of the back-end software engineer, all the metadata (masked pan, kiwi device serial number, mobile device model, etc.) regarding the transactions. Finally, two datasets with raw data were collected: One with the information of the metadata of the transactions and the other with information regarding the shops.

In order to have only one dataset with all the information it is important to merge them based on a key that both datasets share: the transaction reference number. So after merging them, we had a big and complete dataset with information regarding more than 220,000 transactions. However, we have to take into account the fact that we had the information of all the transactions since the very beginning until February 20th, 2017 and with that, many of them were used to test. Therefore, we need to clean the dataset as much as possible, the steps that we followed were:

1. Filtering only those merchants with a device and that are enabled
2. Removing "Libertad Servicios Financieros" merchants, since they are considered special by KiWi
3. Keeping only the transactions with card, therefore no cash transactions are included.
4. Excluding those transactions that were used as test.
5. Separating and reformatting date and time so that the analysis could be easier.
6. Getting more information about the cards (with an external API) with the help of the BIN number: like the type: credit/debit, VISA/MasterCard or Country of issue.
7. Grouping the information regarding all the KiWi devices, mobile devices and CLABES that a shop has in all their transactions. Making sure that these were correct values (like verifying that the CLABE is 18 digits, or the KiWi devices serial numbers start either with 800 or CHB, etc.).

Once we had a cleaned and filtered dataset with all the information regarding the shops, merchants, transactions and metadata, we ended up with around 170,000 transactions and almost 4,000 registered shops.

The next step is to get the information regarding the fraud cases. For this, we referred to the person in charge of the retentions to get a list of the the fraud cases and retentions that they have made. An excel file was provided with all this information and we had to clean it, reformat it again (specially in the dates) and marked each of the transactions as fraud so that we can merge it with the cleaned dataset. We chose 3 keys to merge them:

- MIT branch (1), as this is the branch number that each shop has and that identifies the shop with MIT, which is the entity that deal with the payments and transactions.
- Date (2) and amount of money (3) per transaction.

Finally, after merging everything, we had the most complete dataset with all the information we need to start analyzing the data and modeling it.

2.2 Data Analysis

After cleaning and merging the three datasets (shops-merchants-transactions, metadata and frauds), we ended up with 169,955 transactions, 3,981 shops and 260 transactions marked as fraud. Before continuing with the analysis, we have to make sure some details are covered: (1) To have all the transactions amount in MXN (considering 1 USD is approx. 20 MXN) and (2) that all the cards number (masked pan) are in the same format (XXXXXXffffffXXXX).

Looking at the number of transactions per shop (no matter the transaction result: APPROVED, ERROR, DECLINED), we find that the maximum number is 1,971 and the minimum is only one. However, the latter doesn't tell us anything since we are considering all the shops, and there may be some new shops in the dataset and that's why they have only one transaction. To attack this, we need to filter the shops taking only those that have been registered for more than 30 days, have made more than 25 transactions and are active: in average a shop makes 128 transactions (standard deviation of 204), Since the mean is higher than the median, we can see that there are some shops that really have high amounts of transactions (more than 1,000) and therefore affect the latter. Thus, we stay with the median (58 transactions) as the central value of the transactions per shop. One thing to see is that the number of shops has decreased to an approx. 1/4 of the original number (1,158 of the non-new shops vs 3,981 shops).

Looking at the transactions marked as fraud, like we said before, we have 260 and 70 shops are involved in these frauds. With this, we want to say that either the shops are trying to commit fraud or the cardholders or both. We also see that all the transactions are in MXN, and that from the 260: 72% were approved, 17% declined and

11% showed an error and even more, 52% were transactions with credit card and 28% with debit card, the 20% left can be explained by the fact that the credit/debit card API couldn't find that information.

By looking at the descriptive statistics of the transaction amount for both the non-fraud transactions and the fraud transactions, we can see that the median (Like we explained above, and due to the high standard deviation, we will use the the median as the central value) is way higher in the fraud cases than in the normal (non-fraud cases): 5,400 MXN (frauds) vs 300 MXN (non-frauds), the results for the non-fraud transactions seem to be the same even if we filtered by active and non-new transactions. Let us recall the fact that the KiWi device is meant to be used and help micro-merchants (e.g. small restaurants, small grocery stores, door-to-door selling people, etc.) and therefore the amount of 300 MXN per transaction makes more sense.

We can also investigate these frauds in terms of the industry type of their corresponding shops. The following shows the industry type and the number of shops that have committed at least 1 transaction marked as fraud:

- Food and beverages (Alimentos y bebidas): 17
- Electronics and computers (Electrónicos y computación): 11
- Professional services (Servicios profesionales)
- General services (Servicios Generales): 4
- Others (Otros): 4
- Education, courses and workshops (Educación, cursos y talleres): 3
- Sports in general (Deportes en general): 3 Convinience stores (Tiendas de abarrotes): 3
- Art, photograph and art craft (Arte, fotos y artesanías): 2
- Health services (Servicios de salud): 2
- Clothing and accessories (Ropa y accessorios): 2
- Travel and tourism (Viajes y turismo): 2
- Beauty and personal care (Belleza y cuidado personal): 1
- Transportation (Transporte): 1
- Not provided (nan): 8

It is quite interesting that Food and beverages is an industry that most of the frauds have been committed in. Since one can expect that normally one can spend like 50-150 MXN per meal. Recall again that we are assuming that they are not fancy restaurants, otherwise they would have a bank terminal. Regarding the other shops, for some industry types (Transportation, Travel and tourism, health services, Education or electronics and computers) these high amounts might be more normal but still could be frauds.

Just as a comparison, we get the median of the transaction amount including only the industry type of "Food and beverages" and compare the fraud vs the non-fraud cases: As expected the median of the amount for the fraud cases is way higher (4,500 MXN) vs the non-fraud cases (140 MXN). Figure 2 shows the difference in amount of the normal transactions vs the fraud transactions regarding all the industry types that we saw previously, in which we can see that the amount of money for each transaction is high for all the industries except the "Art, photograph and art crafts" and "Health services" which is unusual since one (as said before) can expect high amount of money on these transactions.

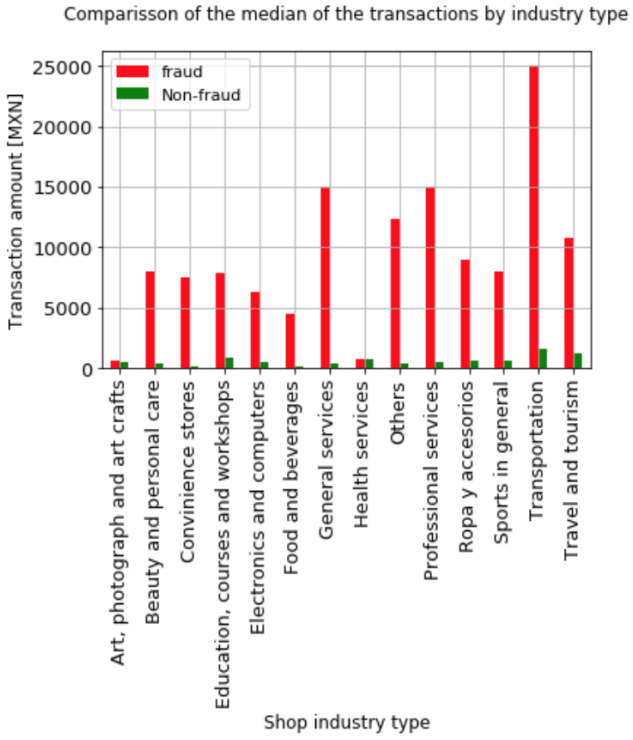


Figure 2: Comparison of the transaction amounts per shop industry type: fraud vs non-fraud cases.

2.2.1 Looking at the shops that have at least 1 transaction marked as fraud

Since none of the shops are fraud-free, and since we know that there are honest shops, we will have that for some shops with high number of transactions there are also some

of them that have been marked as frauds, which is kind of normal, since the most transactional activity you have, the more probability you have to face a fraud.

We can analyze then, those shops which all transactions or at least a ratio (fraud/non-fraud) is 0.15 (15%) of them are marked as frauds. We consider this percentage as we assume that if a shop is honest, and due to what we said before (None of the shops are fraud-free), their number of normal transactions has to be way higher than the fraud ones. If we find a shop with very few transactions and some of them are frauds it's because there is something wrong with the shop. To summarize this we can say that:

- A shop with high number of normal transactions and few number of frauds: It is more likely that a card holder is trying to commit fraud. We can say that the shop is honest.
- A shop with low number of transactions in general, from which very few are normal and there are some frauds: It is likely that either the shop or the card-holder are doing fraud. We don't want to say that the shop is committing fraud but it is just a possibility.
- A shop with all of its transactions marked as fraud: It is more likely that the shop is committing fraud with the kiwi device.

The results of this filtering show that from the 70 shops with at least 1 transaction marked as fraud: 32 have more than 85% of normal transactions, 30 have more than 15% of fraud transactions and 8 have only transactions marked as fraud. We can then go into a deeper analysis of these shops (with more than 15% of fraud transactions and with all transactions marked as fraud) just to analyze behaviors. By looking at the number of KiWi devices that they have used: from the 38 suspicious shops, 35 only have one device and 3 have two devices, and by looking at the number of mobile devices (cellphones devices or tables): 32 shops have only one device, 3 shops 2 devices, 2 shops 3 devices and 1 shop with 7 devices. Therefore, we can say that in terms of the number of KiWi devices, they look quite normal, 1 to 2 (let's assume that one of the devices was broken and therefore the merchant asked for a new one). In terms of the number of cellphones, we can assume that they were changing their cellphones. However, we can investigate a bit more about that shop that has used 6 mobile devices:

Shop id: 3880, it has 47 normal transactions vs 8 fraud transactions. Comparing the affiliation date and the date of the last transaction, the shop was active for 3 months, and the probability that the merchant changed his mobile device 6 times is quite low, the period is so short. However, the it only has 1 KiWi device, and the ratio of the fraud-non-fraud is 17%. Finally, we can just look at the card numbers, just to check if the cards have been used more than once in the same shop: There are 2 cards, that have been used in two transactions marked as fraud in the same shop: if we take a look at the first one, the card has made 4 transactions in the shop in a timespan of 2 minutes, only 1 was approved. By looking at the numbers, there can be two possibilities: the card holder is a fraudsters or he/she actually tried paying 500 MXN but the merchant

made a mistake and he charged 5,000 MXN instead. Even though looking at the time, the latter can be discarded (the first transaction was the Approved one) and after that the card continued making transactions. By looking at the second card, it tried making 3 transactions (2 of them marked as fraud and they have different amounts: 500 and 1,500 MXN) within a timespan of 3 minutes, but all of them didn't pass (the result was either declined or error).

2.2.2 Analyzing fraud by looking at the card information

From the 260 fraud transactions, we only have the information regarding the bank card in 245 transactions. If trying to get the number of cards used in these frauds, we observe that they are only 194, so that means that some cards have been used more than once to commit fraud, in the following list, we can see the cards and the number of transactions marked as fraud in which they were used (we only show the 10 with most fraud transactions, the rest have been used once or twice):

- 491283ffffff7625: 7
- 547046ffffff5579: 4
- 557910ffffff6828: 4
- 415231ffffff7730: 3
- 528843ffffff2376: 3
- 491871ffffff9151: 3
- 557905ffffff8069: 3
- 528851ffffff5399: 3
- 520021ffffff0495: 3
- 523227ffffff1359: 2

We can analyze those cards with 7 and 4 fraud transactions:

- **Bank card: 491283ffffff7625**

This card have a strange behavior, in a difference of 2 days, it has made 7 transactions and all of them marked as fraud, all of them have the same amount of money and very high, (15,000 MXN) the industry type is food and beverage, which we already found that the median of the transaction amount is 130 MXN. If we look at all the transactions that this card holder has made (including non-fraud transactions), they are only the 7 marked as fraud. Thus, this person has a very high probability to be fraudsters.

- **Bank card: 547046ffffff5579**

This is another case of possible fraud: 4 transactions marked as fraud, the same day, different shops, but the difference in time is approx. 5 minutes. Even though, they seem to be different shops, for any reason they are using the same dongle (kiwi device). Again, the transaction amount is almost the same in all transactions and very high. Since this is interesting, we can see more information about the other shops in which this card is involved. These shops (shop ids: 5570, 6013, 6015 and 6068) have joined around the same date from July, 26th to august, 22nd 2017, and two of them have registered with a difference in time of 7 minutes. One shop has a transaction marked as fraud after 6 minutes they joined. We don't have information regarding the location of the the transactions (they could disabled the GPS on the device not to be tracked). However, the geolocation of the shops indicates that they have registered different addresses. We even see that the models of the cellphone device is the same, so those four shops for sure are hiding something. We see another card number within the transactions of these shops: **557910ffffff6828**. By investigating more on this second card, we can say that again, the card was only used to commit fraud, same pattern: same day, difference of time of 5 minutes, high amount of money. Besides the 4 shops that we already found, two more are added, and the dongle device is still the same as well as the mobile phone model. Thing that make us think that these are part of the same person/group of people. We see that the affiliation date of these are from June, 20th to august, 18th 2017 and no information regarding the GPS. Then, we can say that again, there is extremely high probability that these shops are just being created to commit fraud:

- Shop id: 5570 Shop name: Torbellinos
- Shop id: 6013 Shop name: Apvil
- Shop id: 6015 Shop name: Epsi
- Shop id: 6068 Shop name: Piesort
- Shop id: 4914 Shop name: TokioEat
- Shop id: 5248 Shop name: Aldatel

Finally, by looking at all the transactions (including non-fraud) that these shops have made, we can notice that they have transactions not marked as fraud. However, most of them have not passed and follow same patterns: high amount of money, short difference in time between transactions, etc. Since we know that these shops might be fraudsters, we can collect all the card information that their transactions have to make another filter in which we even find more transactions for the shops "Torbellinos", "TokioEat", "Aldatel" and we find a new shop called "Formasv": Again high amount of money (8000) and exactly same kiwi device (same dongle serial number). Besides that, the other transactions are kind of normal, low amount of money (except one which is 1,545 MXN but it can be even possible). Therefore, there is something strange with the KiWi device with serial number **80030152600355000001**. Finally,

by looking at all the shops that have used this device, we found that there are 19 shops, and that the median amount of money per transaction is 6,000 MXN. We can also check that by the mobile phone model, they are the same:samsung SGH-I337M or samsung SM-J105B. Some of the names refer to the same person: For example, you have a shop called "Dentista Gpe" which merchant name is Guadalupe Romero. "Gpe" is a short way to write Guadalupe. Now regarding the shop "Piesort", the merchant name is Dentista Gpz. Another important shop is the Pamscupcakes shop, by the name one may think that they sell cupcakes, but by looking at the transaction amounts, we clearly see a possible case of fraud. We can conclude finally that they might be a group of people that are commiting fraud with the kiwi device, and that we discovered new possible fraud cases and a new possible modus operandi for fraudsters.

- **Bank card: 557910ffffff6828**

The card 557910ffffff6828 was also found in the analysis of the previous bank card.

2.2.3 Analyzing frauds by the country of the card

Since KiWi is only operating in Mexico, we can mark all the Mexican cards so that we can analyze the frauds in terms of the card's country of issue. The percentage both in the non-fraud dataset and the fraud dataset, are almost the same: 70-80% of Mexican cards vs 30-20% of foreign cards. So we cannot say anything about it. Let's also recall that Mexico is a destination for many foreigners, and not because of that they want to commit fraud in small shops. Regarding the amounts of these transactions, they are even higher than the normal frauds (from 7000 to 8000 MXN), some of them go beyond 10,000 MXN and even more than 40,000 MXN.

2.2.4 Analyzing frauds by the location of the transaction

Since the location of the shop doesn't mean that it is the actual location (this has been validated with the people of KiWi) as it may be the address of the merchant, we only take into account the location of the transactions. From the 260 transactions marked as fraud, we found that 52 doesn't have information about the geolocation (they might be hiding something). The next important thing is to check whether these fraud transactions were made in the same place or not. In order to do to this, we have to compute the distance between the coordinates, only taking into account those transactions in which the location accuracy has a value at most 40 (so we can make sure that it is approximately the actual place in which the transaction was done). The results we get tell that most of the transactions marked as fraud belongs to shops which transaction locations is not the same (the shops are moving) and even more, when looking at these moving shops, they are also moving when they make normal (non-fraud) transactions. For the case of the established (non-moving) shops, we notice the same tendency as moving shops, non-fraud transactions are done in the same place and fraud transactions are done in the same place. The latter doesn't tell that both the fraud and non-fraud transactions for non-moving shops have to be at the same place, and after investigating more about this, we see that slightly less than the half

of the non-moving shops makes their transactions for a possible fraud in a different location than they are used to do normal transactions. Which can lead to think that those merchants have to do with the fraud and not only the card holder.

3 Fraud Detection

This chapter illustrates how we used machine learning approach to solve fraud detection problem with a complete pipeline, including data preprocessing, feature engineering, supervised learning in an imbalanced dataset and model evaluation.

3.1 Data preprocessing

It is common to have missing values and redundant attributes in our real world dataset. As most machine learning models are unable to circumvent missing and non-numeric values to accomplish prediction, missing values should be seriously handled. Besides, there could be redundant features spread within all columns which serve no value for our prediction purpose. Including them does not help and even jeopardizes stability of our model, leading to a garbage in and garbage out end. Thus first step in data preprocessing is essential.

3.1.1 Missing data handling

To uncover missing patterns in data, visualization tools have greater power to tell us the story of nullity in a more effective way. Missingno is a Python package for visualizing missing data integrated with missing pattern investigation and basic statistical calculation.[4] After dropping duplicated entries in data, there are 169955 remaining transaction samples in 84 columns with both attributes and target variable. Figure 3 shows how missing values are distributed from both sample-wise and column-wise perspective. The sample with least nullity has 63 columns with valid information, while the sample at top nullity are only 19 columns complete. Given a bunch of white areas which represent missing values, we find that some columns in the middle are with high nullity. Due to the high dimensionality of dataset, we are unable to recog-

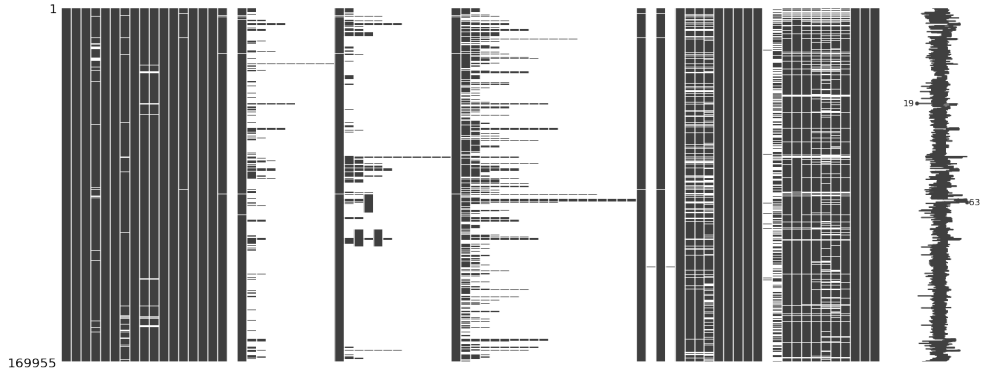


Figure 3: Nullity matrix of data

nize which attributes these columns represent in plot without annotations. By setting a threshold of 50000, we filter out columns with severe nullity, as shown in Figure 4. Referring to the meaning of most attributes below, we would say our data integrity is not affected as they only involve the on-working or wrong serial number and mobile

model name of KiWi devices a shop owns, CLABE number of a merchant and so on. The reason why there are so many columns derived is that a shop could have more

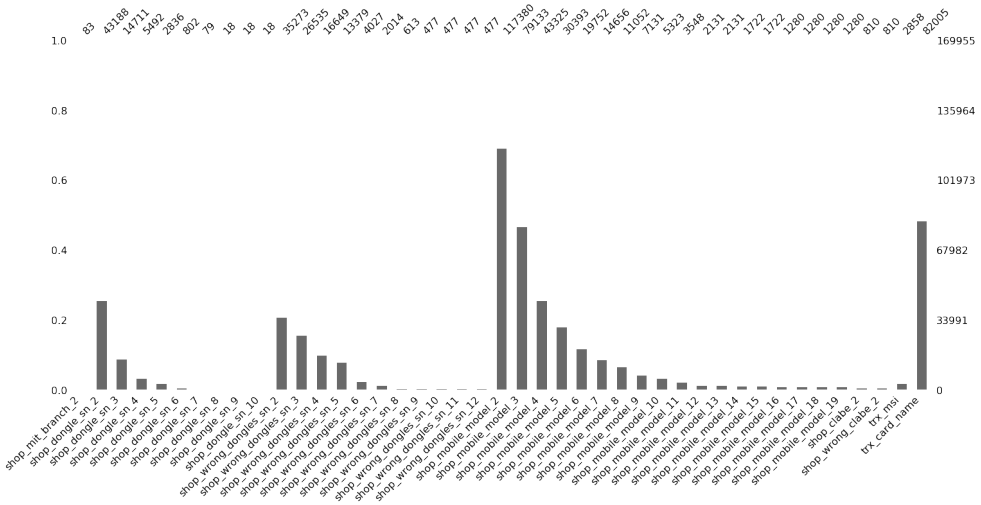


Figure 4: Columns with high nullity

than one devices. Obviously, they have no benefit to our detection process. We could transfer them into new features like the number of devices and CABLE number a shop possesses. Note that trx_msi also undergoes a high nullity, which means only 2858 transactions were differed in 3,6,9 or 12 months. So we can fill NaN with 0 which make sense for non-differed transactions. trx_card_name as a credential information is sure to be discarded.

As for the other attributes with slight nullity, if a value is numerically consecutive, like latitude, longitude and location accuracy of transaction corresponding to the shop location, we imputed them with median value. And for categorically discrete variables, like unknown industry type, Internet and transaction result we can give an extra value like unknown for later encoding which will be specifically combined with the following procedures in feature engineering.

3.1.2 Data filtering

In our data, we not only have shop information but also merchant information. What we are interested in is that if one shop corresponds to one merchant registered in Kiwi company for coherence inspection. Attributes like shop id, shop name, zip code, merchant id and so on are repetitive and valueless attributes. By investigating into matches of shop and merchant, we conclude that in our data, there is only one-to-one relationship. Then most shop attributes and all merchant information are omitted and only shop attributes with actual meanings like affiliation time, industrial type, shop status and so on are remained.

3.2 Feature engineering

Feature engineering is a core part in machine learning, and can be interpreted as input X's designing. Model performance can depend on the quality of feature engineering a lot. Feature engineering mainly consist of feature derivation, feature abstraction and feature selection. Here we are not about to talk about feature selection as we will embed it in our models.

3.2.1 Feature derivation

Feature derivation means creating new features based on combination of existing features. At transaction level, we only have transaction date, time and also location accuracy regarding each card identified by masked digits of the card. Grouped by each card after filtering out obviously wrong masked pan numbers like 0000000000, time interval and location accuracy difference between this transaction and last one is derived as new features. If the first transaction of a card, we regard its time difference as a distinguishably large number compared to other time intervals in seconds. For each transaction, first added time and last modification time, we could derive the life cycle time of a transaction. Contrary to the time when transaction happened, we are more inclined to transforming it into time period in a day, considering whether a fraud is more likely to be committed in shopping rush hours or not. For simplicity, 24 hours are divided into 8 periods uniformly.

3.2.2 Feature abstraction

Feature abstraction is about transforming some data formats like ordinal or nominal values into model understandable ones. By looking into description of dataset, we can easily recognize whether a variable can be regarded as ordinal or nominal values or neither of them. In our dataset, there is no ordinal data like sequential ranks. All of them are nominal, or in other words, categorical.

One-hot encoding is a popular way to deal with categorical data in machine learning data preparation. For each unique value of one original categorical variable, a new virtual feature is created and given value of either 0 or 1 regarding each sample. This procedure could also help to encode missing categorical values into a new column considered as unknown.

With the completion of data preprocessing and feature engineering, data prepared for feeding into machine learning model should be totally complete, numeric and model comprehensive.

3.3 Machine learning modeling

For simplicity, we treat our fraud detection problem as binary classification, that is whether a transaction is a fraud or not. Compared to regular binary classification problems, data imbalance in fraud detection needs to be carefully handled. What's

more, model complexity is taken into account to avoid either overfitting or underfitting issues. In the following parts, our solutions to these problems are introduced.

3.3.1 Decision tree

Unlike linear models, information based models gain more strength in mapping non-linear relationships and empower predictive models with more interpretability and are more robust in dealing with outliers. Decision tree is one of information based models widely used in supervised classification problems.

Referring to its name, decision tree has an either binary tree or not structure. The sticking point in the model is to split attributes in each non leaf node and gain more purity in all subsets, in other words, to find optimal split that generates greatest information gain from parent node to children with respect to information theory. Most segmentation algorithms are based on top-down greedy strategy, two common among which are ID3 and C4.5 based on entropy.

Due to the characteristics of decision tree, feature selection is automatically embedded in the algorithm. Thus, unlike using other classifiers like logistic regression, feature selection is exempted. What's more, as decision tree does not use gradient descent to converge to optimal state with minimum loss, there is no need to do feature scaling to accelerate converging speed like standard scaling before fitting our model into data. However, drawbacks in decision tree can not be neglected. Decision tree can be too complex in depths to generalize the data. Lack of robustness makes the model weak in stability. As we mentioned before, greedy core of the algorithm in decision tree may trap model in a local rather than global optimum. And imbalance of dataset can affect performance of decision tree much. Optimization on decision tree to overcome these drawbacks will be explained in 3.3.2 and 3.3.3.

3.3.2 Optimization on imbalanced dataset

Among over 150 thousand samples, there are only approximate 200 samples were labeled as fraud. Imbalance in dataset can bring a lot of problems.

There are many alternatives to deal with data imbalance. Most common one is known as re-sampling. There are two major sampling approaches: over-sampling and under-sampling, either by adding samples or removing samples to reach balance in dataset. Here, under-sampling is not suitable in our problem due to huge gap between sample number of two classes. While over-sampling can significantly boost data to an uncontrollable size limited by our computational power. Thus re-sampling approach is discarded by us.

Regarding defects of decision tree in handling data imbalance, we seek for ensemble methods to solve the problem, like random forest. Random forest, literally, consist of randomly picked decision trees and there is no correlation between trees. Besides maintaining several strengths by decision tree like exemption of feature selection,

random forest exploits bagging algorithm to help solve imbalance problem. More specifically, random forest creates child sample datasets with an equivalent size of their original parent dataset by bootstrap sampling, which means repetition is allowed within and among datasets. Bootstrap effectively sooths the bias between positive and negative samples. What's more, complexity of each single tree can be reduced in ensemble forest to avoid overfitting while in the mean time guarantee the model is still qualified in complexity.

Another ensemble methods considered in the cost-sensitive framework to handle imbalanced domains which biases the learning towards the majority class in learning phase is AdaBoost, the most representative algorithm of the boosting family.[5] Adaboost uses iteration to update weight of each sample. In each iteration, we earn a learner and based on its performance we update the weights based on the strategy of lowering weights of correctly classified samples while increase the wrong. Last output will be a linear combination of learners with bias on weight respect to their performances. In our AdaBoost classifier, we use decision tree as base learner.

Besides model optimization, performance metrics should also be carefully chosen when we evaluate our models. Though accuracy is generally used in most classification problems, in imbalanced domains, it is typically not suitable as we can easily cheat by labeling all samples to the majority class and gain a high accuracy. Under this circumstance, we care about correctness among samples predicted as fraud and also how much fraction of all fraud is detected. Based on confusion matrix, we can calculate precision and recall. And scoring metrics used in our learning phase is F1 score, a trade off between precision and recall.

3.3.3 Hyper-parameter tuning

Hyper-parameter is a manually defined parameter before leaning process which can not be learned by training, related to model complexity. In tree and ensemble methods, hyper-parameter can be max depth of a tree or number of trees in a random forest. When a model is too complex, it is prone to overlearn in training data and lose its generality. On the contrary, insufficient complexity can lead to insufficient learning from training data and make model not representative enough on the problem. Besides splitting training and testing data before learning process, we can use cross validation on our training partition to tune our model and get a balancing point between model bias and variance. Figure 5 gives a graphical explanation of mechanism of K-fold cross validation. Considering imbalance, we use stratified K-fold here to guarantee in each fold there exist samples of minority class.

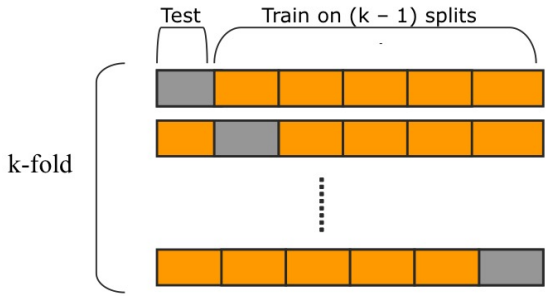


Figure 5: K-fold cross validation

Source: http://qingkaikong.blogspot.ch/2017/02/machine-learning-9-more-on-artificial.html

Usually, there could be more than one hyper-parameter. And a common strategy is to use grid search on hyper-parameter space. Increasing dimensionality of space help tuning model towards optimum while in the mean time bring extra cost in cross validation process as the time complexity grows exponentially to the dimensionality. And trade off between grid accuracy and time complexity should also be considered.

3.3.4 Experiment and model evaluation

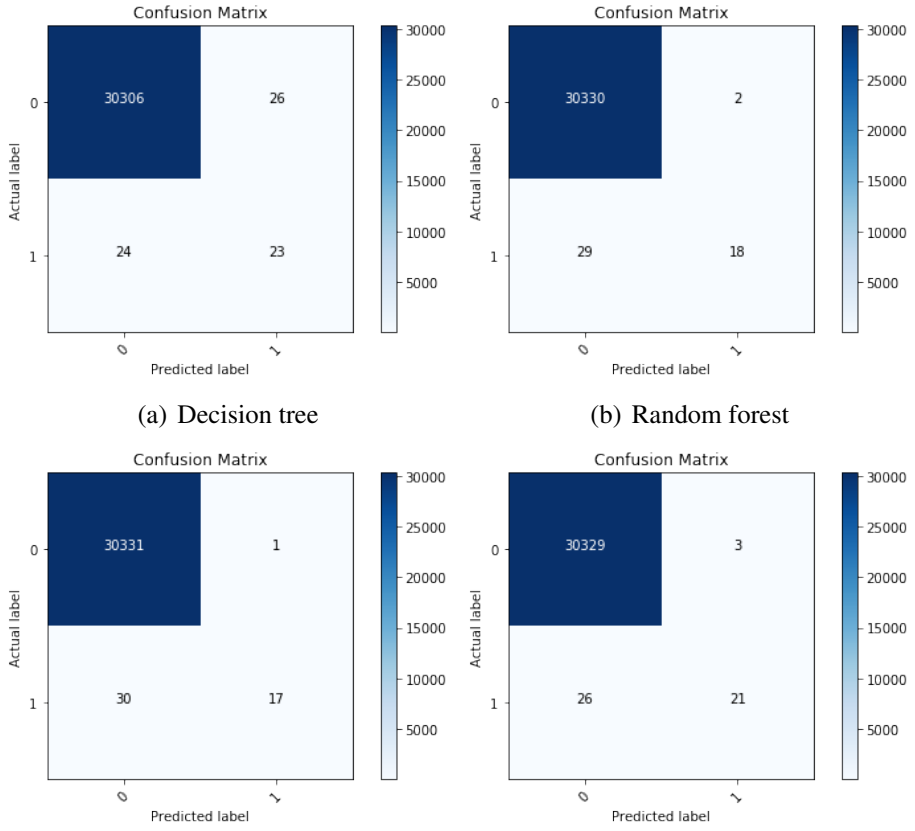
In our experiment, we randomly split data into training and testing set with a ratio of 8 to 2. Stratified five fold cross validations were done separately on each classifier to pick out best hyper-parameters regarding mean F1 score on validation set. Then we fit best estimators on testing data to evaluate our trained classifiers based on different performance metrics.

Decision tree classifier optimizes its performance at depth of 20, reaching a F1 validation score above 0.4 as its best. Among features chosen by the classifier, transaction amount is given top priorities. The other features involving location of transaction are also of high importance. By testing the classifier on 20% testing data, we generated a confusion matrix as shown in figure 6(a). There is a close tie between precision and recall. Among 49 samples predicted as fraud, over 50% are false positives. And nearly 50% fraud cases are detected. As we said before, due to defects of decision tree on imbalanced domain, we can just regard it as a baseline and see how random forest and AdaBoost perform.

We build a random forest with 100 estimators with depth of 20, referring to best depth in cross validation of decision tree classifier. Optimal number of features used is chosen on stratified five fold cross validation similarly. And an optimum of 53 features is chosen over 69 features in total, among which transaction amount is still the most important with a normalized importance score of 0.25. Referring to confusion matrix of random forest in Figure 6(b), false positive rate is significantly reduced though there is a decline in recall.

Best AdaBoost classifier using 100 decision trees of depth of 20 as base learner has a

learning rate around 1.6. It could get a precision score of 0.94, though having same low recall problem as random forest, given confusion matrix in Figure 6(c).



(a) Decision tree (b) Random forest

(c) AdaBoost (d) Soft voting of three classifiers

Figure 6: Confusion matrices of classifiers on testing data

As three classifiers have performing strength differently, we try to combine them using soft voting to balance out their individual weakness and aim for a better performance. Each classifier is given equal weight as we consider them as equally well performing. As we can see the confusion matrix in Figure 6(d), compared to every single predictor, the voting classifier has better performance on either precision and recall, possessing highest F1 score of 0.59.

Receiver operating characteristic(ROC) curve is another way to evaluate machine learning models. ROC curves for four classifiers are generated as shown in Figure 7. The dashed diagonal line is as random guess and no classifier should have worse performance than this. For each curve of one classifier, the coordinates are about true positive rates and false positive rates dependent on probability threshold we set varying from 0 to 1. The closer a curve is to the left upper corner, the better performance a classifier has, which is also reflected on the area under curve(AUC). AdaBoost and classifier by soft voting have largest AUC close to maximum as 1. And decision tree classifier has poorest AUC of 0.74.

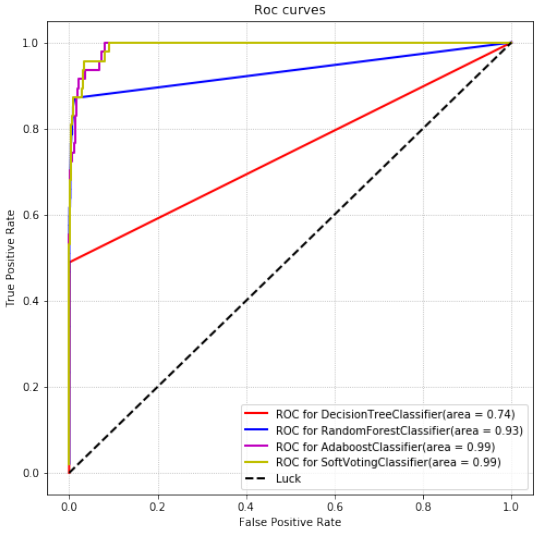


Figure 7: ROC curve of classifiers with calculated area under curve

4 Conclusion and Deployment

4.1 Conclusion

In our subject, we analyze the fraud behaviors through the transaction dataset and fraud case dataset originating from 17.11.2014 to 19.02.2018, provided by the company, KiWi Bop. During this period, 169,981 transactions took place (let's recall that we are filtering those shops which own a KiWi device and which transactions have been made by card and not by cash, that's why we have less than the 200,000 transactions that they in the whole dataset), including 260 fraud cases. Among theses fraud cases, the transaction amount ranged from 350 MXN to 22,000 MXN. Although the fraud cases happened rarely, the company loss was large, not only in profit, but also in reputation. The fact is that the number of shops using KiWi decreased to around 0.25 of the original number during this period. Despite the fraud, which is likely to make the merchant lose money, was not the immediate cause, the company should pay attention to this and develop fraud detection mechanisms or rules.

Firstly, in the step of understanding dataset, we checked the overall situation of the whole transactions and the activeness of shops. Then, we focused on the fraud analysis. Some typical characteristics were sorted, like most of the frauds are approved when doing the transaction , all the frauds are paid in Mexican pesos and fraud paid by credit cards is more likely to be affirmed. After that, we analyze the fraud cases by the statistics, mainly focusing on the transaction amount and industry types. Considering that the users of KiWi devices are often small shops which often don't have bank terminals, the amount of each fraud case is quite large with a median of 5,400 MXN. After finding the largest industry where the fraud detected happen mostly is "Food and beverages", we conducted in-depth analysis at three levels, shop level, card holder level and location level. At shop level, there were 38 shops which had more fraud transactions than normal transactions or only fraud transactions. After finding an abnormal shop by analyzing the number of devices, cellphones and transactions, we looked at the fraudulent cardholders with most fraud transactions. When we turned our attention to the location of the transaction, we concluded that some merchants also did fraud and not only the card holders.

As for the modeling part, machine learning is adopted in our project. On the basis of our analytical finding, we found that no matter which modeling method, the total amount of each transaction is always the most important features on the fraud behaviour. Therefore, the company should take some measures based on the transaction amount, like setting the threshold of amount per transaction or a largest amount one card can pay per day. After dealing with the effect of imbalanced datasets, we found that ensemble methods, like Random Forest, AdaBoost and Soft voting, have a good performance of detecting fraud. To some extent, false Positive which means the non-fraud is evaluated as fraud by prediction is more dangerous in real business cases than the condition when the model cannot detect a fraud. If our model detects an honest customer as fraud, the customer will be mad at KiWi, which results in a bad reputation

in business and seriously has a bad influence on the development of KiWi. Therefore, we shall also consider the misjudgment rate of each models. In this case, modeling based on AdaBoost and Softvoting has a good performance, as there are less honest consumers misjudged as fraud. Considering the influence of imbalanced dataset, we selected F1 values as assessment of models, which measures the trade-offs between the value of True Positive (real fraud is predicted as fraud), False Positive (real honesty but is predicted as fraud) and False Negative (real fraud but is predicted as non-fraud). Softvoting of three classifiers, which are Decision Tree, Random Forest and AdaBoost, has a highest F1 score. In other words, this model makes a good prediction which not only keeps the judgment accuracy, but also reduces the probability when a customer is misjudged as a fraud.

4.2 Deployment

So far, we manage to provide a sufficient fraud detection solution for company, whereas the false fraud cases is still not easy to tackle with. In KiWi transaction records, it happens sometimes that a normal transaction is classified as a potential fraud and the money is retained by KiWi bank. The retention may cause severe problem to clients' social life and draw dissatisfaction of clients. If KiWi fails solving this question, it will lose clients. To avoid customer churn, KiWi company needs to set threshold more precisely to distinguish false detection from real fraud. Based on our feature analysis, we basically want to sort out several important features related to fraud and set them as rules to further determine a transaction is a fraud or not in the following work.

Since the most connected feature of fraud is transaction amount, we may start fitting rules from it. From the data analysis, we know that the fraud amount ranges from 0 to 50,000 MXN and concentrates between 0 and 10,000 MXN. Thus, we could set threshold somewhere, taking the median 5400 MXN for example, as a critical value to detect fraud. However, we definitely cannot determine fraud solely by transaction amount. There are some other features we need to combine with. According to preceding data analysis, the transaction frequency of a card is also an important feature. For example, in our datasets, a credit card made 4 transactions within 5 minutes in different sops, which has been labeled as fraud. Thus, we could set rules of having consecutive transactions or not within a short period to screen fraud. Another feature that worths to notice when setting rules is shop affiliated industry. Each industry has its own transaction level.It does not make sense having a transaction of 15,000 MXN in food and beverage industry, which did happen and labeled as fraud. One way to figure it out is to set appropriate transaction amount as threshold to each industry. This can avoid large amount of fraud in micro-merchants to some degree. Iteratively, we can specify more features and integrate them together to determine at what condition a transaction could be classified as fraud and minimize false alarm at the same time.

Given the fact that KiWi Mexico is an emerging business with not that huge total amount of transactions, let alone fraud cases. The extremely unbalance between fraud transactions and normal transactions brings difficulty in precisely targeting false posit-

ive. What we currently do by adopting ensemble methods does not necessarily include all critical features related to fraud. As the inevitable emergence of different types of fraud in the future, we will need to add more specified rules to detect fraud.

A Figures

A.1 Machine learning

A.1.1 Decision tree classifier

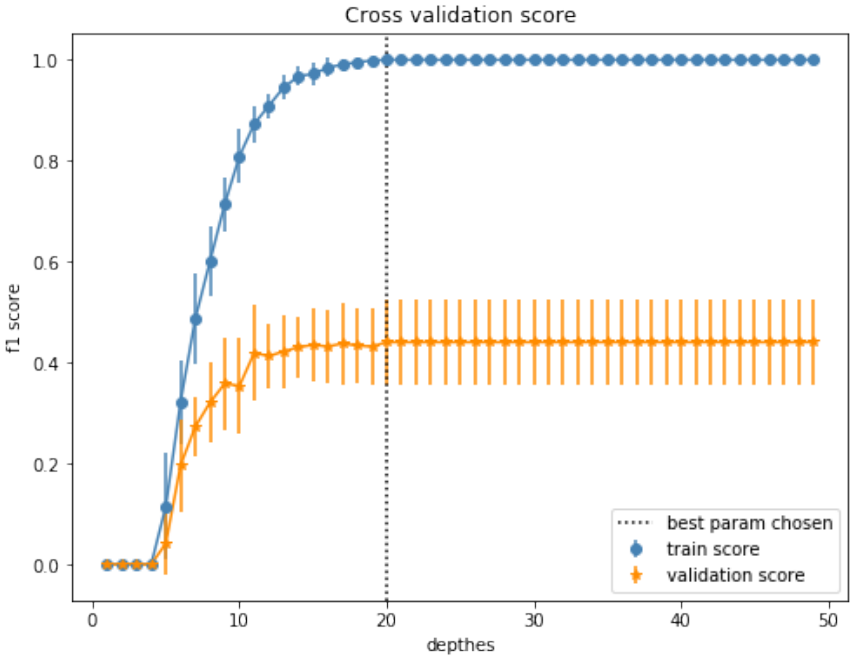


Figure A.1.1.1: Cross validation result of decision tree classifier

Figure A.1.1.1 shows the 5 fold cross validation result of decision tree classifier. Best depth is chosen at 20 with respect to top mean test F1 score. With the depth of tree increasing, F1 score of training set grows to 1 with less variance. For testing set, F1 score fluctuates more when depth becomes deeper.

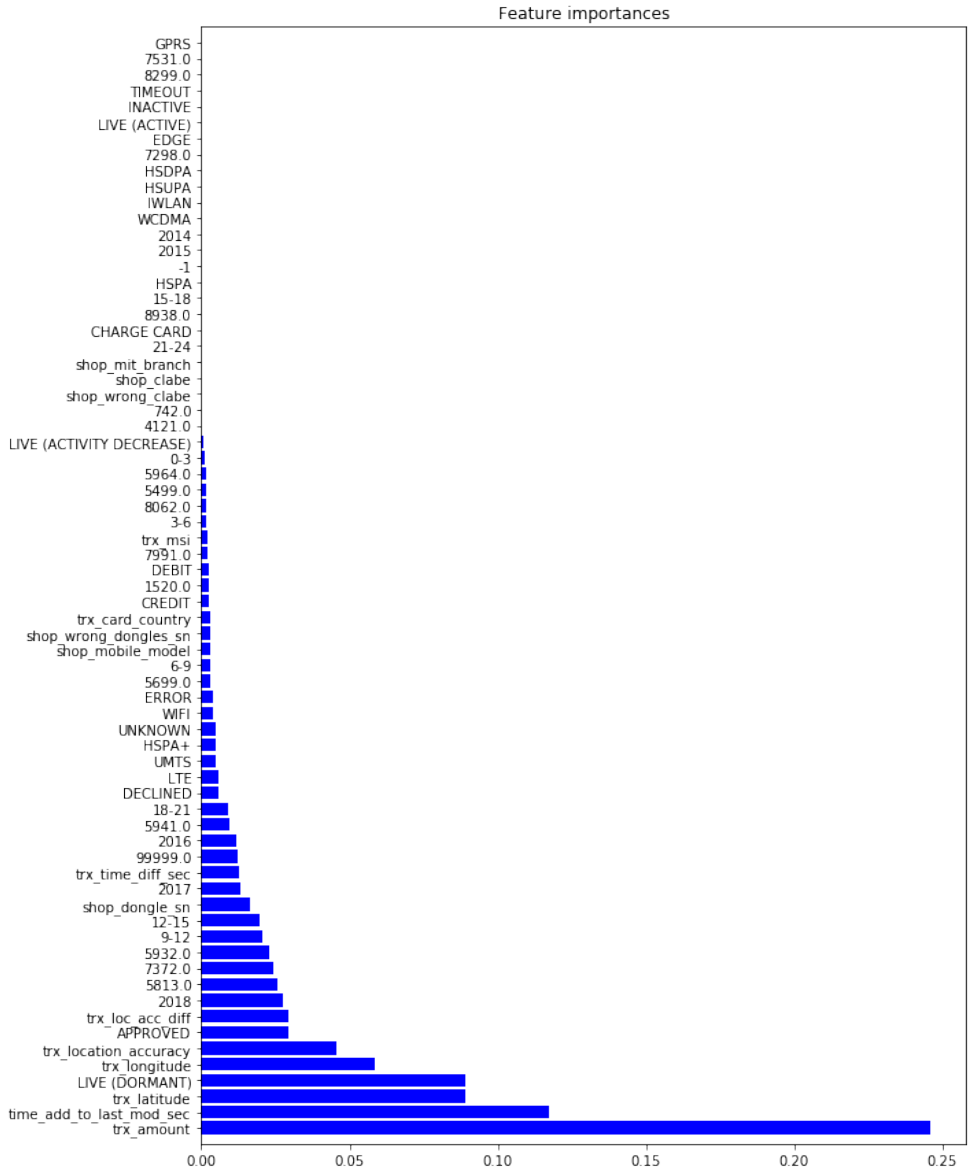


Figure A.1.1.2: Feature importance given by decision tree classifier

Sorted feature importance given by decision tree classifier is given in Figure A.1.1.2. Transaction amount is given top importance with a weighted score of around 0.25. Location and the time factor of transactions are also considered as important features.

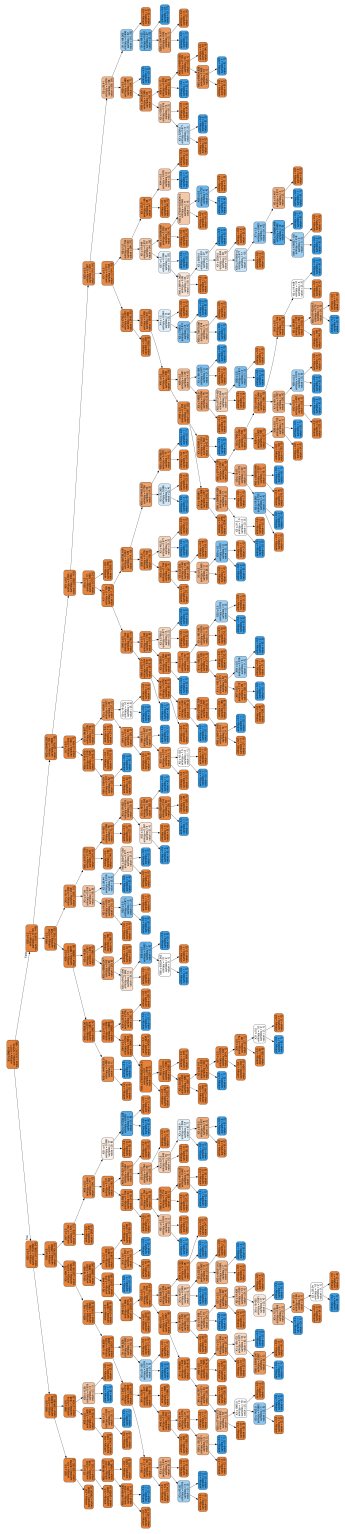


Figure A.1.1.3: Tree structure fitting on training set

Figure A.1.1.3 gives the tree structure of decision classifier after fitting on training set. At each node, entropy, split decision made and sample number are given.

A.1.2 Random forest classifier

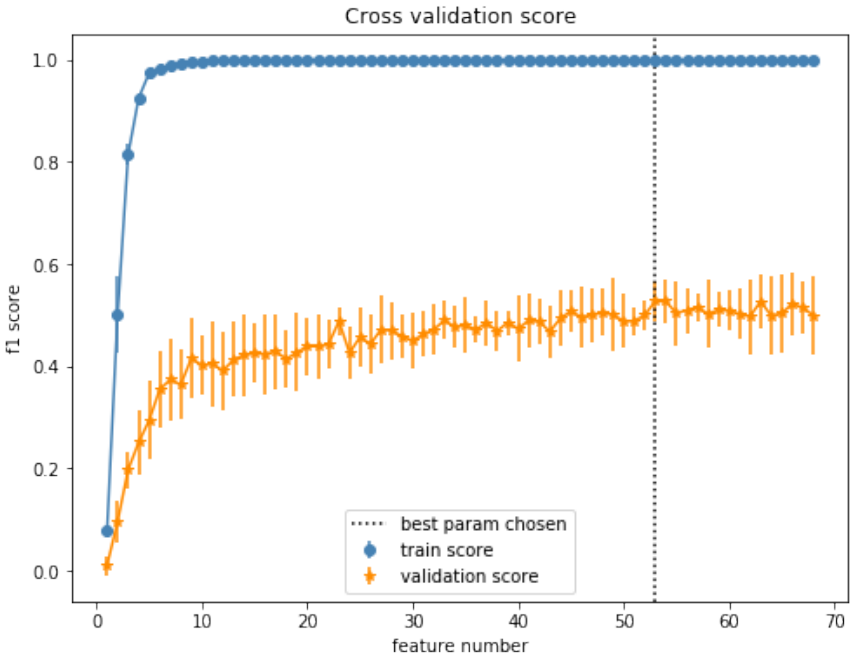


Figure A.1.2.1: Cross validation result of random forest classifier

Similar to decision tree classifier, Figure A.1.2.1 shows the cross validation result of F1 score on both training and validation set with respect to different feature numbers with random forest classifier consisting of 100 random trees with depth of 20. Best hyper parameter is chosen at 53.

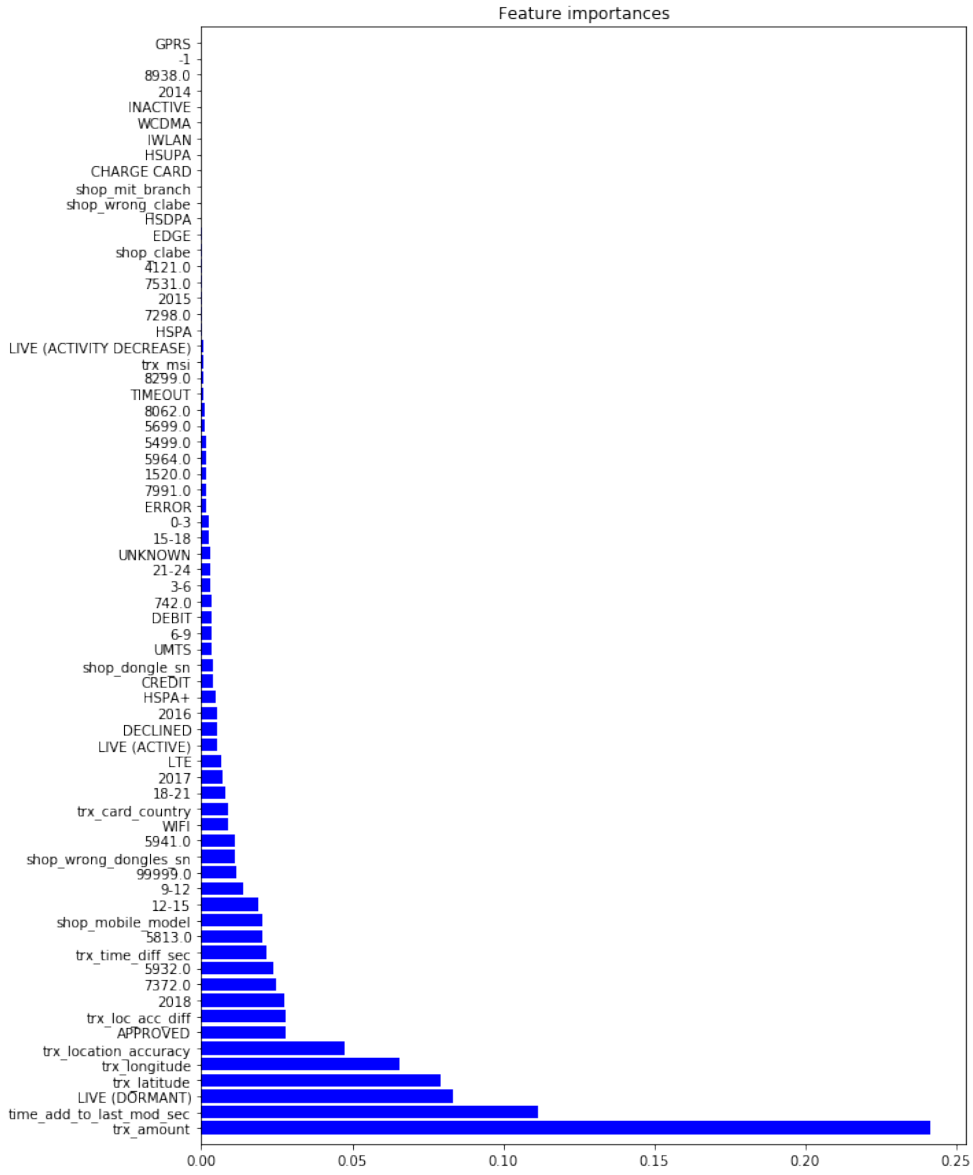


Figure A.1.2.2: Feature importance given by random forest classifier

Figure A.1.2.2 gives feature importances by random forest classifier. Not so different from rank given by decision classifier in Figure A.1.1.2, random forest also considers transaction amount and location information as most important ones.

A.1.3 AdaBoost classifier

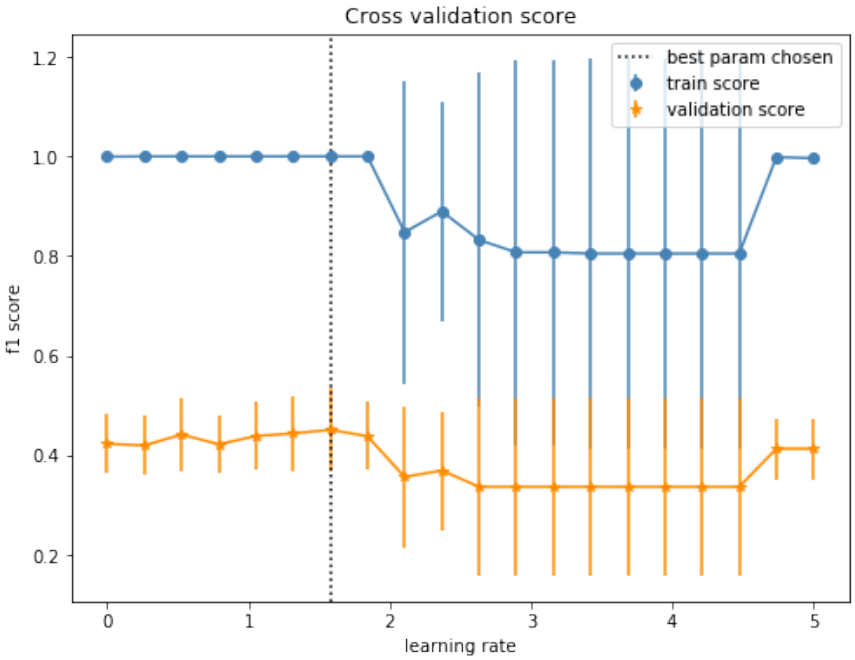


Figure A.1.3.1: Cross validation result of AdaBoost classifier

Figure A.1.3.1 is about cross validation result of Adaboost classifier with decision tree as its base learner. According to the results of validation set, best learning rate is around 1.6.

References

[1] "KiWi – making fintech work for micro-merchants in Mexico," http://scbf.ch/wp-content/uploads/2011/03/SCBF-PU-2016-01-KiWi-Mexico-Final-Report-edited1.pdf, December, 2016.

[2] "Introduction of KIWI company," https://gust.com/companies/UseKiwi.

[3] T. R. Hoens and N. V. Chawla, "Imbalanced datasets: from sampling to classifiers," *Imbalanced Learning: Foundations, Algorithms, and Applications*, pp. 43–59, 2013.

[4] A. Bilogur, "Missingno: a missing data visualization suite," *Journal of Open Source Software*, vol. 3, no. 22, p. 547, 2018.

[5] P. Branco, L. Torgo, and R. Ribeiro, "A survey of predictive modelling under imbalanced distributions," *arXiv preprint arXiv:1505.01658*, 2015.