Exploration and Analysis of Retail-Banking Transaction Data

Pantelis Zoumpoulidis
Department of Engineering Mathematics
University of Bristol
Bristol, BS8 1UB, UK
pantelis.zoumpoulidis.2020@bristol.ac.uk

****
Department of Engineering Mathematics
University of Bristol
Bristol, BS8 1UB, UK
****

****
Department of Engineering Mathematics
University of Bristol
Bristol, BS8 1UB, UK
****

***Abstract*—Provided with synthetic transaction data from Lloyd's Bank, we aim to extract useful information for the bank. We architect our analysis under three main topics: customer, store, and customer-store insights. As this data is brand new and previously unseen, we begin by carrying out rigorous exploratory data analysis on the raw data before pre-processing and cleaning the data to formats useful for subsequent data science-oriented tasks. Using unsupervised and supervised methods for clustering and regression, we formulate and visualize insights directly relevant to LBG's business interests. These insights range from customer spending and risk clustering to customer-store correlation coefficient analysis. We present our methodology, implementation, and results for each particular topic grouped in sequence and end our report with ideas for future research.**

***Keywords*—transaction, customer, risk, store**

I. INTRODUCTION

The power of the internet is most significant than ever, and people choose to use it to buy goods and services. The lockdown caused by the pandemic enhanced the frequency of this behavior even more. The people who prefer to go to a local shop instead will probably choose to pay with their card or smartphone rather than cash. All these transactions, along with their time details, are recorded by the consumer's bank, creating a vast database of trades for every customer. This paper tries to address how a bank can take advantage of all these data and extract valuable intelligence out of them.

II. DATASETS

Due to confidentiality reasons, the datasets we use in this problem are generated from a data synthesizer Lloyds Banking Group (LBG) created in 2020, which simulates real-life transactions. Thus, the results demonstrated in the subsequent sections most probably apply in realistic conditions.

Dr. Marie Anderson, the chief of data office of LBG in Bristol, provided us with a couple of datasets with a different amount of complexity. The first one consists of monthly data for nine months of a year. It includes four distinctive characteristics: the bank account number from which a transaction is made, the quantity of money involved, the name of the account where the money went, and the day of that transaction. The second dataset contains transactions for the first quarter of 2020. It consists of richer information with seven features in total, giving us the ability to be more creative. These features are the following: the bank account number from which a transaction took place, the amount of money involved, the balance of the bank account after that transaction, the account number where the money transferred, the name of that account's holder, and finally, the date and the time of the occurred transaction.

III. PRE-PROCESSING

A. Dataset A

We begin by breaking down the pre-processing steps in the first dataset. The primarily pre-processing step was to concatenate all nine months of data into one dataset. That enabled us to find all distinct values for every feature involved and made it easier to analyze the data. Next, we searched and removed rows with missing data. The total number of transactions in the final dataset was 13,234,863. After that, we added a new column that corresponded to the weekday of the data. Afterward, we studied the 79 different names of the target accounts linked to a specific type of business or a bank account. Using text analytics tools, we identified and corrected misspelled words, such as *jewllery* or *seafood resaurant*; we also did the same thing for the value butcher, which was in both singular and plural. Then, we created two more features in the dataset with more generic types for these values, i.e., A_LOCAL_COFFEE_SHOP → Coffee Shop → Food & Drinks.

To store and distribute this large dataset, we created a relational database on the AWS cloud coupled with python scripts for receiving and uploading new datasets or updates. That allowed the data to be rapidly imported without the need to keep versions of CSV files. Tableau also has support from pulling from remote databases.

B. Dataset B

Following this, we analyze the pre-processing phases in the second dataset. Here, all data are in one file, so we do not have to merge them into one dataset manually. We tried to find inconsistencies in the transactions, but we were not able to. We checked for empty cells in the columns, where the only missing values we found were the account names for bank account numbers and the account numbers for two specific account names, Deliveroo and Halifax. In the first case, we filled the account name as "Bank Account" because every account number was corresponding to a specific customer in the bank and not a business, while in the latter, we used the account names as the account numbers. Even though the second transformation seems wrong, it was valuable because we treated every account number as a string and not an integer. Furthermore, we confirmed there were the opposite transactions when a customer was transferring money. Moreover, we discovered that every third-party account name was linked to a real-life business. Hence, we used Google to find every shop's type and added it as a new feature column; for example, "A Yarn Story" is an "Arts & Crafts Shop" in Bath.

Lastly, we discovered some strange sequential transactions when we were studying the dataset. Therefore, we developed an algorithm that detects frauds inside the dataset. The function takes a window of time and a given number of transactions as configuration. Then for each customer's account in the dataset, it searched for consecutive transactions to a specific account number. After searching completion, it returned for every holder's account the transactions that met the given conditions. To be more precise on fraud detection, we also let the algorithm keep examining for entries to the same account number after the time span if it already met both conditions. For instance, if we set the inputs to ten transactions in 1 minute, and there were ten transactions in 1 minute to a specific account number, and then three more similar in the next minute, all 13 transactions would be included in the frauds list, even if the three transactions alone did not meet the conditions.

To reduce the noise in our data, we used the fraud detection algorithm we developed with the parameters 15 minutes and 20 transactions and removed the frauds from the dataset. In total, 48 customers' accounts were affected, with most of the money transferred to Deliveroo and Halifax. Our final dataset consists of 151.487 transactions.

IV. ANALYSIS

A. *Customer Analysis*

1) *Customer Risk Clustering*

Since LBG is in the loan business, it is practical to avoid bad debt losses by identifying customer risk levels. What we tried to do in this section was to divide customers into groups based on their risk levels. Since there is no risk level label in the original dataset, clustering is the best technique for this cause. The clustering algorithm we used was k-means, suitable for situations where the data points are in the plane and the number of clusters is small. Furthermore, to confirm the optimal number of clusters, we used the elbow method.

For Dataset A, because the customer's income and expenditure could reflect the customer's debt repayment ability and consumption ability, we used the income and expenditure of each account as the input for clustering. We assumed that when the customer had high income and low expenditure, his risk was considered low. In contrast, when the customer had low income and high expenditure, the customer's risk was high. While this assumption is not complete, it is reliable to a certain extent as the accounts' balance is not provided in Dataset A. After the first attempt, we found that the results of direct clustering were very unsatisfactory. In the image obtained by direct clustering, most of the data points are compressed on one side of the image, making it challenging to draw concrete conclusions. Due to these findings, we decided to separately cluster the customers with a positive net income and a negative net income, as the distribution density of their data is different. After the data segmentation, we finally got a decent clustering result. The yellow clusters on the image below represent high-risk customers who spend more but have low repayment ability. The light blue clusters represent regular customer groups with partial repayment capabilities, although they have high expenditures. Fig. 2 also presents different clusters. Since their net income is positive, we consider these customers to be low risk, whereas the customers in the red cluster have more significant consumption potential.

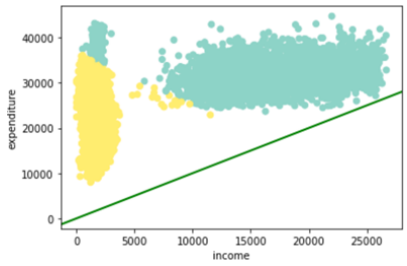

Fig. 1. Customer risk clustering result (net income < 0)

Table 1. Customer spending tally

Account Number	Restaurant	Clothes Shop	…	Pub
1088	412.44	0.00	…	49.80

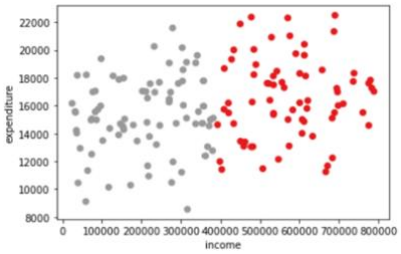

Fig. 2. Customer risk clustering result (net income > 0)

Next, we applied the same clustering algorithm in Dataset B. Nevertheless, the main difference was the use of customers' net income and balance as the inputs. Net income reflects the current repayment ability, while the customers' balance reflects their standby repayment ability. In other words, the balance represents the spare repayment ability when a customer's income is zero. Compared to the clustering in Dataset A, these inputs can lead to a more comprehensive reflection of the customers' risk level. As shown in the following figure, green clusters represent low-risk customers. Even though some customers in this cluster have a low net income, they are still considered low-risk as their deposits' amount is significant. Yellow clusters symbolize regular customers since their net income and savings are roughly zero. Finally, red clusters and gray clusters signify two different types of high-risk customers. Red clusters correspond to customers with high debt, while gray clusters customers with low repayment ability.

By clustering these customers, we objectively classified their risk levels. LBG can set different credit lines according to the customers' risk level to avoid bad debts.

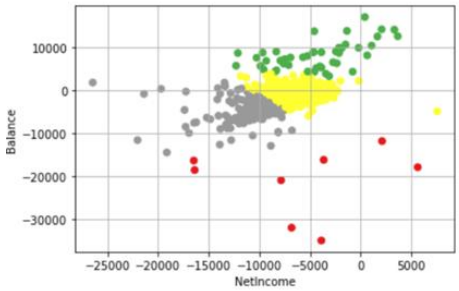

Fig. 3. Customer risk clustering result

2) *Customer Spending Clustering*

a) Reasoning

The next aspect of the customers' that we explored was whether we could obtain "types" of customers based on their spending habits. This should be useful to LBG because it would allow it to gain a better understanding of its customers' spending habits. That could help LBG to build profiles of typical customers, which could aid them in many ways. For example, one use of these profiles could be to better target customers using advertisements. Types of customers inferred from their spending could also be tied to other characteristics such as TV viewing preferences or the type of holidays they prefer to go on.

b) Implementation

Using the 15 store category labels that placed each store into a general bin of similar stores (discussed in Section 3), we calculated the sums of money that each customer had spent at each type of store over the entire collection of transactions. See the example row below (with some columns omitted).

The total spending of each customer was also calculated, and after inspecting the distribution of total expenditure, we

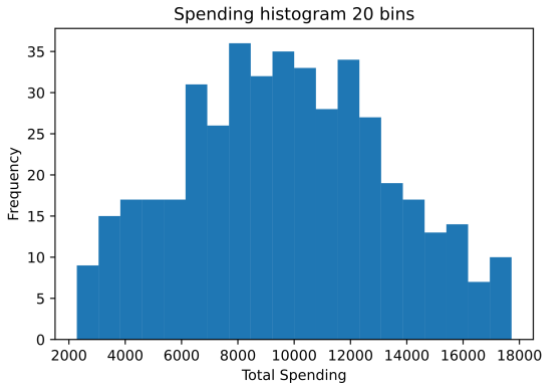

Fig. 5. Customer spending histogram

found there were several outliers that spent far more than the average customer, and these heavy spenders were removed to reduce skew from the results. After reducing the customers down to the 90th percentile, the distribution was roughly Gaussian (see Figure 4).

The next pre-processing step was to create several versions of the data:

A) Original data - the net spending.
B) Proportional data – which converted the net spending to proportions of the customers' total spending.
C) Normalized data – which took the mean spending over all the categories away from each column and then divided each by the standard deviation between all the categories.
D) Normalized proportional – which applied the normalization step in C to the proportional data.

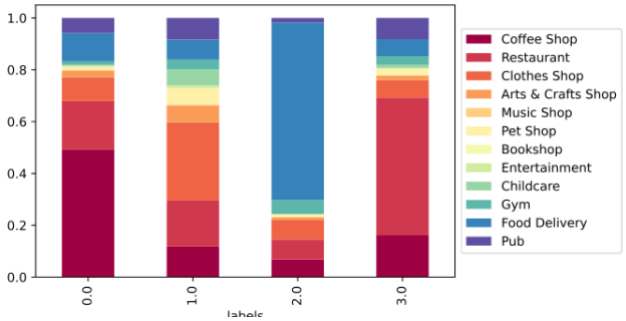

Fig. 4. Customer cluster spending comparison

Our approach to finding types of customers based on spending was to compare the results from both K means clustering (KM) and Gaussian Mixture Models (GMM). Principal component analysis was also used to reduce the components from 15 to between 2 and 7. Elbow plots from each of the datasets suggested that between 3-5 clusters would be sensible, although, for the normalized data, the elbow plot was almost a perfect diagonal. This suggested that the normalized data would not cluster very effectively, and after further investigation, the normalized dataset was dropped. After an initial round of testing, we found that the Supermarket, Bank Account, and Banking categories were being focused on heavily by the algorithms when creating clusters, but the resulting plots did not provide even or distinct clusters within any of the other components. For this reason, we removed the Supermarket, Bank Account, and Banking categories from the data. We then performed testing of all combinations of:

A) Clustering algorithms (KM, GMM)
B) PCA components (2-7, including not using PCA)
C) Datasets (Original, Norm, Prop, Norm prop)
D) Number of clusters (3-5)

This produced too many results to assess each by eye, so we used Silhouette Score – the mean of the Silhouette Coefficient of all samples. Scores range from the best value 1 to the worse -1. Values near 0 indicate that the clusters overlap. The average scores over each dataset are presented below.

Table 3. Silhouette Scores

Dataset	Average Silhouette Score
Original	0.432
Normalized Proportional	0.142
Proportional	0.204

Table 2. Customer cluster spending proportions

Cluster	Coffee	Rest	Cloth	Arts	Music	Pet	Book	Ent	Child	Gym	Food Delivery	Pub	No. of custom
0	0.49	0.19	0.09	0.03	0.0	0.01	0.0	0.01	0.0	0.01	0.11	0.06	150
1	0.12	0.18	0.3	0.07	0.0	0.06	0.0	0.01	0.06	0.04	0.08	0.08	145
2	0.07	0.08	0.07	0.01	0.0	0.01	0.0	0.0	0.0	0.05	0.69	0.02	28
3	0.16	0.53	0.07	0.02	0.0	0.03	0.0	0.0	0.01	0.03	0.07	0.08	114

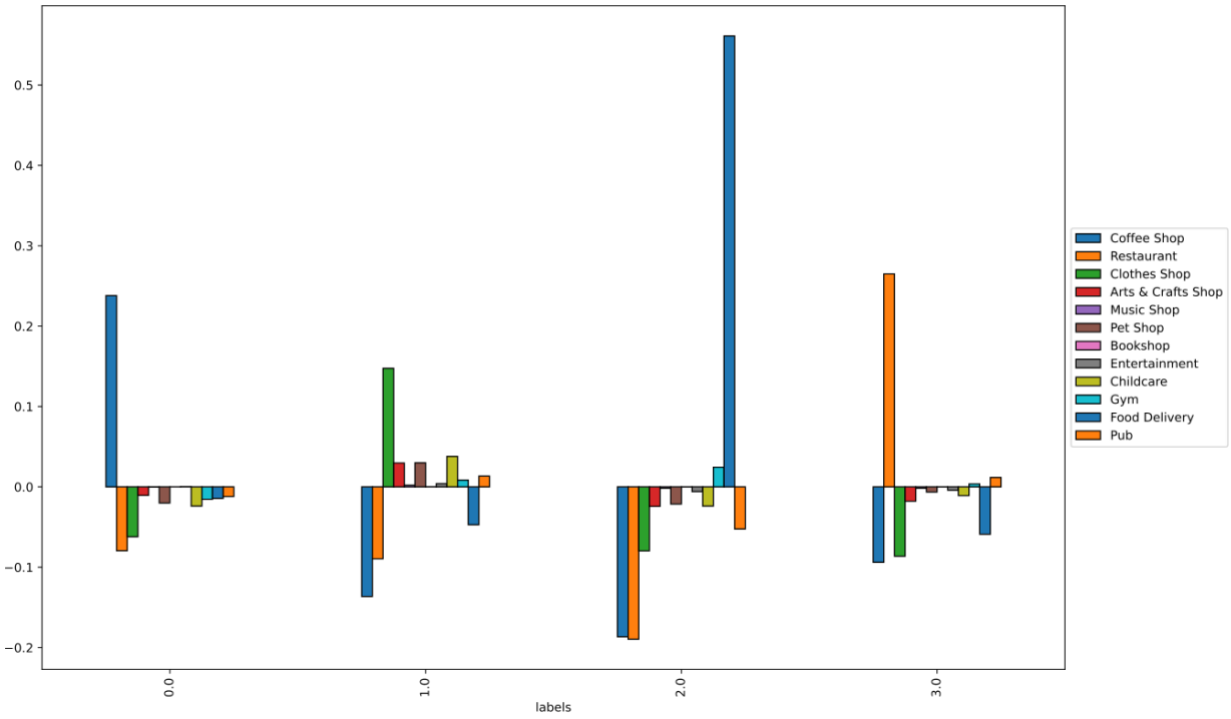

Fig. 6. Mean subtracted cluster spending proportions

While the results suggested that the original dataset produced the most promising results, the silhouette score was later found not always to represent the best visual clusters. However, the silhouette score did help us to find the best-looking clusters when we investigated the highest-scoring proportional results. The configuration found to give the best results was with the cluster labels obtained from the proportional dataset using K means clustering of the components calculated from the PCA algorithm with the number of components set to 3.

c) Results

Figure 5 below is the bar chart (and supporting table 6) showing the average proportion of spending on each of the store categories by each cluster of customers. Also included is a plot of results with the mean for each category subtracted (Figure 6).

Type0 spends significantly more on coffee than the others, while the childcare spending of 0 suggests that they do not have children (or perhaps need childcare?) and do not go to the gym.

Type1 spends significantly more on clothes, childcare, pets, and arts and crafts. Intuitively this sounds like an older, possibly more family-oriented customer.

Type2 spends on average well over half their money on food deliveries. This knowledge, paired with the much lower count of only 28 customers, suggests that this type could possibly be due to noise or errors in the data.

Type3 spends much of their earnings on eating out in restaurants and pubs.

3) Spending Regression

It is essential for LBG to have projections of future cash flow to ensure they have the money for future investments, to give their economists information on the future state of the economy, and to monitor their customers' likelihood to be in debt, among other things. For these reasons, we imaged a system where the bank could input transactional data up to the present point in time to a machine learning model, and the model could predict the spending over the next X period in the future. Due to the limitations on time and data with our project, we decided to simplify the overall problem to predicting weekend spending from weekday spending.

To do this, the transactions from a pre-processed version (mentioned in Section 3) of dataset B were further processed to create a table of customer spending by week along with some extra features. The weekly spending was split into 12-hour frames, each having a feature within the dataset. The extra features consisted of: Week of the year, the customers' spending cluster (from Section 4.A.2), and the estimated salary. The label we were trying to predict was the total weekend spending. An example row is shown below in Table 4.

We created and tested five models:

1) A benchmark regressor

Simply predicted the mean weekend total of the test set for every prediction.

2) Ridge regression

A simple linear method with L2 regularisation to combat overfitting by reducing the magnitude of the weights.

3) A multi-layered perception (MLP)

Table 4. Example feature vector for regression

Account number	Monday AM	Monday PM	…	Friday PM	Expected Salary	Week of the year	Spending cluster	Weekend total
1000	-27.91	-24.91		-127.33	1540	4	1	-162.97

A basic neural network with a single layer paired with a non-linear activation function. This model can learn non-linear relationships.

4) A linear regression model built using tf.Keras

Another linear method but using normalized data with no regularization.

5) A deep neural network (DNN) built using tf.Keras

A neural network with multiple layers. We tested this model with L2 regularization, dropout, and different input layer sizes.

After trying many combinations of features, we obtained the best results using only the summations of each day, resulting in only five features together with the label Weekend total. Table 5 below contains the results.

Table 5. Model Scores

	Root mean squared error (RMSE)
Benchmark	446.68
Ridge regression	445.10
MLP	445.11
Linear regression	392.23
DNN	394.17

While the results show that our current models were relatively ineffective in predicting weekend spending, there are some takeaways. The results suggest that the data may not be in the best format for this task. Additionally, the amount of data is likely to be too small, at least for the neural network architectures that can require a lot of data to learn effectively. The underlying relationship between the features and the labels is not linear, or we would expect a more considerable improvement from the benchmark to the linear methods.

B. Store Analysis

1) Store transaction pattern

In this section, we studied the transactions of each store, which can help LBG cooperate with efficient businesses to obtain more deposits. In addition, we analyzed the transactions of various stores in Dataset B with different time granularities. Discovering transaction patterns is a win-win situation for both LBG and the business customers of the bank. LBG can launch joint promotions with these businesses based on the customers' transaction patterns. For example, during the Christmas period, where people spend more money than the rest of the year as we will show later, LBG can offer these consumers discounts when paying with an LBG bank card at partner stores. The most significant advantage of this promotion is that the stores will see an increased transaction volume and profits, while the total amount of deposits in LBG will not decrease. Since both parties of the transactions will be LBG's customers, the money will only transfer within the bank. All the previous analysis was made in Tableau, as it can visually display results without the need for complex code.

First, to understand the businesses' operating status, we considered the transaction volume of each of them as an essential metric. We sorted all store types based on the number of transactions, and then we did the same for the stores of each category. In the first dataset, bar, pub, and supermarket are the three store types with the biggest turnover, while "Pub", "Bar", and "Express_Supermarket" are the stores with the most significant revenue in each of these categories. A large number of transactions will result in a large number of deposits, which is why LBG should cooperate with these businesses.

To better comprehend the transaction patterns of the stores, we performed two different analyses in each dataset. Since we had a larger time span in Dataset A, we studied the changes on a monthly basis. We concluded that there would be a significant increase in the transaction volumes for most of the store categories in December. Therefore, we believe that LBG can launch special promotions in December to increase the utilization rate of credit cards.

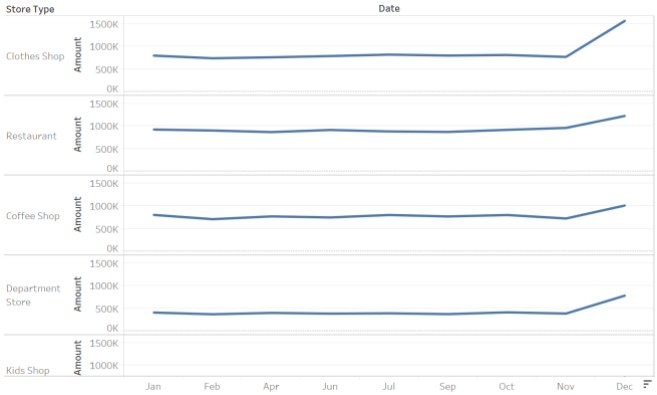

Fig. 6. Monthly transaction pattern

On the other hand, we used daily and hourly granularity to analyze transaction patterns in Dataset B. Take "Food" and "Drink" store categories as an example. In the daily analysis, we discovered that customers tend to go to restaurants in the middle of the week, while they incline to buy drinks on weekends or Mondays. While analyzing hourly, we found that customers will mainly buy drinks from a coffee shop from 8 am to 11 am, whereas they will most likely get food from a takeaway platform three different times of a day; 7 am, 1 pm, and 6 pm. We also examined the transaction patterns of retail stores and supermarkets. The conclusion is that apart from weekends, retail stores will also have more transactions on Wednesdays. Additionally, people are more accustomed to shopping in these retail stores around 9 am and 6 pm. Also, supermarkets have fewer transactions the first three days of the week and more the last four days. During the day, people mostly tend to go to the supermarket at 8 am or 8 pm.

After we analyzed all the data, we got two pieces of information with commercial value. In retail stores, people are also willing to spend on Wednesdays. Besides that, many people will order takeaways even in the morning. Therefore, we recommend LBG launch joint promotional activities with other businesses, giving the opportunity to the customers to get a discount when they use their Lloyd's credit card on Wednesdays. We believe this would help the bank increase its transaction volume.

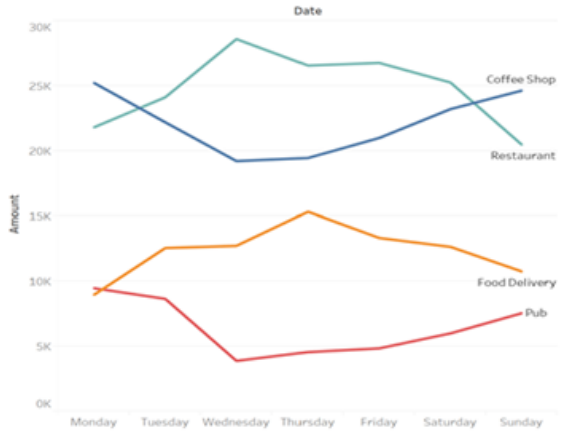

Fig. 7. Weekly transaction pattern

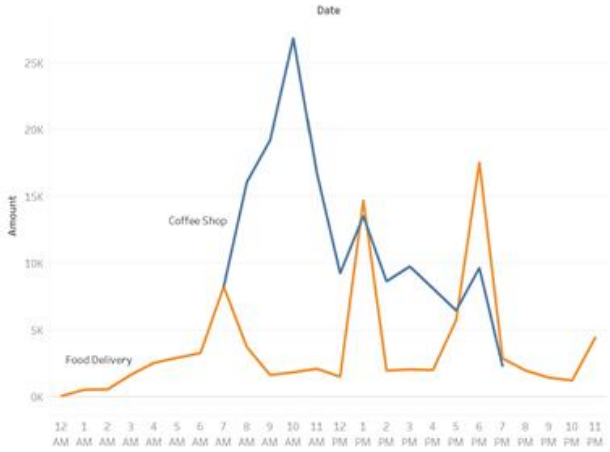

Fig. 8. Daily transaction pattern

C. Customer-store Analysis

1) Correlation coefficient analysis

The recommendation system inspires the analysis of the correlation coefficient. The function of the recommendation system is to help customers discover new potential needs, and what we hope to do is help LBG discover possible consumer behaviors. In Dataset A, we combined the previous customer analysis results with the store analysis results. Based on the previous cluster analysis, we found a list of low-risk customers. These customers have high net income and great consumption potential. On behalf of LBG, we hope that these customers can use their bank cards to make purchases so that the bank can get a commission during the transaction. Combined with the transaction records in the store analysis, we obtained the transaction records of all potential customers. After importing the data into Tableau, we wrote formulas to obtain store correlation coefficients for potential consumers. Fig. 7 presents the visualization image of the correlation coefficient we obtained.

These correlation coefficients show the preferences of consumers in a particular type of store. The greater the color difference in a column, the more apparent consumers' preferences are in that store type. For example, takeaway consumers are strongly related to "lunch", "DVD shop", and "bookshop". The correlation with the "lunch" category reaches 96.5%, which means that most takeaway consumers have a record of consumption in a "lunch" business. Therefore, LBG can launch a special discount for "lunch" stores in order to promote potential customer consumption.

In addition to discovering customers' behavior in different store groups, we also studied their behavior in the same categories. As we can notice from the following correlation coefficients table of "takeaway" businesses, there is a strong relationship between "Sandwich_shop" and "Kebab_shop". That means that recommending the second store to customers who have only visited the first, will most probably persuade them to buy also from that store. In this way, LBG can also promote customer consumption.

	CHINESE_TAKEA..	KEBAB_SHOP	SANDWICH_SHOP	TAKEAWAY	TAKEAWAY_CUR..
CHINESE_TAKEAWAY	1.00000	0.98038	0.98058	0.90562	0.90716
KEBAB_SHOP	0.98038	1.00000	0.99799	0.98089	0.98091
SANDWICH_SHOP	0.98058	0.99799	1.00000	0.98046	0.98009
TAKEAWAY	0.90562	0.98089	0.98046	1.00000	0.90447
TAKEAWAY_CURRY	0.90716	0.98091	0.98009	0.90447	1.00000

Fig. 11. Correlation coefficient of takeaway stores

V. MINOR WORK

A. Relationships Finder

Motivated by the fraud detection process, we developed an almost opposite algorithm that finds possible relationships of customers using three features; an hour window of the transactions, the minimum number of transactions of the possible friends, and an hour window for the possible friends to transfer the money. To understand how it works, we present the following example with the parameters 1, 2, and 0.5: assume customer A with account number 1 who spends £20 on a pub at 16:00. If in the next hour (until 17:00), there are at least two transactions that take place in a maximum period of half an hour, transferring money back to account

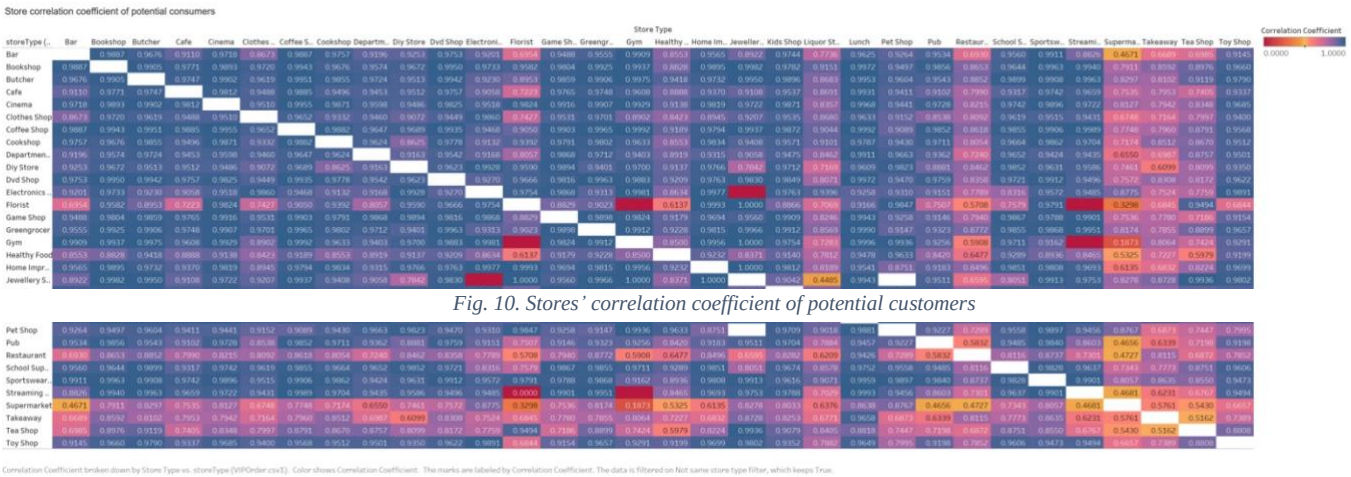

Fig. 10. Stores' correlation coefficient of potential customers

number 1 with the total sum of two to be less than £20, they are likely closely related to customer A.

There are two ways a bank can use that kind of intelligence. The first is to share this information with the bank's marketing team along with the spending habits of these customers, which in return can generate specialized advertisements or deals for them. A second technique is through the bank's mobile app. After a transaction, using Bluetooth or Wi-Fi technology, the app can check if closely related people are next to the customer. Then, it can send a notification to these people if they want to split the bill with just a tap. That convenience would maximize the user experience, and consequently, the customers' gratification with the bank's services.

B. Account Balance Estimator

While searching for patterns inside the second dataset, we came across some transactions that caught our attention. There were bank accounts that earned money from specific account numbers that were related to businesses. These transactions were happening either the first or the last of a month, for every month. That clearly indicates that that business employs the account holder who gets this money, which is his salary. Hence, we developed an algorithm that predicts the balance of the bank accounts at the beginning of every month, using the average of the previous months' salaries. Note that if a bank customer earns his salary at the end of the month, we do not add it to his balance. There was room for performance improvement by using machine learning algorithms to predict the salary. However, due to the small number of months we had in our data, we could not implement one.

The logic behind this idea is that a bank needs to have an insight into the cash flow at the beginning of an upcoming month, so it adapts its policies and investments accordingly.

C. Bank Customers Classification

During our experimentation with Dataset A, we thought it would be fundamental for a bank to identify how its customers will perform using half of the data, meaning six months in a year. Therefore, we added the total spending and earning for each bank account, for the three missing months data, by taking the average of $n-1$ and $n+1$ months. Then, using all twelve months' sum of income and expenditure for each account, we calculated the end-of-the-year result percentage for all of them as:

$$ResultPercentage_x = \sum_i^{N=12} \frac{earnings_{x_i} - spendings_{x_i}}{earnings_{x_i}}$$

, where x is each account number, N is the total number of months, $earnings_{x_i}$ is the total of earning of account x for month i, $spendings_{x_i}$ is the total of spending of account x for month i. When the value of $earnings_{x_i}$ we simply added $-\infty$ for that bank account.

Following this, we added eight different labels for a given span of percentages. Afterward, we removed the last six months from our dataset, split it into training and test sets, and trained a k-Nearest Neighbors classification algorithm. We performed cross-validation in the training set, finding that eight neighbors returned the most significant accuracy. Lastly, we used that value of our tuned hyperparameter to make predictions on the dataset, achieving a total of 84.7% accuracy.

Even though this number seems quite astonishing, we decided not to continue further with this method, as most of the bank accounts had an income of zero or close to zero, and measuring the missing months' earnings and spendings with just an average added much bias in our dataset.

VI. FUTURE WORK

Due to the limited time at our disposal, we were not able to implement all of our ideas. Though, if the circumstances were different, we would do the following:

For the prediction model, we did not find a suitable prediction perspective. Therefore, we would improve our Neural Network model we built to provide better results. Furthermore, we would use Time Series algorithms, like ARIMA, to forecast the values of the missing or forthcoming months of our datasets. Both studies could help LBG grasp customer consumption trends in advance and find appropriate profit strategies.

In terms of coefficient analysis, due to the limitation of data entries in Dataset B, the user preferences we got were not clear enough because of the low number of transactions for some stores. In future work, we would analyze the correlation coefficient more extensively. Ultimately, we would achieve the same effect as the recommendation system of Dataset A, where we determined which users are likely to spend in a particular store. That would add valuable intelligence to the marketing team of LBG, which then could perform targeted promotion strategies to its customers.

Finally, we would consider optimization involving data science ethics. We were able to discover some of the customers' privacy details from simple transaction records and understand each business's current financial conditions. We are confident that risk analysis can tell us whether to give a loan to the businesses or customers, but we need to continue studying how these analysis results are fed back to the bank.

VII. CONCLUSIONS

Overall, the content in the report has reference value. First, through text tokenization, missing value filling, and fraud detection algorithms (abnormal data detection), we have obtained a dataset with low noise and high credibility. These data pre-processing methods have a certain degree of reusability, and we think this can help LBG better complete its data cleaning work.

Secondly, in the central part of the report, we analyzed the pre-processed data from the perspective of customers, stores, and the combination of these. From the customers' point of view, we mainly completed the following tasks:

a) We used the KM algorithm to cluster risks and classify customers into different risk levels according to their financial status and daily expenditures.

b) We tested GMM and KM clustering on consumption data, including after applying PCA dimensionality reduction, and finally divided customers into different consumer groups based on their consumption in various stores.

c) We applied supervised regression techniques taking as input the consumers' workday expenditure and predicting weekend spending. We tried several models of varying complexity and learning capabilities and found that the best results were from a linear model. However, the performance

of this solution was still poor. Suggesting that the data was not set up in a suitable fashion or that the underlying relationship was too complex for our models (or likely, a combination of both)

Our main tasks in terms of store analysis are presented below:

a) Through the data aggregation method, we roughly estimated the operating conditions of different stores and visualized the results.

b) We studied the transaction patterns of the stores at different time granularities and drawn regular conclusions.

Lastly, on the customer-store aspect, we combined the risk level results with the store aggregation results and discovered part of the consumption preferences of low-risk customers through correlation coefficient analysis.

In the real world, the significance of our analysis work is as follows: first, the customer risk level we divided can help LBG confirm the customers' credit limit and loan amount and reduce their financial risk. Secondly, the consumption clustering and correlation coefficient analysis we completed can help LBG discover potential consumption behaviors and guide customers to make more use of their Lloyds bank cards as well as provide customer profiling potential for Lloyds to make use in advertisement campaigns, among other things. Finally, LBG can benefit by finding suitable partners and bringing them more deposits based on our research on store transaction models.

Lastly, in addition to these realistic research results, we also made some creative attempts, such as finding a relationship between bank accounts through short-term transaction behaviors and predicting the balance of the accounts in forthcoming months depended on their salary.

REFERENCES

All source code used in this project is available at:
https://github.com/CyrusDobbs/DS-mini-project