Merchant Churn Analysis

Sai Pavani Cheruku

Part A: Understanding payments activity to infer the types of merchants. Using only the given data, identify different kinds of businesses in the sample. Generate assignments for each merchant.

Solution:

The two variables which can help us categorize the merchants based on the given data are -

1) **amount** a merchant spends on transactions
2) **frequency/count** of the transactions per merchant

Based on variable1, we can have three categories for merchants namely Low_Value, Medium_Value, High_Value.

Based on variable2, we can have three categories for merchants namely Low_Frequency, Medium_Frequency, High_Frequency.

Combining the above two we have the below nine categories in which the merchants belong to –

- Low_Value_Low_Frequency
- Low_Value_Medium_Frequency
- Low_Value_High_Frequency
- Medium_Value_Low_Frequency
- Medium_Value_Medium_Frequency
- Medium_Value_High_Frequency
- High_Value_Low_Frequency
- High_Value_Medium_Frequency
- High_Value_High_Frequency

These class categories will be able to point us in the direction of the kind businesses the merchants are into. For ex, we can consider a merchant whose transactions are low value but of high frequency to be the ones from a local supermarket, vendors etc. High value high frequency could be an online travel ticketing business. High value low frequency could be high end boutique store business where there might not be high number of transactions every single day, but the value of each transaction would be high.

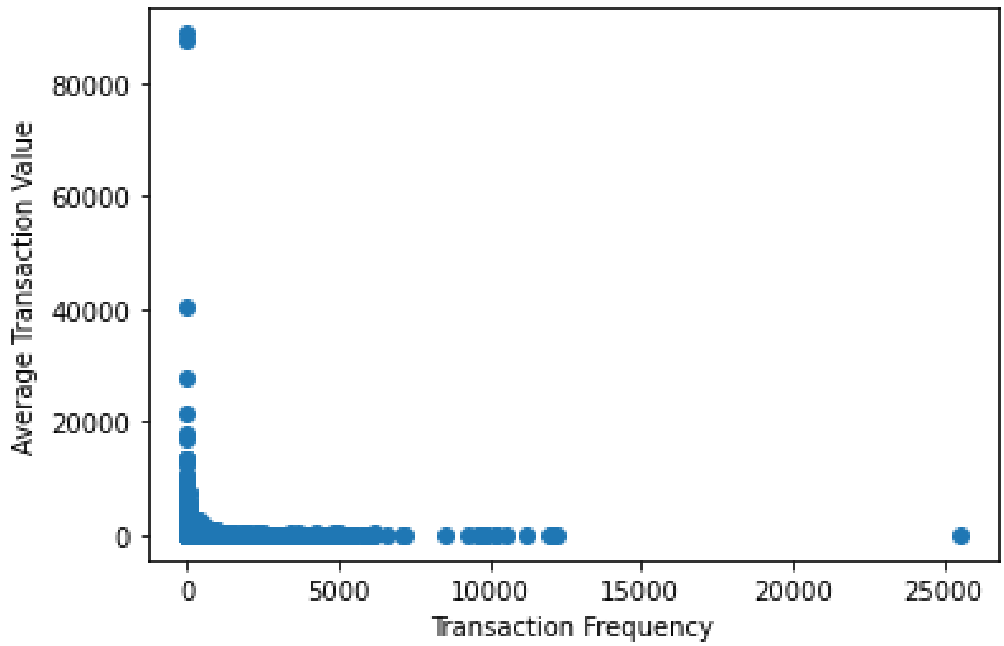
Step1: Data Preparation

	merchant	time	amount_usd
0	3e72388b82	2033-01-01 09:38:32	43.35
1	32cd721943	2033-01-01 12:53:52	60.19
2	a8ff2d667e	2033-01-01 15:08:55	39.42
3	cad5cd6286	2033-01-01 15:16:32	19.26
4	878047f4b9	2033-01-01 15:19:16	48.39

Step2: Analyzing the count and average transaction value of each merchant

Grouping by the merchant ID, we can fetch the count of transactions per merchant and the average transaction amount per merchant.

[Assumption] – I have an assumed here that the values which are very high compared to the distribution are not outliers, and that it might be possible to have wide range of transaction amounts as well as counts. Assumption that the data is clear of outliers has been made.



Step3 - Creating categories for amount and frequency:

Analyzing the distributions of count and average amount values, I was able to come up with the values that can give us the three classes under each category.

Count – [0-500], [500-1000], [>1000]

Avg Amount – [0-500], [500-5000], [>5000]

With this, I was able to classify and provide a label to each merchant based on the boundary values mentioned above.

	merchant	sum	min	max	count	mean	median	diff_mean_median	Merch_Type
7370	83c1813b63	2.01	2.01	2.01	1	2.010000	2.010	0.000000	Low_Freq_Low_Value
12354	dcc0c1c755	4.12	2.06	2.06	2	2.060000	2.060	0.000000	Low_Freq_Low_Value
14259	fe9bf14103	2.09	2.09	2.09	1	2.090000	2.090	0.000000	Low_Freq_Low_Value
8609	9963a11d63	6.52	2.12	2.28	3	2.173333	2.120	0.053333	Low_Freq_Low_Value
6421	7347d3b20c	4.36	2.09	2.27	2	2.180000	2.180	0.000000	Low_Freq_Low_Value
...	...	...	...	...	...	...	...	...	...
4384	4e5d7ec3de	86595.03	6136.44	47836.07	4	21648.757500	16311.260	5337.497500	Low_Freq_High_Value
2156	26f51e4c7e	110520.25	1466.15	67962.85	4	27630.062500	20545.625	7084.437500	Low_Freq_High_Value
3694	42229128c1	40475.21	40475.21	40475.21	1	40475.210000	40475.210	0.000000	Low_Freq_High_Value
459	0838e4078e	263459.53	2063.00	259202.80	3	87819.843333	2193.730	85626.113333	Low_Freq_High_Value
10370	b993083163	533247.91	79114.88	103855.08	6	88874.651667	85706.435	3168.216667	Low_Freq_High_Value

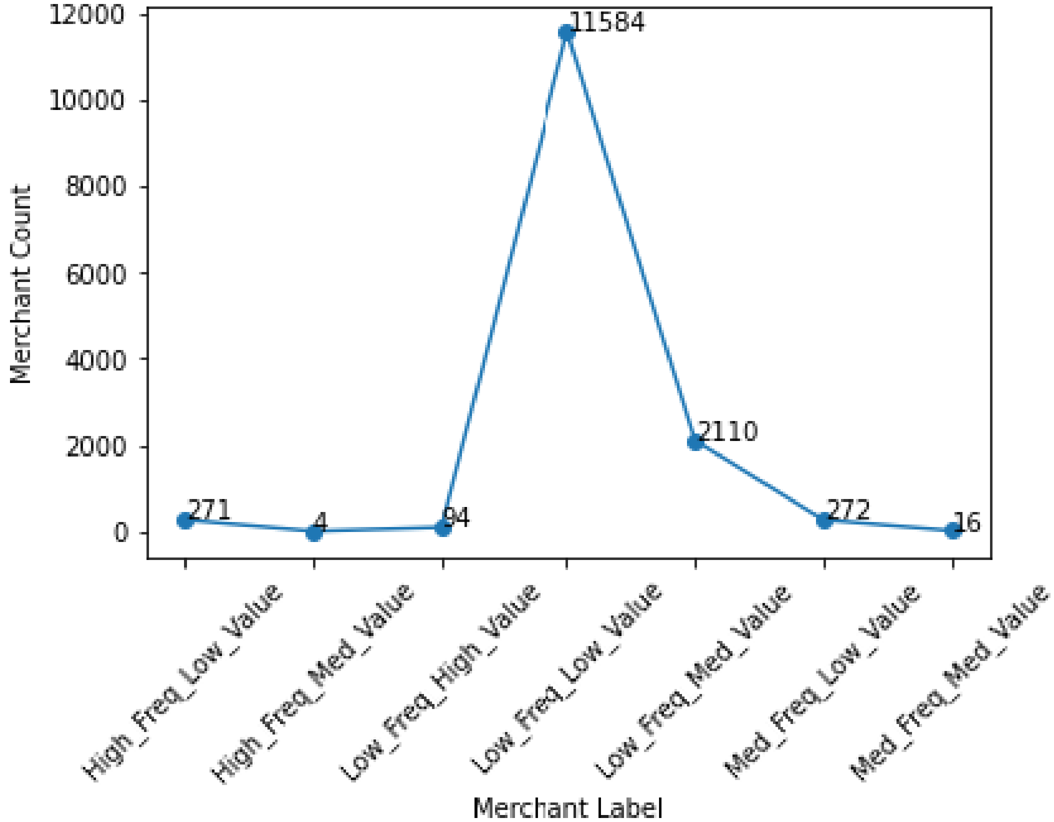
14351 rows × 9 columns

Step4 – Visualizing the merchant categories

In addition to categorizing the merchants, we can also see how many merchants fall into each of these categories. It gives us a picture of what kind of businesses are involved and how.

- Highest number of merchants belong to Low_Value_Low_Frequency category.
- There are no merchants in the High_Value_High_Frequency and High_Value_Medium_Frequency categories

Number of merchants falling into each of these categories is depicted with the graph below:



The numbers for the same are shown below:

	Low Value	Medium Value	High Value
Low Frequency	11584	2110	94
Medium Frequency	272	16	0
High Frequency	271	4	0

Part B:

a) come up with a concrete definition for churn

b) identify merchants that have already churned in the dataset

Solution:

To find the trends in merchant transactions that indicate that they are likely going to stop using the services of companyX, I considered the frequency at which the transactions from the merchant happen.

We have two key features to derive for the computation –

1) No of days between consecutive transactions of each merchant "days_btwn_consecutive"
2) No of days between each merchant's last transaction and the end date in the data set 'days_btwn_lasttransact'

Here, I have considered the churn period as 180 days and below are my inferences –

1) If # of days between consecutive transactions is more than 180 days, it means the merchant had an inactivity of 30 days but still returned.
2) If # of days between consecutive transactions is more than 180, it means the merchant had an inactivity of 180 days but still returned.
3) If # of days between consecutive transactions is less than 180 days but the difference between last date of data and the final transaction date is more than 180 days, it means the merchant churned.
4) If the # of days between consecutive transactions and the difference between last date, final transaction date both are less than 180 days, we can say that the merchant hasn't churned but also has not gone beyond 180 days of inactivity.
5) Count of churned customers divided by count of churned + 180 day returned customers is 72% which is significant in our case where we have data of about 730 days.
6) Although, ideally this percentage had to be more in the range of 85 to 100. However, considering the size of data set I have decided to freeze the churn period selected to 180 days.
7) We can now use the same logic to determine who have churned already and who will churn in the future.
8) Details of the below data is explained along with the code in the python notebook file.

	time	consec_flag180	merchant	last_transaction	days_btwn_lasttransact	lasttransact_flag180
0	2033-05-16 20:07:57	<180	0002b63b92	2034-12-31 07:59:40	593	>180
4	2034-12-15 09:56:19	<180	0002d07bba	2034-12-31 07:59:40	15	<180
32	2033-08-04 04:26:40	<180	00057d4302	2034-12-31 07:59:40	514	>180
33	2033-08-09 20:18:36	<180	000bcff341	2034-12-31 07:59:40	508	>180
34	2033-06-02 13:25:12	<180	000ddbf0ca	2034-12-31 07:59:40	576	>180