Data Analytic 96-777 Team 8: Analyzing Customer and Agent Transactions on the EKO platform*

Daniel Noguchi
daniel.noguchi@sv.cmu.edu

Poorva Jain
poorva.jain@sv.cmu.edu

Pantelis Takos
pantelis.takos@sv.cmu.edu

ABSTRACT

This paper provides an analysis of transaction data patterns and customer/agent behavior for EKO, a Mobile Cash-to-Account Money transfer service. EKO provides peer-to-peer money transfer service to individuals through next door mom-n-pop stores. Each transaction involves stakeholders like customers, retail agents, distributors and banks. We analyzed the behaviors of customers, agents and general transaction patterns during the period of December 2014 to February 2015 for the dataset of transactions to SBI, one of EKO's clients. Our analysis aimed to identify the root cause of loss in customer base and profitability for EKO. Initial findings suggested that the datasets analyzed were not large enough to draw conclusions on profitability or fraud detection, but the results of customer behavior were good. Our model for detection customer changing agent or not had an accuracy rate of 95.12%, which can help EKO to better serve their customers or target their efforts depending on the profile of the user.

1. INTRODUCTION

In this paper we will examine money transaction patterns and agent behavior for a Mobile Banking company. We received our data set from an Indian financial company named as "EKO India financial services". EKO is a business correspondent(banking agent) for the two largest banks in India named as State bank of India(SBI) and ICICI. EKO provides peer-to-peer money transfer service to individuals. Each transaction involves stakeholders like customers, retail agents, distributors and banks. All datasets followed the same header structure, with the following columns: "TX_Time", "SCSP_Code", "CSP_Circle", "CSP_code", "Depositor_Mobile", "CSP_Mobile", "NEFT_Account", "Tx_Amount", "Fee_Amount", "Income post ST", and "IFSC_Code". These columns collectively represent the transaction time, the agent that initiated the transaction, the sender, recipient, the amount and finally, the commission. In our report we are focusing our analysis:

- To find out if there is any money laundering scenario using R analytics.
- To find transactional pattern behavior by different Data Analytics method for improving company business in future.

Our goal for this project is to understand the correlations between the customers, beneficiaries and the agents. We have broken down our task into two parts: *Customer behavior analysis* and *Agent behavior analysis*.

For Customer behavior analysis we tried to understand the transaction patterns of each customers. To achieve this task we did pattern analysis on datasets covering a three month period and tried to find out the following:

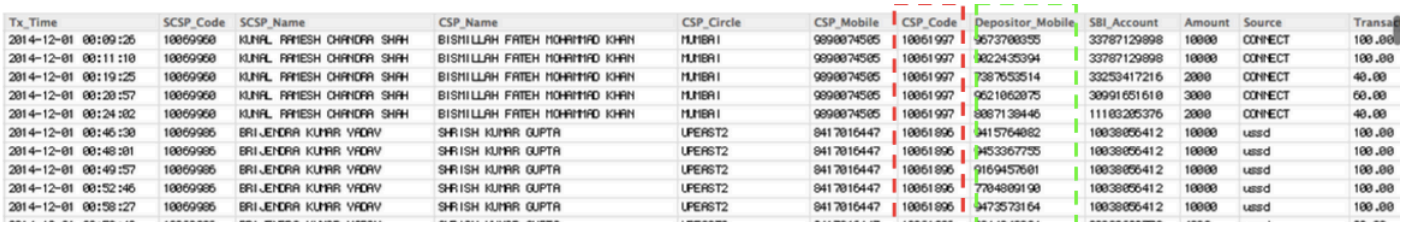
- Percentage of repeat customers within each month for three months and for the overall the 3-month period.
- Transaction patterns: timing (dates/days/weeks), amount, customer/agent bonding.
- Transaction repetition behavior in beneficiary account.

For Agent behavior analysis we focused on finding the transaction time and number of transaction by each agent. To achieve this task we analyzed three months of datasets and tried to find out following patterns:

- Percentage of repeated transactions.
- Distribution of transaction on the basis of time.
- Beneficiary account and agent relationship.
- Number of transaction for the same account in a time period.
- Agent preferred source of transaction.
- Agent preferred transaction time.

We used our analysis to identify the root causes of loss in customer base and profitability for EKO. Our findings also may help EKO to improve its business and leverage their strategy with data-driven insights for upcoming product.s

*EKO India Financial Services



Tx_Time	SCSP_Code	SCSP_Name	CSP_Name	CSP_Circle	CSP_Mobile	CSP_Code	Depositor_Mobile	SBI_Account	Amount	Source	Transa
2014-12-01 00:09:26	10069960	KUNAL RAMESH CHANDRA SHAH	BISMILLAH FATEH MOHAMMAD KHAN	MUMBAI	9890074505	10061997	9673708855	33787129898	10000	CONNECT	100.00
2014-12-01 00:11:10	10069960	KUNAL RAMESH CHANDRA SHAH	BISMILLAH FATEH MOHAMMAD KHAN	MUMBAI	9890074505	10061997	9822435394	33787129898	10000	CONNECT	100.00
2014-12-01 00:19:25	10069960	KUNAL RAMESH CHANDRA SHAH	BISMILLAH FATEH MOHAMMAD KHAN	MUMBAI	9890074505	10061997	7387655514	33253417216	2000	CONNECT	40.00
2014-12-01 00:20:57	10069960	KUNAL RAMESH CHANDRA SHAH	BISMILLAH FATEH MOHAMMAD KHAN	MUMBAI	9890074505	10061997	9621062075	30991651610	3000	CONNECT	60.00
2014-12-01 00:24:02	10069960	KUNAL RAMESH CHANDRA SHAH	BISMILLAH FATEH MOHAMMAD KHAN	MUMBAI	9890074505	10061997	8087138446	11103205376	2000	CONNECT	40.00
2014-12-01 00:46:30	10069986	BRIJENDRA KUMAR YADAV	SHRISH KUMAR GUPTA	UPEAST2	8417016447	10061896	9415764882	10038056412	10000	ussd	100.00
2014-12-01 00:48:01	10069986	BRIJENDRA KUMAR YADAV	SHRISH KUMAR GUPTA	UPEAST2	8417016447	10061896	9453367755	10038056412	10000	ussd	100.00
2014-12-01 00:49:57	10069986	BRIJENDRA KUMAR YADAV	SHRISH KUMAR GUPTA	UPEAST2	8417016447	10061896	9169457601	10038056412	10000	ussd	100.00
2014-12-01 00:52:46	10069986	BRIJENDRA KUMAR YADAV	SHRISH KUMAR GUPTA	UPEAST2	8417016447	10061896	7304809190	10038056412	10000	ussd	100.00
2014-12-01 00:58:27	10069986	BRIJENDRA KUMAR YADAV	SHRISH KUMAR GUPTA	UPEAST2	8417016447	10061896	9473573164	10038056412	10000	ussd	100.00

Figure 1: Transaction Log

2. EXPLORATORY WORK

Our goal is trying to find the patterns from the dataset that generalize the relationships between customers and agents, describe the reasons that lead to customers decisions of switching agents, and optimize the profitability or revenue by tuning the fee or commission rate. Figure 1 shows the dataset of transaction log. We use the column of the depositors mobile number to identify the customers, and CSP_Code to identify different agents.

2.1 Pattern Analysis

2.1.1 Customer and Agents Clustering

To gain more insights on the interactions among repeat customers, agents, and distributors, and to identify the top performing agents and distributors, we conducted a clustering analysis of the banking transaction data in the context of transaction networks, which are constructed based on the connections between repeat customers and agents, between repeat customers and distributors, and between agents and distributors. In the transaction network for repeat customers and agents, for example, the nodes denote the repeat customers or agents, while the edges represent the transactions that have been made between a repeat customer and an agent. A similar transaction network is constructed for repeat customers and distributors, in which the edges denote the transactions conducted between a repeat customer and a distributor. The transaction network for agents and distributors is defined slightly differently. In this case, an edge between an agent and a distributor indicates that the agent conducted the transaction through that particular distributor. Investigating these transaction networks will allow us to identify the most popular agents and distributors from the network visualizations and observe customer behaviors. The tool we used to conduct this visual clustering analysis is Gephi, which is a powerful visualization tool that makes it easy to intuitively discover patterns. As Gephi can handle networks containing up to 50,000 nodes and 1M edges, it can generate complete network visualizations from the monthly banking transaction data. Prior to generating the networks, we processed and filtered the transaction data to extract only the features to be used as nodes in the transaction networks. For example, in the agent-repeat customer network, we first extracted out all of the repeat customers who have conducted at least one transaction within the month. Since we are only interested in identifying the connections between agents and repeat customers, we removed the duplicate transactions between the same pair of customer and agent. Therefore, the result data consist of two columns: CSP_Code, which represents the agent who has transferred money for at least one repeat customer, and Depositor_mobile, which is the identifier for the repeat customer who has initiated more than one transaction. The same filtering process is applied to the distributor-repeat customer network, in which case SCSP_Code represents the distributor who has transferred money for repeat customers. A similar filtering process is applied to the distributor-agent network. In this scenario, the same agent might also have conducted multiple transactions through the same distributor. However, for the sole purpose of visualizing the agent-distributor relationships, we removed those duplicate data and keep only SCSP_Code and CSP_Mobile as identifiers for distributors and agents in the processed dataset. Once we filtered the data, we added to the processed data columns an additional feature to indicate the type of the connection between the nodes in each transaction network. Since these are transaction data, the edge in the network is defined to be undirected. Once source node, target node, and type of connection are all established, the data is imported into Gephi to generate network graphs along with a statistics panel that computes a set of network metrics. After applying the same data processing procedure to all three transaction networks, we summarized the network statistics and visualizations. In addition to visualizing the networks, Gephi uses various algorithms to computes a set of metrics that can statistically characterize the networks. The community detection algorithm, which can detect the community structure of the network, is of special importance to the transaction networks being studied. By applying the community detection algorithm, each transaction network is divided into clusters such as sets of nodes within a cluster is densely connected internally and sparsely connected with nodes from different clusters. By partitioning each transaction network into communities using the community detection algorithm, we can easily identify the top performing agents and distributors in each transaction network. In addition, we can visualize customer behavior such as their loyalty to agents by examining how repeat customers are tied with agents or distributors in each network.

Firstly, we constructed a transaction network to study the agent-to-repeat customer relationship. By filtering the nodes based on their degree, we extracted a graph that shows the nodes with the highest degrees in the network. As illustrated in figure 1, each node represents a popular agent connected to a large number of repeat customers. On the node label, the 8-digit number is the agent identifier, followed by a number which signifies the degree of each node. Relating to the transactions data, the node degree of the transaction network represents the number of repeat customers an agent is connected to. Based on this visualization, we can easily identify the most popular agents. In addition to identifying the most popular agents, clustering analysis of the

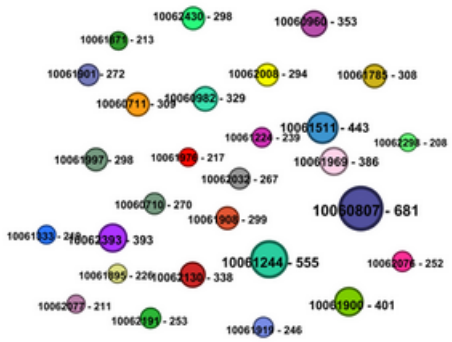


Figure 2: Most popular agents in the repeat customer-to-agent transaction network

Figure 3: Most popular agents in the repeat customer-to-agent transaction network

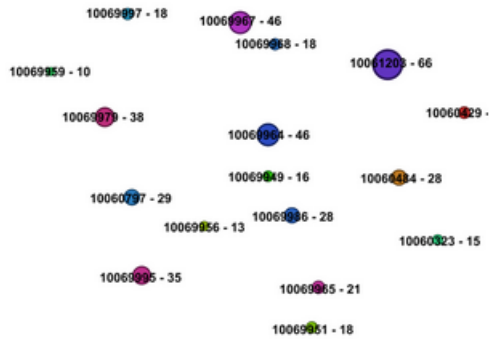


Figure 4: largest distributors in the distributor-to-agent transaction network

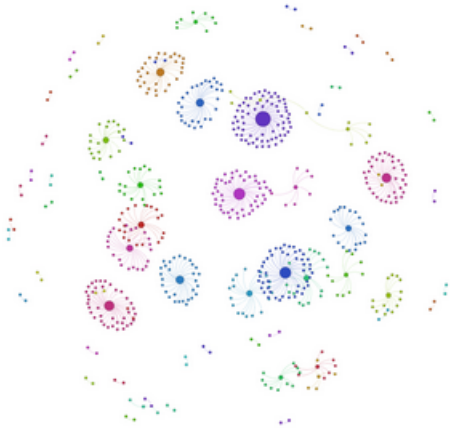


Figure 5: largest distributors in the distributor-to-agent transaction network

transaction network also allows us to discover customer behavior patterns. By visualizing the network edges, which represent connections between agent and repeat customers, we can gain insight on how repeat customers are associated with agents. In the second transaction network visualization, edges are shown as well as the nodes to demonstrate how repeat customers are connected to the agents. In order to improve the visibility of the network, we picked up five popular agents and their associated repeat customers from the original transaction network. As shown in figure 1, the resulting network graph is separated into five clusters. The five popular agents are seen as the "hubs" in the network, with each connected to a large base of repeat customers. In addition, we can see that repeat customers tend to be associated with one agent only, which implies that repeat customers are very loyal to their agents and are resistant to switching agents.

Similar to the agent-repeat customer network defined above, we also constructed transaction network to study the relationship between distributors and agents. In EKOs business model, distributors act as the super agents who manage a number of agents. By filtering the nodes based on their degree, we extracted a graph that shows the nodes with the highest degrees in the network. As illustrated in figure 2, each node represents a popular distributor connected to a large number of agents. On the node label, the 8-digit number is the distributor identifier, followed by a number that represents the degree of each node. Relating to the transactions data, the node degree of the transaction network represents the number of agents connected with each distributor. Based on this visualization, we can easily identify the largest distributors. Furthermore, we can verify that all of the big nodes represent distributors, which are easily distinguished from agents by their node labels. Since the number of distributors is not as large as in the case of the agent-repeat customer network, we can generate a visualization of the complete distributor-agent transaction network with relatively good visibility. As shown in figure 2, the network is clearly segregated into different clusters with nearly no connections between different clusters. There are many large distributors seen as "hubs" in the network, with each connected to a large base of agents. Agents, on the other hand, are typically connected to a single distributor, which implies that agents tend to work under a single distributor.

Finally, we constructed a transaction network surrounding

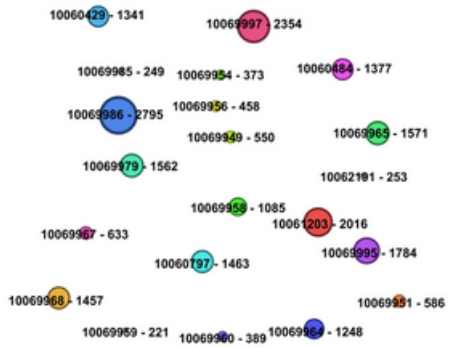


Figure 6: most popular distributors in the distributor-to-repeat customers transaction network

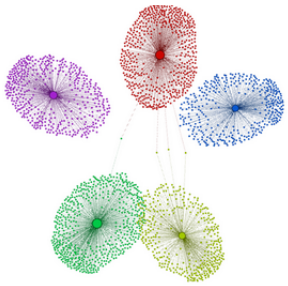


Figure 7: most popular distributors in the distributor-to-repeat customers transaction network

distributors and repeat customers. In this network, edges represent transactions occurred between repeat customers and distributors. By filtering the nodes based on their degree, we extracted a graph that shows the nodes with the highest degrees in the network. As illustrated in figure 3, each node represents a popular distributor connected to a large number of repeat customers. On the node label, the 8-digit number is the distributor identifier, followed by a number which signifies the degree of each node. Relating to the transactions data, the node degree of the transaction network represents the number of repeat customers connected to each distributor. Based on this visualization, we can easily identify the most popular distributors. In addition to identifying the most popular distributors, clustering analysis of the transaction network also enables us to discover customer behavior patterns with regard to the distribution channel preferred by customers. By visualizing the network edges, which represent connections between distributors and repeat customers, we can visualize how repeat customers are associated with distributors. In order to improve the visibility of the network, we picked up five popular distributors and their associated repeat customers from the original transaction network. As shown in figure 3, the resulting network graph is separated into five clusters. For all five clusters, the five popular distributors are seen as the ”hubs” in the network, with each connected to a large base of repeat customers. In addition, we can see that most repeat customers send money through the same distributors, which makes sense as agents are loyal to their distributors and repeat customers are loyal to their agents. However, there are a few repeat customers who are connected to more than one distributor, which means they have made transactions through different agents and hence exhibit agent-switching behavior. Further analysis will investigate what makes these repeat customers switch to different agents.

2.1.2 *Customers Behavior on Switching Agents*

To analyze these data, the first task is to clean the data to get only those we are interested. So we firstly identify one time customers and repeat customers. Figure 8 shows the proportion of transactions made by one time customers and repeat customers. Then, from repeat customers, we tried to identify loyal customers and fickle customers in terms of whether they are switching agents or not. Figure 9 shows

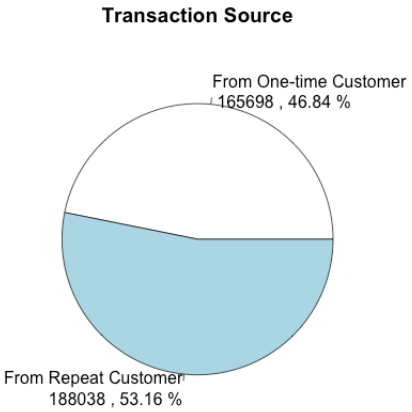


Figure 8: One-time and Repeat Customers

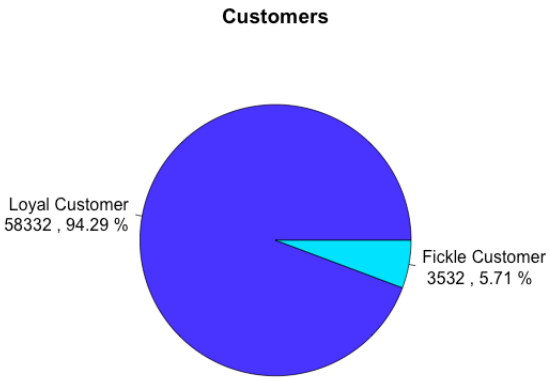


Figure 9: Customers Preference

the proportion of loyal customers and fickle customers. And then we kept only the transactions made by these fickle customers and filtered out the rest of them. Figure 2.1.2 shows the proportion of transactions made by fickle customers from loyal customers. Transactions made by fickle customers will be our clean data, because we can only observe agents switching behaviors from these data. We sorted the transactions by customer id and time, such that the transactions made by the same customers are aligned together and sorted by time. We define 4 customer behaviors here regarding the agents they chose with respect to time, as depicted in Figure 11. And we applied a simple algorithm to label the behaviors in the whole data set to create a target feature in this table. We named this feature *"switchMove"*. After we engineered this new feature, our problem becomes a typical association rule mining problem: to try to find the association rules between all other features in the datasets and the target feature, the *"switchMove"*. Before we tried to apply association rule mining algorithm, we first used some histogram to explore the relationship between our target feature and other features. Firstly, we analyzed the general features including SCSP_Code, CSP_Circle, and Source. As depicted in Figure 12, the distribution of the target feature among different values of these general features are evenly distributed, which means that no feature here is dominating the target feature. We then took a look at monetary features, including the amount of money in a transaction, the commission rate, and the fee rate. As depicted in Figure 13, again these features are not dominating the distribution of our target features either. We also analyzed the time related features regarding the time when a transaction was happened. For example, we observed the months, the days of a month, the weekday of a week, or even the hours of a day when the transaction was taking place. And we came up with the observation from Figure 14. We found that the days of a month played a significant role in a third portion of the data set.

Following this analysis, we also conducted some simple correlation on the data to visualize the transaction behavior

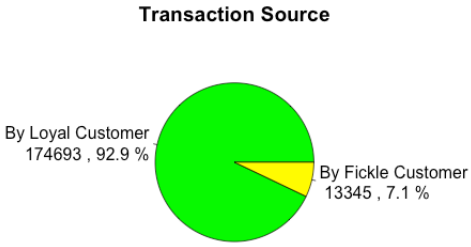


Figure 10: Transactions from Loyal/Fickle Customers

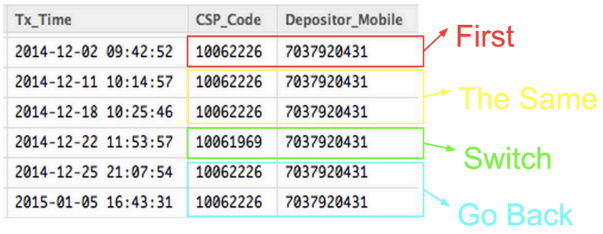


Figure 11: Defining and Labeling Customer Behavior

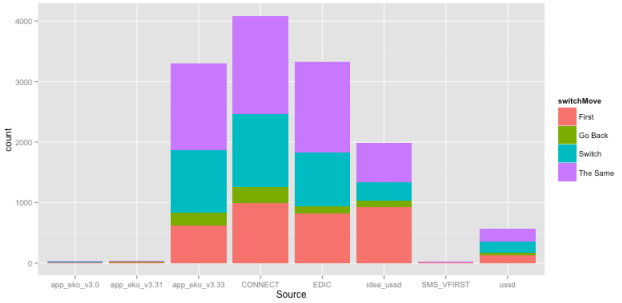


Figure 12: Transactions by Source

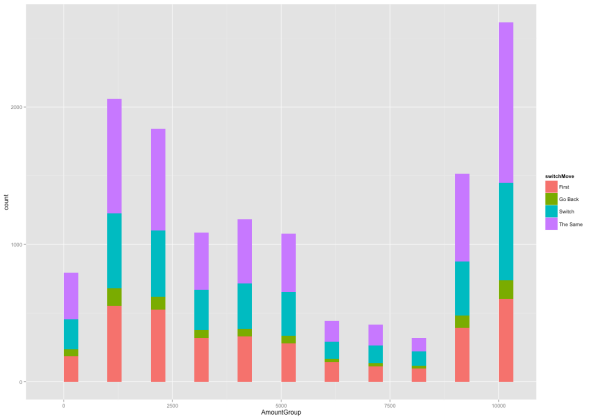


Figure 13: Transactions by Amount

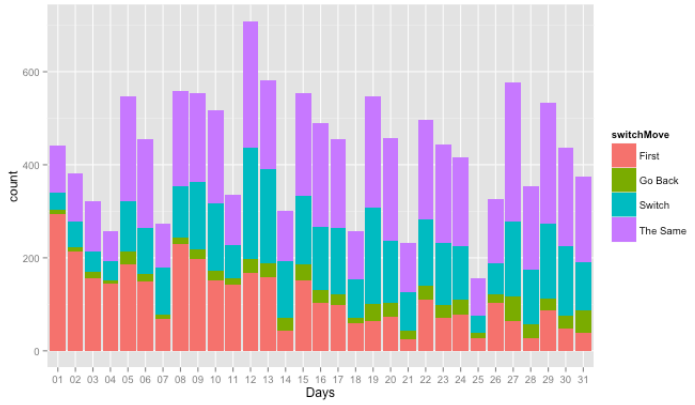


Figure 14: Transactions by Days

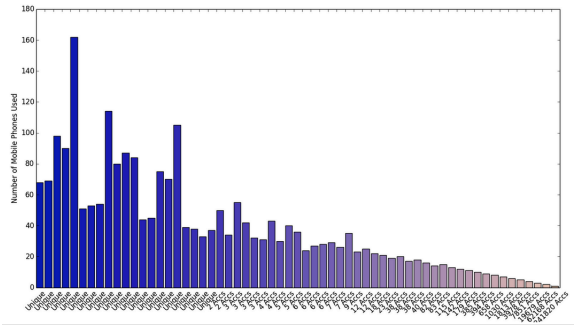


Figure 15: Mobile Numbers and their Account Destinations

across users and destination accounts. In order to do that, we correlated the number of mobile phones that were used to do transactions to a particular account.

We can see that the majority of users only use a single or a few mobile numbers to do transactions to a single account. This is validated by the rightmost portion of Figure 15, where 241820 Destination Accounts had *only one mobile number associated with it.* The leftmost part of the figure revealed some interesting unique accounts, which more than 50 mobile numbers where in association with them. Although this does not indicates fraudulent behavior, it raises suspicions on why these accounts have so many mobile numbers associated with them. Further analysis with other techniques (Classification Model based on Naive Bayes Classifier) were done to further investigate money laundry behavior.

2.2 Regression

Another way to explore our data is to use some statistical methods to find out relationship between each features. Therefore, here we propose the most simple model, linear regression analysis[2], to deal with our data.

Before implementing the real transaction data, we have to set up predicted target for the regression model. The current challenge for EKO is that he needs to provide higher commission rate to attract agent and distributor. Because there are also many similar competitors in the market and all need agent and distributor to help to establish business

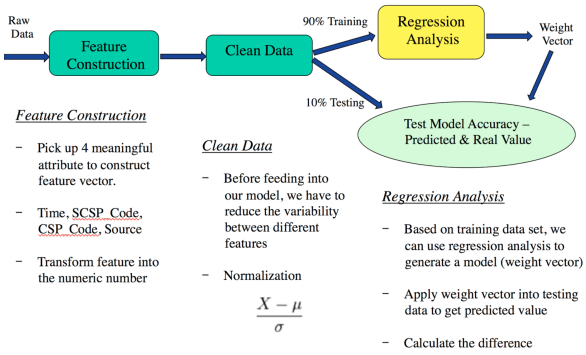


Figure 16: Schematic of Regression Analysis

model. However, if EKO provides very attractive commission rate to them, this will reduce the profit of company. Thus, the goal for EKO is to adjust commission rate to make company earn the maximum profit possible without losing agent and distributors loyalty. Based on this object, we attempt to predict the total transfer amount of each transaction because it has strong connection with commission rate structure. If we can establish the prediction model to predict the money transferring first, we can design the different commission rate structure for different transaction to earn the maximum profit. Figure 16 shows the whole procedure of regression analysis. The detail implementation of our real transaction data will be illustrated in the later result section.

2.3 Optimization

As we continue the exploratory work, we decided to use the gradient descent algorithm. GRG2 is an iterative numerical method that involves "plugging in" trial values for the adjustable vector and observing the results calculated by the constraint vector and the optimum vector. The constraints and the optimum vectors are functions of the adjustable vector. The first derivative of a function measures its rate of change as the input is varied. When there are several values entered, the function has several partial derivatives measuring its rate of change with respect to each of the input values; together, the partial derivatives form a vector called the gradient of the function. Derivatives provide clues as to how the adjustable vector should be varied. For example, if the optimum vector is being maximized and its partial derivative with respect to one adjustable vector is a large positive number, while another partial derivative is near zero, it will probably increase the first adjustable vectors value on the next iteration. A negative partial derivative suggests that the related adjustable vector's value should be varied in the opposite direction.

To apply this technique, we used an excel plug-in named Solver, which approximates the derivatives numerically by moving each adjustable vectors value slightly and observing the rate of change of each constraint cell and the optimum cell. It can use either forward differencing or central differencing. Forward differencing uses a single point that is slightly different from the current point to compute the derivative, while central differencing uses two points in opposite directions. Central differencing is more accurate if the derivative is changing rapidly at the current point, but requires more recalculations. The forward differencing is fine

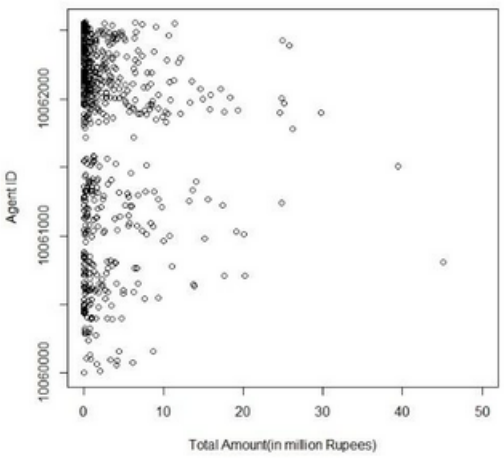


Figure 17: Total Amount Transacted versus Agent ID

in most situations.

The problem with this technique is that some problems may have many locally optimum points where the partial derivatives of the optimum cell are zero. A graph of the optimum cell function in such cases would show many hills and valleys of varying heights and depths. When started at a given set of adjustable cell values, the methods used by Microsoft Excel Solver will tend to converge on a single hilltop or valley floor close to the starting point.

2.4 Clustering

Initially we tried to identify the amounts transacted through EKO by the agent IDs. Unfortunately, the only cluster identified for our dataset was that agents with the largest IDs (most recent ones) contributed badly to total amounts being transferred through EKO compared to the former agents. However, the same cluster was not reflected to the unitary number of transactions being processed by the same agents. To further clarify this finding, we decided to use the J48 algorithm. In order to do that, we had to explore the correlation of our dataset using the correlation plot, but that initiated a major drawback in our dataset, since many of the columns in our dataset were not numeric and even those who were, could not generate a safe conclusion because the numeric values in most cases represented unique IDs.

Manipulating data in a way that it would switch string attributes to numeric sounded like the solution to our case, but that would require a lot of explanation from the EKOs business unit to help us to classify a numeric system that would distinguish our data frame between good and bad values. Furthermore even in the case that we achieved in such a short period to distinguish the numeric values, there would have to be calculated a condition by which data set would be treated to generate a TRUE or FALSE condition. Subsequently, our efforts to generate a condition like that always failed as it was an outcome of existing insufficient features (columns), generating a very short decision tree and providing a low successfully classified instance result (approximately 75%).

2.5 Classification

In the last part of our exploratory work, we attempt to establish a classification model based on the results from pattern analysis, regression analysis, optimization and clustering and use this classification model to define customer and agent's behavior.

Here we propose to use Naive Bayes Classifier[2] as main method to deal with our data and the following equation 1 is the formula of it. Bayes classifier is based on conditional probability theory. As equation shown, we can calculate the probability of class c given the feature x and y. Simply to say, take our transaction data for example, we can feed in some features such as Time or Transaction Source into naive bayes model and determine whether this transaction belongs to money laundry or not.

$$p(c_i|x,y) = \frac{p(x,y|c_i)p(c_i)}{p(x,y)} \quad (1)$$

The advantage of using naive bayes classifier is easy to implement and it doesn't need too much computation. Moreover, it only requires a small amount of training data to estimate the parameters. Because the amount of our dataset is limited, this characteristic of naive bayes classifier is suitable for our case. Most importantly, naive bayes often can get better result in practical use. In the latter result section, we will use naive bayes classifier to establish the classification model on customer changing agent behavior and agent's money laundry behavior.

3. RESULTS

3.1 Pattern Analysis

3.1.1 Customers Behavior on Switching Agents

As described in Section 2.1.2, we analyzed the transaction log to get a picture of customers behavior on switching agents. To get a more formal association rule, we resorted to a well-known association rule mining algorithm, Apriori[1], which is available in R. The library provided by R also gives the users the quality of the rules it found. However, the results are not significant. Although all the quality measurements indicate that no significant rule is found, we can still visualize these rules even they are actually independent to each other (Figure 18 and Figure 19). After all our work, we realized that our features are not relevant to the customers behavior of switching agents, but the silver lining is that, if we look at the agents ID and compare the proportion of the "switchMove" of the agents among their transactions, as in Figure 20. We found that, while some agents only got first customers, without any repeat customers or return customers, some agents could always steal customers from others. We argue that there must be something interesting with these agents. Maybe we can gather more information about them to create more features, like their major occupations, their appearances, star signs, or even their personalities, and well probably find some quality patterns.

3.2 Regression

As discussed in section 2.2, in this section, we will show the real implementation of regression analysis on our transaction data. As Figure 16 shown, before feeding our data into regression model, there are two steps for the data preprocessing. The first step is feature construction. In our

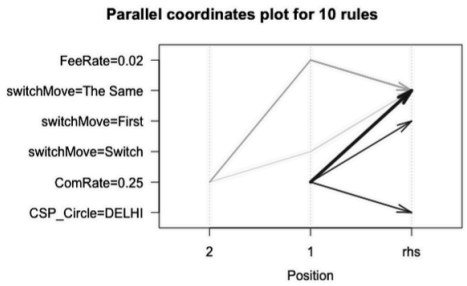


Figure 18: Association Rules Parallel Coordinate

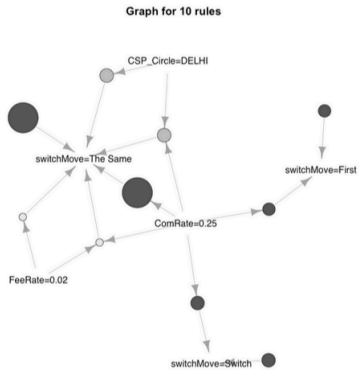


Figure 19: Association Rules Graph

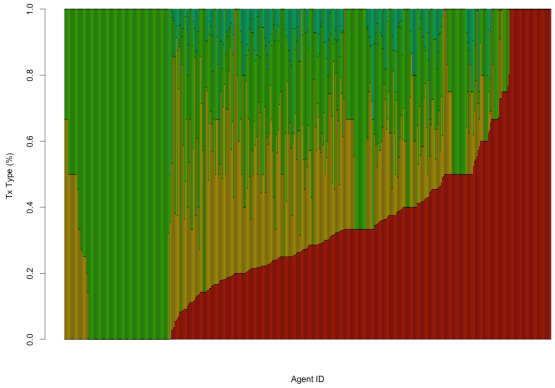


Figure 20: "switchMove" Proportion by Agents

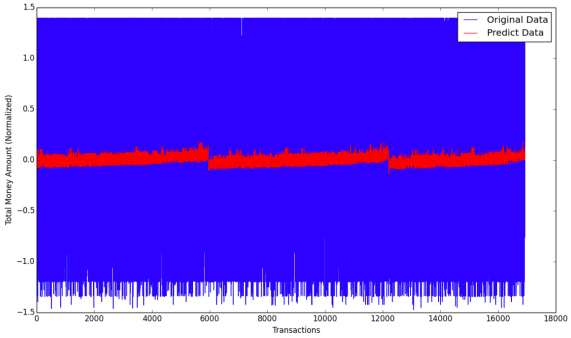


Figure 21: Real Value (Blue) and Predicted Value (Red) for the amount of money transferring of each transaction

transaction data, not all the features are useful for predicting the amount of money transferring and we have to only extract the meaningful ones. In this model, we choose *Time*, *SCSP-CODE (distributor's code)*, *CSP-CODE (agent's code)* and *SOURCE* to be our input features. Next step is to clean our data. Because the variation between different features is large, we have to diminish this effect to improve the prediction accuracy. Here we just use simple statistical way which is minus mean and divide into standard deviation to normalize them.

After data pre-processing, we divide our dataset into two groups: 90% for training and 10% for testing. Next we use training group to train our regression model which is constituted by the vector of weights and then apply our testing data into this model so that we can get the predicted value. Figure 21 shows the result between observed value and predicted value on total money transfer. From the result we know this prediction model doesn't work (the root mean square error (RMSE) is very large). That is to say, it is hard to predict money transfer of each transaction only based on our current input features which are not so relevant to the predict target. If We can use more diverse features to train our model, we can expect to get better prediction performance on this model.

However, there is one valuable thing from this regression analysis. We investigate the generated weight vector: *[0.02498, -0.0044, -0.01293, 0.03374]*, which means the first (Transaction Time) and fourth features (Transaction Source) are more relevant to our predicted value. This can give us a hint for the future work : we can design different commission rate structures based on customer transaction time and the source they used.

3.3 Optimization

In order to solve the locally optimum points issue, external knowledge of the problem is needed. Either through common sense reasoning through experimentation, you must determine the general region in which the global optimum.

In EKO's case, we defined that the ideal transaction mix visualization to optimize revenue and profitability such that:

- The same number of transactions yield highest revenue ***or*** profitability.
- The least number of transactions that yield same revenue ***or*** profitability.

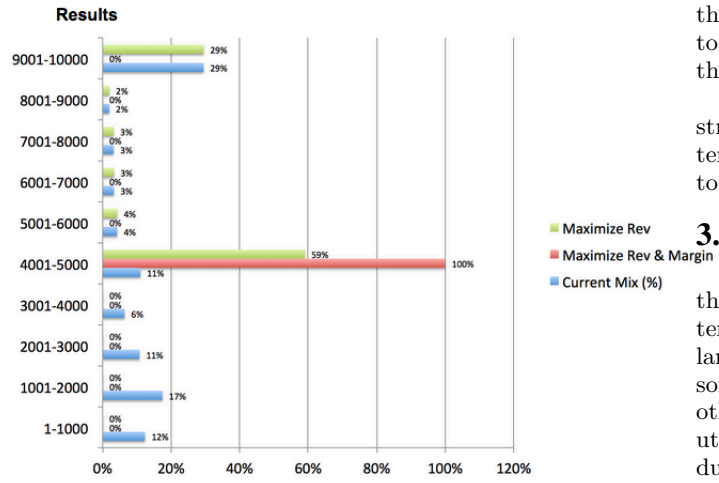


Figure 22: Transaction Amount Range versus Optimization Rules

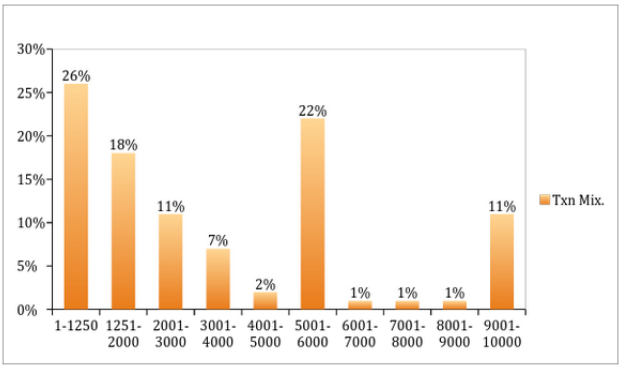


Figure 23: Transaction Amount Range versus Small values Category

- Maximize Revenue and Profitability at same number of transactions.
- Profitability is fixed at 1.5%, revenue can vary, but transactions constant.

For Figure 22 The results were as follows:

- The graph showed that at Rs. 5000 the profitability and Revenue moth maximize, therefore in the 1st case, all transactions should be of value 5000, which is practically not possible.
- Since revenue above Rs. 5000 is same anyway and the transaction mix from 5000-10000 is maintained, it suggested that rest all transactions should be abandoned, which again is not a possibility.
- The 3rd scenario did not yield a meaningful result probably because the scenario 1 already covers that.
- Scenario 4 yielded that following results.

In Figure 23, it is clear that transaction of Rs. 5000 is the most critical one and more the transactions in the small value category, higher will be the profitability. Currently the ratio of 5000:10000 transactions is 1:1, however, it needs to change to at least 2:1 for the company to move closer to the higher profitability and margins.

This method yielded a wonderful way of providing constraints and hence maximizing the desired vector. EKO intends to use this method to experiment with agent pricing to yield a desired transaction mix.

3.4 Clustering

As discussed under section 2.11, the clustering analysis of the transaction network allows us to visually discover patterns of customer behavior and agent and distributor popularity. From all three transaction networks, there are clearly some agents and distributors who are more popular than others. Identifying the top performing agents and distributors will help EKO devise strategies to improve the productivity of their agents and distributors. In addition, we find that most of the customers are loyal to their agents, which further highlights the importance of improving agent productivity for achieving maximum retention of repeat customers. However, there are a few repeat customers who are connected to more than one agent, which means they have made transactions through different agents and hence exhibit agent-switching behavior. Further analysis will investigate what makes these repeat customers switch to different agents.

3.5 Classification

In the last part, we summarize the analysis results from previous section and attempt to use Naive Bayes Classifier to establish classification model. We make the concentration on the following two behaviors: customer will change agent or not and agent's money laundry behavior.

The first case is to make classification on customer will change agent or not. Like regression analysis, before feeding our data into naive bayes classifier to train the model, we have to make feature construction. As Figure 24 shown the features and predicted target, here we use a technique to map our feature from continuous into different categories. Take the feature of Time for example, if the transaction happened during 0AM to 6AM, it will be assigned with the numerical value "1" and 6AM to 12PM will be assigned "2" and so on. Through this process, we can improve the classification rate in the final result. After mapping the data, again, we divide our dataset into two groups: 90% for training and 10% for testing. Use training data to train our naive bayes classifier model and apply the testing data into this model.The final result shows that this model can achieve up to 95% classification accuracy for detecting customer will change agent or not in the future. That means behavior of changing agent can be well predicted and also can be used as reference for creating new products in the future.

The second case is to detect the money laundry potential of the agent. The dataset including features is similar to the Figure 24 and the only difference is the predicted target. The predicted value here is the frequency of agent's account appear in the same month. The detail definition of money laundry behavior has been defined in the previous section. Based on this analysis, we also can use naive bayes classifier to classify this behavior. Here we map the predicted target "frequency" into three level: value is lower than 5 (low frequency), value is between 5 to 10 (medium frequency) and value is higher than 10 (high frequency). Again, we

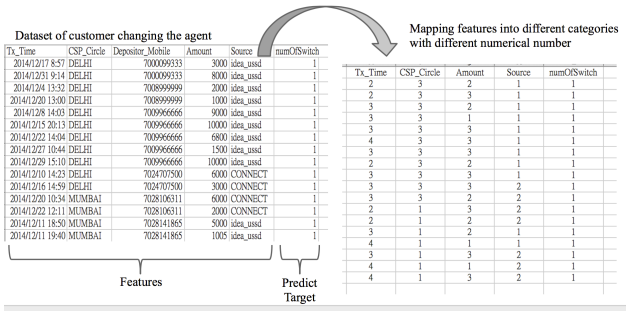


Figure 24: Example of feature mapping on real transaction data

map the target into three categories because we want to improve the classification rate of our model. After constructing the features, the other steps are similar with the first case. However, the final result shows that this model can only have 10.53% classification accuracy. The feasible problem is that there is a skewness on our dataset. Specifically, for most of our data, the the "frequency" term belongs to lower level which means our dataset is not so diverse to establish a classification model. If we want to accurately classify money laundry behavior in the future, we have to collect more date to diversify our training pool.

4. CONCLUSIONS

Our work identified some key points that will help EKO better serve their customers. The clustering analysis showed that we did not had enough data nor features to classify the users. If EKO wants to better understand their behavior, it will need to expand the data that it gathers from their customers. Some potential fraudulent behavior was detected by correlating transaction mount, account and mobile numbers, but this was only possible by manual inspection, which does not gives us any further insights based solely on the data. Nevertheless, classification models showed to be fruitful, where we achieved a 95% accuracy when predicting if a user will change Agents. This is valuable for EKO so it can better classify its users and draw strategies to handle agents.

Further work can be done in the fraudulent analysis and customer behavior, but our findings show that a much larger dataset would be necessary, or additional features so we can train the models properly.

5. ADDITIONAL AUTHORS

Five additional authors:
Chih-Feng Lin (*chih.feng.lin@sv.cmu.edu*)
Pyiush Gupta (*pyiush.gupta@west.cmu.edu*)
Na Li (*na.li@sv.cmu.edu*)
Ning Du (*ning.du@sv.cmu.edu*)
Jeremiah Lin (*jeremiah.lin@sv.cmu.edu*)

6. REFERENCES

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, pages 487–499. IBM Almaden Research Center, September 1994.

[2] K. P. Murphy. *Machine Learning - A Probabilistic Perspective*. The MIT Press, Cambridge, Massachusetts, 2012.