Financial Programming

Professor: Minh Phan

Group Project – Financial Data Set

Fernando Delgado

Hina Hussain

Nixia Sancy John

December 7, 2021

Table of Contents

Case Background

General Context

The following report describes the creation of a base-table for the Prague Bank, along with its description and analysis. The objective of creating the base table is to use it for further data science analysis and modelling as well as predicting the future behavior of their clients.

The Prague Bank offers services to private customers, such as account management, and loan offering. The bank keeps a data record about clients, accounts, loans, and credit cards issued (as a financial dataset).

With the bank's financial dataset, the team has constructed a base-table using Python programming. This report explains how the raw data was treated to create the base-table, displays the base table, and analyzes the independent and dependent variables of the table.

Raw Dataset

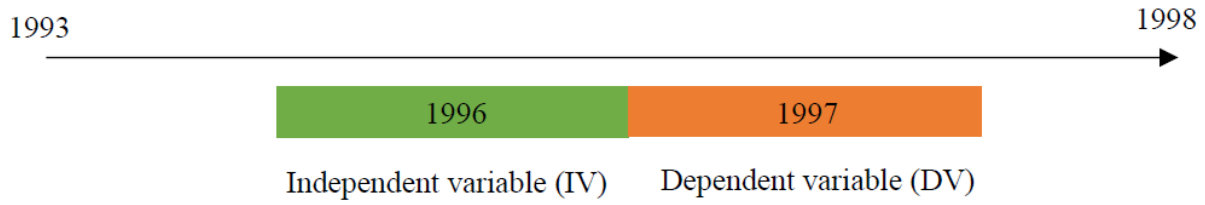
The dataset provided by the bank consists of the following tables:

- Account (4500 objects in the file ACCOUNT.ASC) - each record describes static characteristics of an account
- Client (5369 objects in the file CLIENT.ASC) - each record describes characteristics of a client
- Disposition (5369 objects in the file DISP.ASC) - each record relates together a client with an account i.e. this relation describes the rights of clients to operate accounts,
- Orders (6471 objects in the file ORDER.ASC) - each record describes characteristics of a payment order,
- Transaction (1056320 objects in the file TRANS.ASC) - each record describes one transaction on an account,
- Loan (682 objects in the file LOAN.ASC) - each record describes a loan granted for a given account

- Credit Card (892 objects in the file CARD.ASC) - each record describes a credit card issued to an account
- Demographic (77 objects in the file DISTRICT.ASC) - each record describes demographic characteristics of a district.

Timeline Analysis

For the analysis, we followed the timeline criteria below:



Time window for independent variables is 1996, while for dependent variables it is 1997. In other words, our base-tables' features are all generated while using data of 1996, while the target variables (those that our model would like to predict) is made using data of 1997. Moreover, as the case stated, we calculated two dependent variables:

- Clients that were granted a loan in 1997
- Clients that had a credit card issued in 1997

Data Correction & Transformation

Before creating the base-table, first, data correction and transformation was performed on each individual table. All data correction and transformation were done with Python, utilizing NumPy, Pandas, and Datetime libraries.

The following steps were taken for each individual table:

Accounts Table

- The account creation year was extracted from the date variable
- Account creation date was formatted to YYYMMDD
- The accounts table was filtered to only keep accounts made before 1996 to be in alignment with the timeline.
- Length of relationship was created from year account created to 1996
- The variables under statement frequency were renamed from Czech to English
- Columns were renamed to more understandable names
- Unwanted columns were dropped

Clients Table

- Client birth year was extracted from the birth number variable
- Client gender was extracted from the birth number variable
- Client age was calculated as of 1996
- A new column was added to categorize clients into different age groups – Children, Youth, Adults, Seniors.
- Dummy variables were created for the age groups
- Unwanted columns were dropped

Orders Table

- Missing and NA values in the order type column were replaces with type “Others”
- New column was added for total order amount for each account
- New column was added for amount per order type for each account
- Missing values were treated
- Columns were renamed to more understandable names

Transactions Table

- The table was filtered to keep only transactions which took place in 1996
- Transaction year was extracted from the transaction date
- A new column was created to show the total credit per account
- A new column was created to show the total withdrawal per account
- A new column was created to indicate the transaction type i.e. credit or withdrawal
- New columns for opening balance and closing balance of the year were calculated for each account
- Missing and NA values in the k symbol column were replaces with type “Others”
- A new column was added to show the total number of transactions per account
- Transaction date was formatted to YYYMMDD
- Columns were renamed to more understandable names
- New columns were created for total transaction amount per each transaction type for every account
- Created an RFM model in order to analyze customer segmentation by transaction behavior (columns for recency, frequency, and monetary value were added)
- Average Monthly balance was calculated as the mean of the opening and closing balances.
- Average salary per month was extracted by dividing the annual total credit by 12.

Loans Table

- Loans table was filtered for loan granted in 1996
- Loan granting year was extracted from the date
- Loan granting date was formatted to YYYYMMDD
- Dummy variables were made for each of the four loan statuses. One column for each status was created with Booleans.
- Unwanted columns were dropped
- Columns were renamed to more understandable names
- Duration left and amount to be paid were calculated as of 1996/12/31

Credit Cards Table

- Cards table was filtered for cards issued in 1996
- The time column was split to take out the time and drop it
- Card issue date was obtained from the date column
- Dummy variables were made for each of the three card types. One column for each type was created with Booleans.
- Columns were renamed to more understandable names

District Table

- Dropped column A15 and A12, which represented crimes in 1995 and unemployment in 1995 respectively
- All culumns were renamed to their proper meaning. For example, A1 was renamed as 'district_id'
- Dummy variables were made for each of the regions. One column for each region was created with Booleans.
- Each district was categorized as either a high or low unemployment area. Districts with an unemployment rate of more than 5% were labelled as high and the rest as low.
- Each district was categorized as either a high or low salary area. Districts with an average salary level higher than the mean average salary across all districts were labelled as high and the rest as low.

Disposition Table

- Applied a filter to select only the account Owners

Fix Missing Values

For both Order and Transaction tables we fixed missing values that should have been zero instead. For example, if an account did not perform an insurance payment, it would erroneously show as a Null value. Therefore, we replaced the nulls with 0's, to display that the total amount of insurance payment was indeed zero.

For categorical values which do not have values, we are replacing nulls with "missing" and creating a variable '<Col_name>_is_missing'.

Treat Outliers

In order to find the columns that have outliers we used the statistical method that processes: if any value is 3 times more than the 75th quantile value, then it is consider an outlier. Moreover, we used the *winsorize()* function to replace the outliers values.

Base-table

To merge our tables and create a base-table, first we took Clients table as a base (since clients is the lowest granularity level).

Then, we performed the following merges:

- Left joined with Dispositions on client_id
- Right joined with Accounts on account_id
- Left joined Loans on account_id
- Left joined orders on account_id
- Left joined transactions for each account_id
- Merged credit card table on account_id
- Merged district table on district_id

The result was a base-table with 2239 observations and 94 variables. Each row is for one account owner.

Additional Variables

Customer Segmentation: RFM Model

To perform in-depth customer segmentation by their transaction behavior, wan RFM model was performed:

- Recency: number of days since last transaction
- Frequency: number of transactions made during the given period
- Monetary Value: total amount transacted by account

To do so, first we took our 'last' date of the period as 31st of December of 1996. The date difference within the account's last transaction and the 'last' date would show us the recency. Then, simply by counting the number of transactions per account during the period, we calculated the

frequency. Finally, we summed the total amount of money transacted (withdrawals and credit) per account to calculate monetary value.

Following, we had to assign levels for each RFM value, given that they are not comparable between each other. To do so, we used a percentile approach. By dividing the range of values into 5 groups (quintiles), we assigned 5 different levels to each RFM value, where 0 is the worst and 4 is the best. For example, in the table below we can observe that those values with a recency of 0 have an R score of 4, meaning that it's been 0 days since their last transaction (which gives them the highest score of 4).

account_id	recency	frequency	monetaryvalue	R	F	M	RFM_Score
1	0.0	148	105645.2	4.0	1.0	0.0	5.0
2	0.0	170	563117.1	4.0	1.0	1.0	6.0
4	0.0	78	88898.1	4.0	0.0	0.0	4.0
6	0.0	126	163310.9	4.0	1.0	0.0	5.0
7	0.0	4	41505.4	4.0	0.0	0.0	4.0

Then, simply by summing the RFM values we can obtain a total RFM Score. Thus, we can compare each account by their 3 levels of frequency, recency, and monetary value.

Furthermore, we can segment each account on a particular level. In this case we applied the following criteria:

- Bronze Account: RFM score of 5 or higher
- Silver Account: RFM score of 6 or higher
- Gold Account: RFM score of 7 or higher
- Platinum Account: RFM score of 10 or higher
- Needs Attention: All accounts below an RFM score of 5

With the applied criteria, we could easily segment our accounts into groups of client types as shown below:

		Mean Values		
RFM Level	Account Count	Recency	Frequency	Monetary Value
Platinum	11	0	314	1,632,270.0
Gold	367	0	200	1,023,689.8
Silver	634	0	155	542,796.5
Bronze	1311	0	139	224,185.6
Needs Attention	1279	3	38	100,261.1

Dependent Variables

Once the independent variables are set in the base table, two dependent variables were created with Boolean values (1 if True, 0 if False):

- Loan granted in 1997
- Card issued in 1997

Base Table Description

The final base table has 94 variables with 2239 observations. Below are the number of columns by datatype:

Datatype	No of Columns
int64	16
int32	24
float64	38
object	4
datetime64[ns]	2
uint8	10

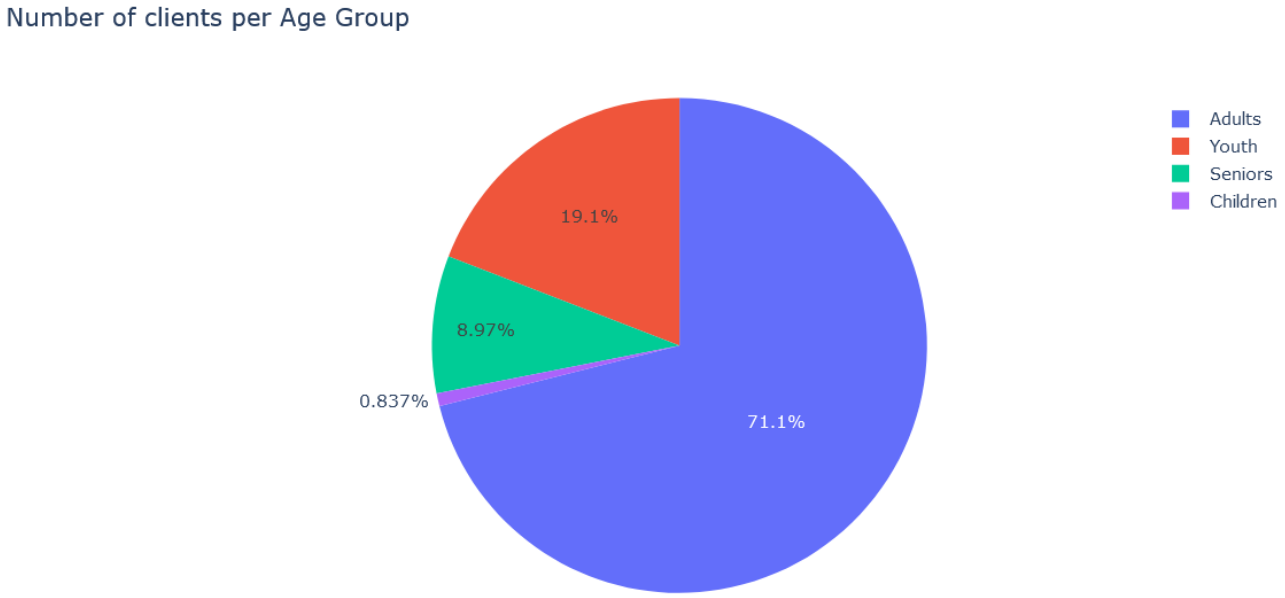
Please refer to the annexed file, *Base-Table Description.pdf* to read a detailed description of every variable within the base-table.

Base-Table Description.pdf

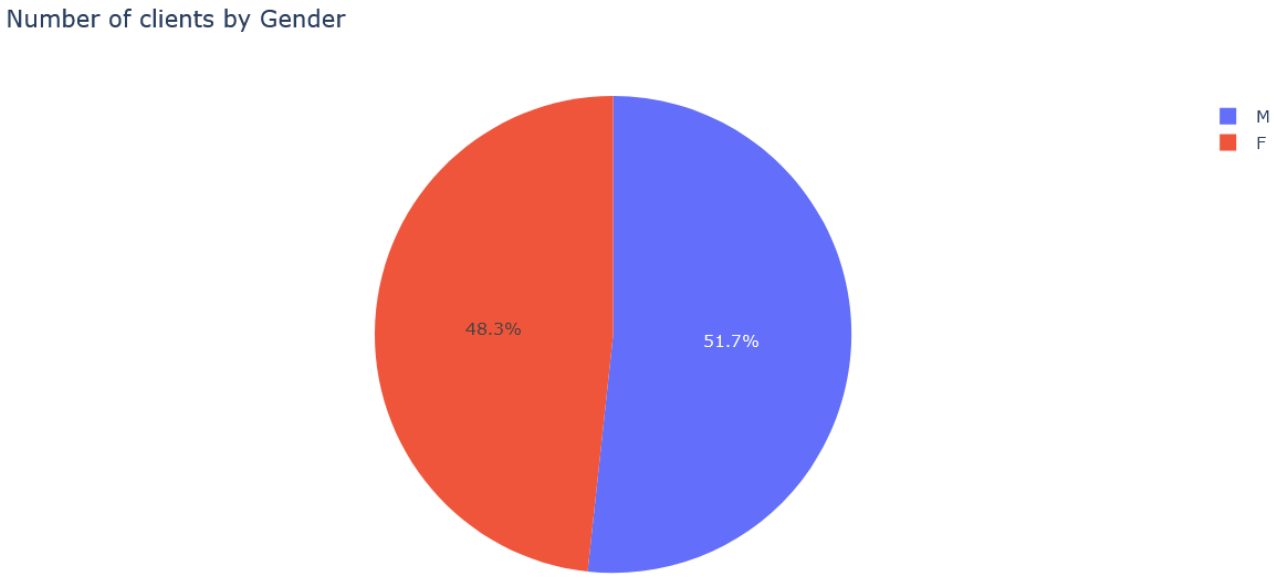
Visualization Analysis

Customer types

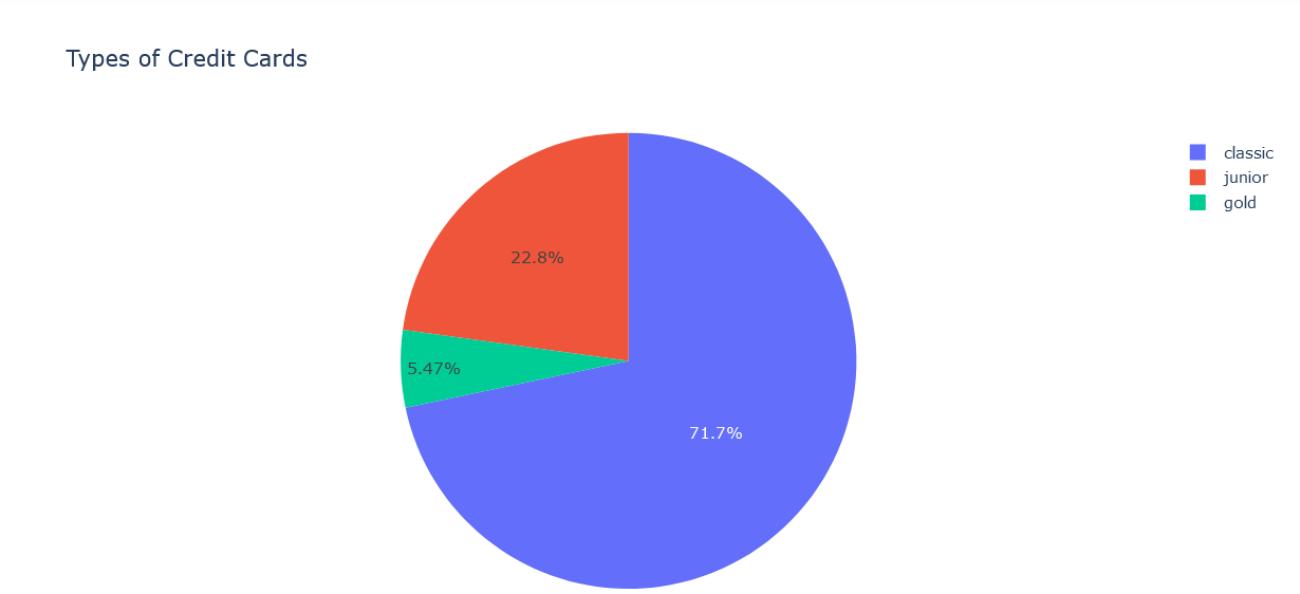
The following graphs shows the distribution of clients by age group, displaying that most of them are Adults:



Following, there is a slight majority of males within the database:

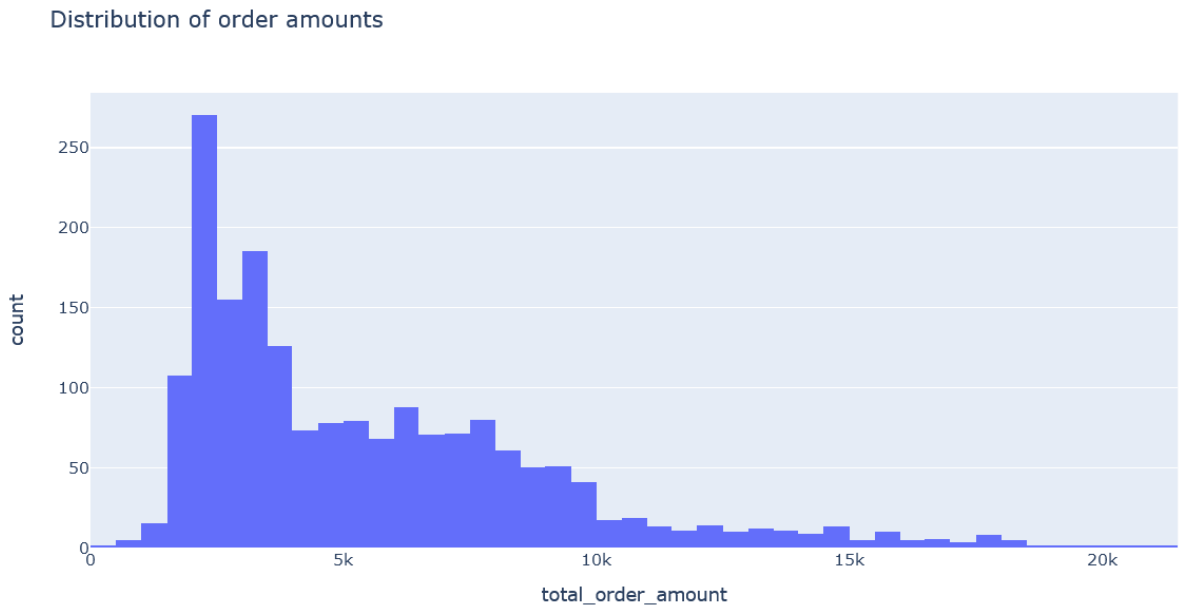


In addition, while analyzing credit card performance in 1996. we can observe that the majority of the credit card holders belong to the ‘classic’ program:

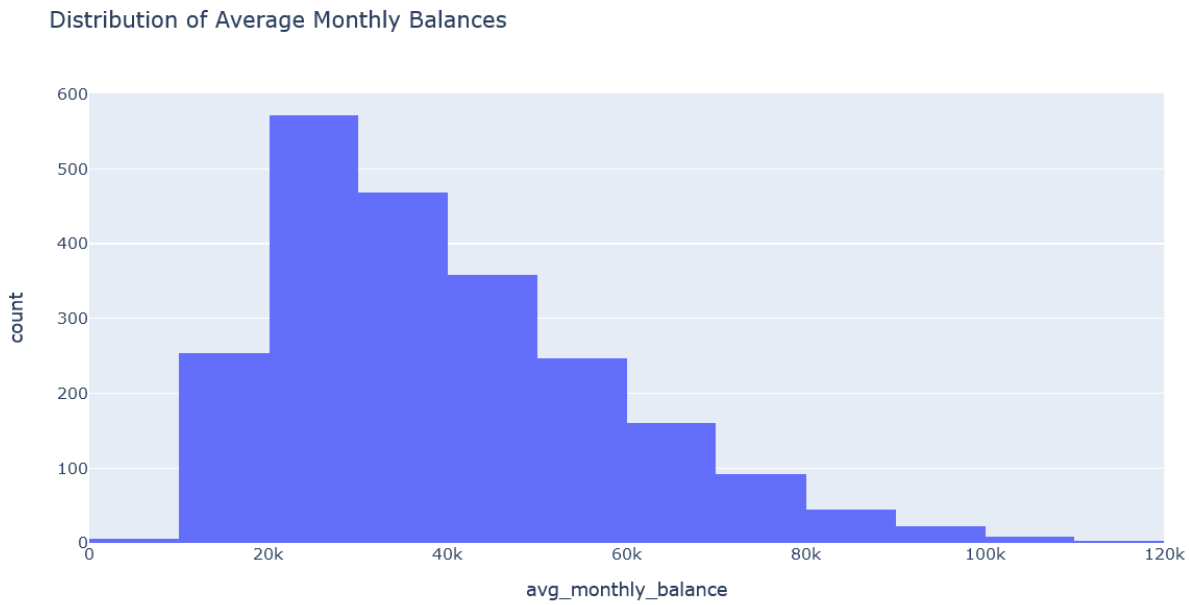


Distribution Analysis

Additionally, we can observe that the majority of orders are done for amount of between 0 to 5,000, giving us a right-skewed histogram:

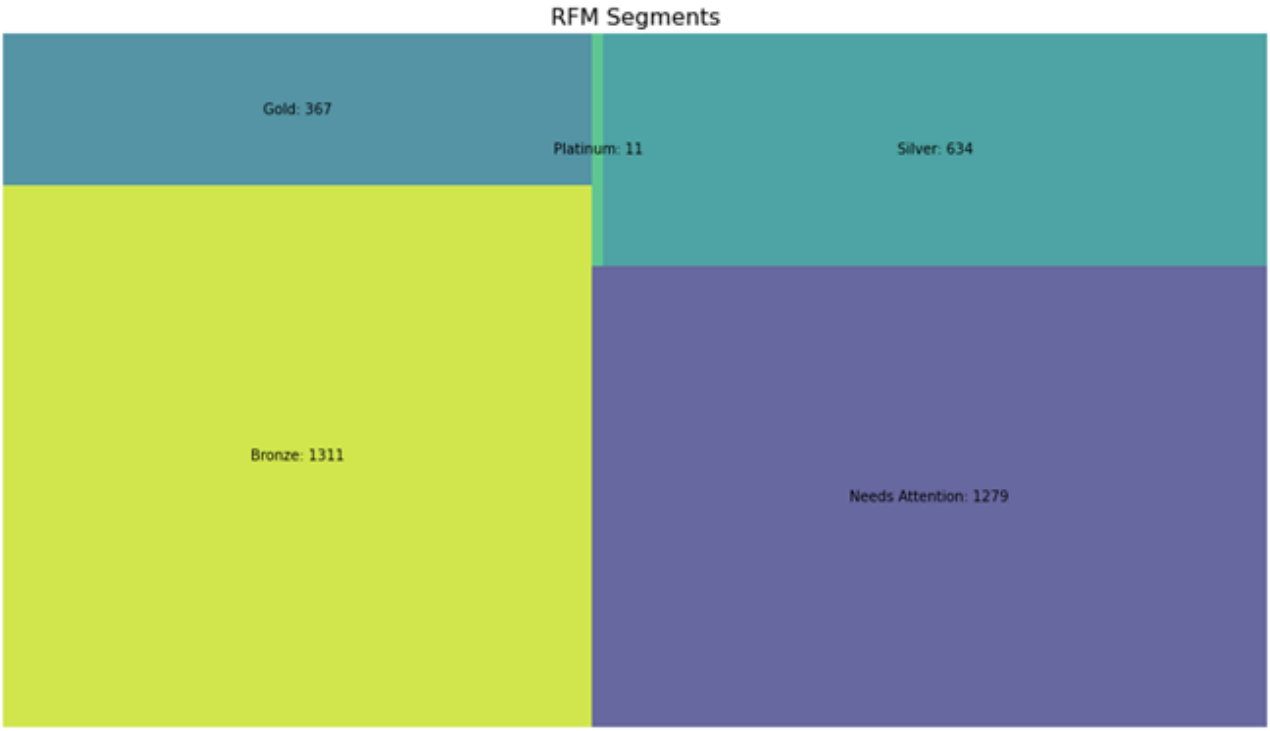


Moreover, the distribution of Average Monthly Balance per account is also right skewed:

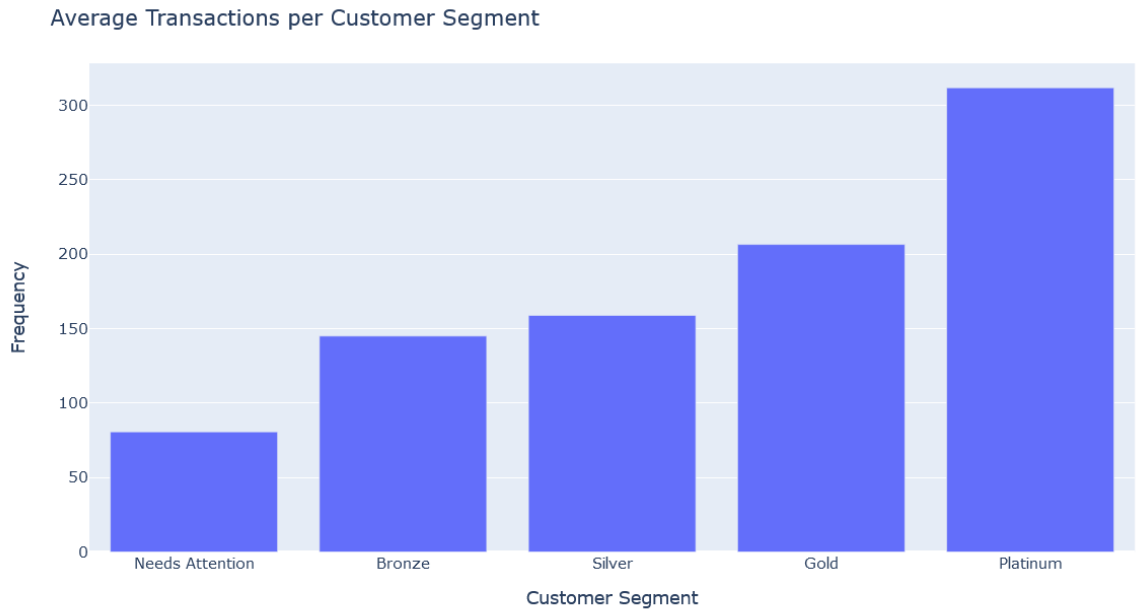


RFM Analysis

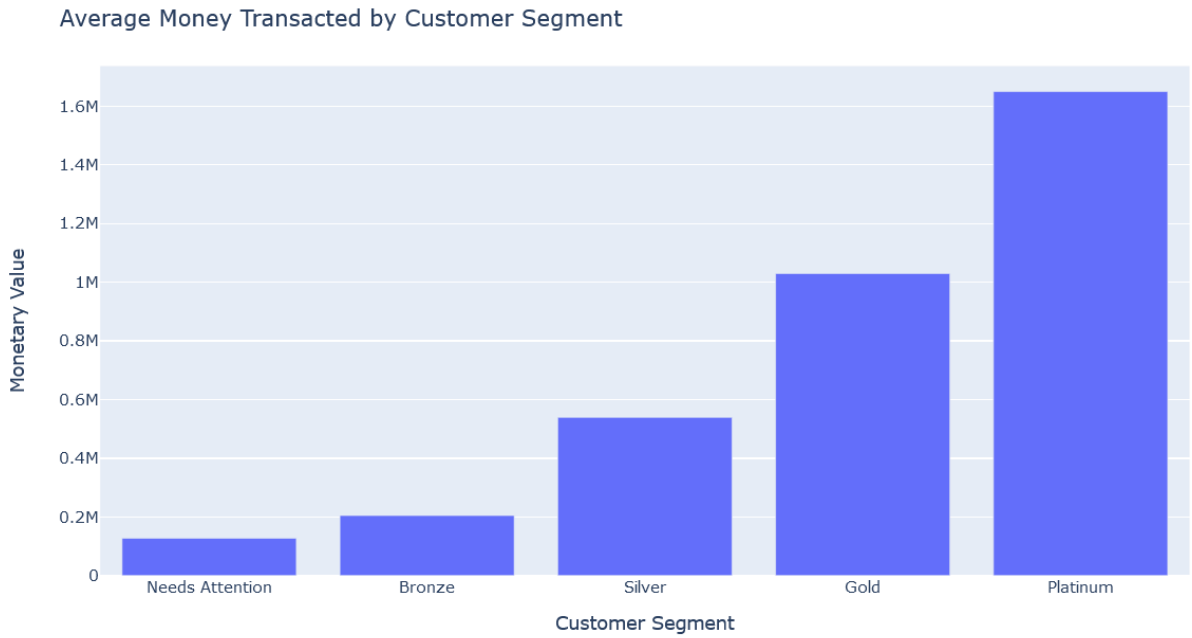
We can observe our customer segmentation with the given levels with the following graph:



Furthermore, our customer segment analysis with RFM shows that the average number transactions per customer for platinum segment is the highest:

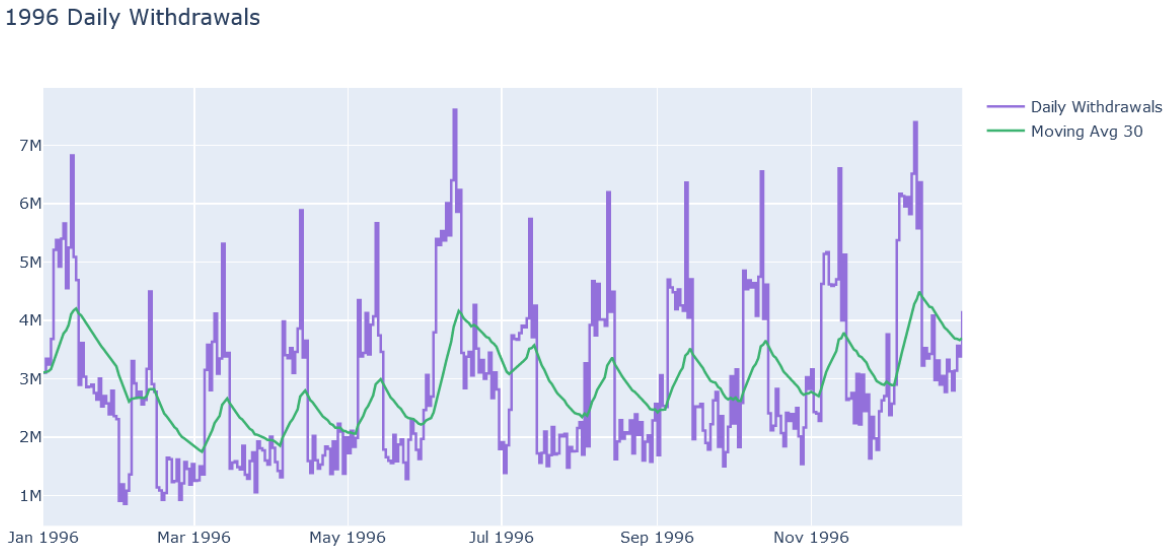


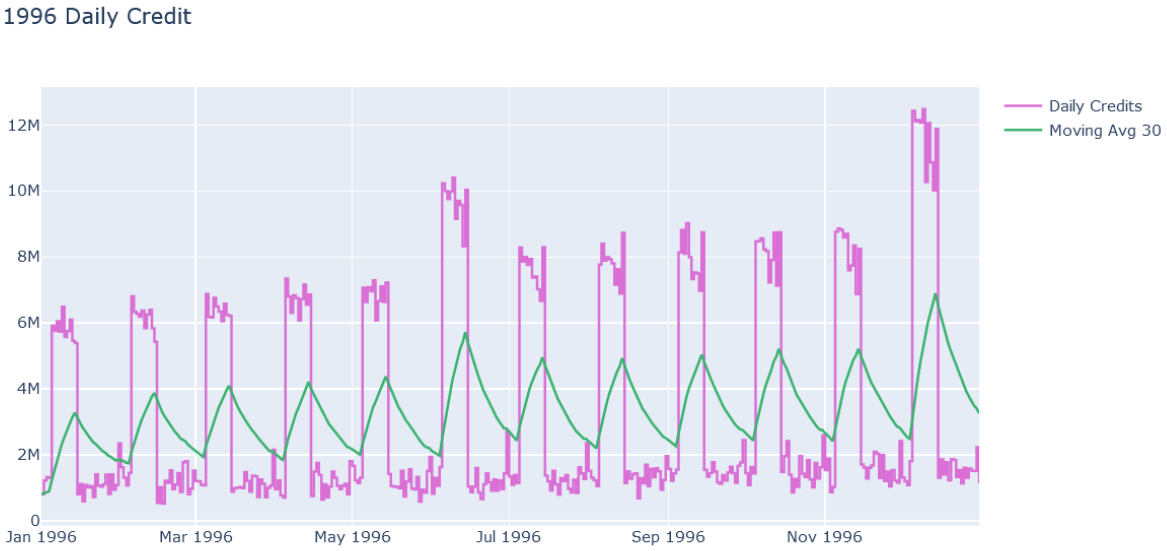
While the amount of money transacted has a similar distribution:



Daily Transactions

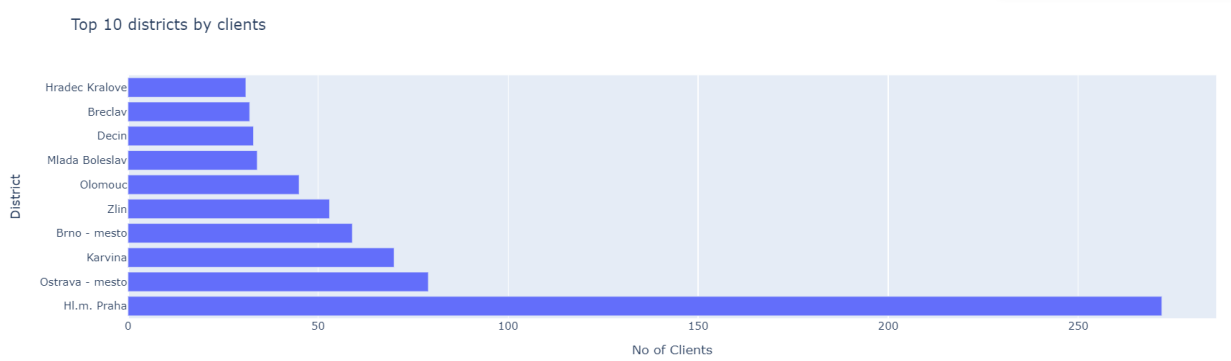
Daily total withdrawals and credit transactions for the period of 1996 graphed with a moving average of 30 days (monthly). We can observe peaks at the start of every month, which most likely represents monthly operational transactions:





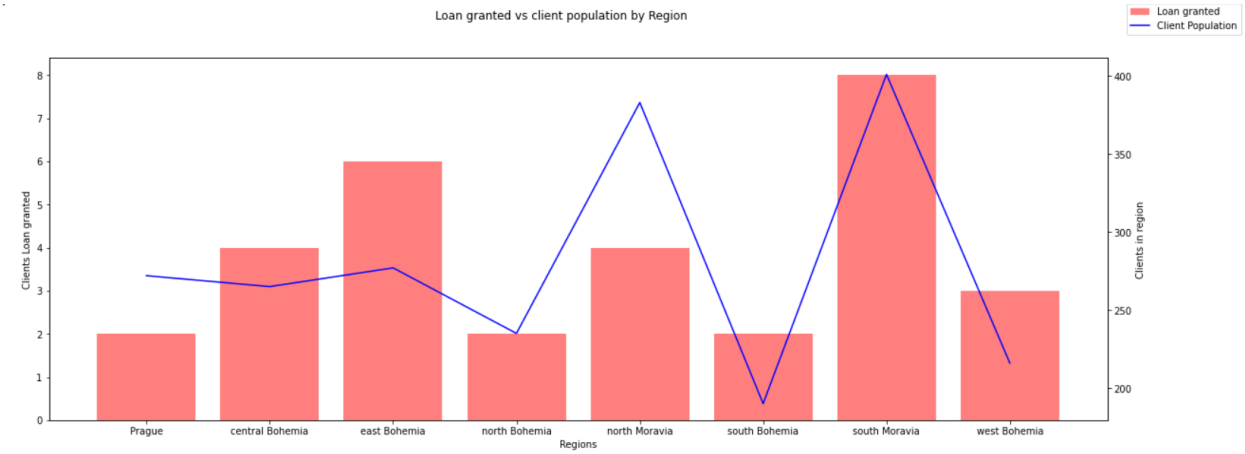
Clients by Districts

Highest number of clients come from the district of Hl.m.Prague. Meanwhile, the number of clients from other districts are nearly equally distributed.



Loans granted by Region

The below is a graph which shows the number of loans granted to each region vs the clients in the region. Though the regular pattern appears to be that the loans are higher where the clients are higher, we see in North Moravia, despite the low number of clients the loans granted are higher in propotion.



Percentage of Loans granted, and cards issued

The below graph shows the distribution of clients who were granted loan in 1997 or had a card issued to them in 1997.

Model Implementation

The base table was used to run a logistic regression model on each of the two dependent variables. Few important features were selected as independent variables using the Pearson correlation method. The resulting coefficients from the models are as below:

Logistic regression to predict Loan grant:

OLS Regression Results

Dep. Variable:	loan_granted_97	R-squared:	0.041
Model:	OLS	Adj. R-squared:	0.039
Method:	Least Squares	F-statistic:	19.04
Date:	Sat, 04 Dec 2021	Prob (F-statistic):	2.41e-15
Time:	16:20:39	Log-Likelihood:	1407.6
No. Observations:	1791	AIC:	-2805.
Df Residuals:	1786	BIC:	-2778.
Df Model:	4		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-9.936e-06	1.49e-06	-6.656	0.000	-1.29e-05	-7.01e-06
acc_created_year	1.738e-05	4.46e-06	3.895	0.000	8.63e-06	2.61e-05
lor	-0.0198	0.003	-6.655	0.000	-0.026	-0.014
total_order_amount	3.085e-06	7.43e-07	4.153	0.000	1.63e-06	4.54e-06
order_miscellaneous	1.586e-06	1.9e-06	0.834	0.405	-2.14e-06	5.32e-06
opening_balance	1.61e-07	1.14e-07	1.411	0.158	-6.28e-08	3.85e-07

Omnibus:	2477.069	Durbin-Watson:	2.018
Prob(Omnibus):	0.000	Jarque-Bera (JB):	354145.027
Skew:	8.119	Prob(JB):	0.00
Kurtosis:	69.948	Cond. No.	1.49e+20

Logistic Regression to predict Card Issue:

OLS Regression Results

Dep. Variable:	card_issued_97	R-squared:	0.075
Model:	OLS	Adj. R-squared:	0.069
Method:	Least Squares	F-statistic:	13.10
Date:	Sat, 04 Dec 2021	Prob (F-statistic):	2.67e-24
Time:	16:20:39	Log-Likelihood:	180.77
No. Observations:	1791	AIC:	-337.5
Df Residuals:	1779	BIC:	-271.7
Df Model:	11		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-0.0091	0.061	-0.150	0.881	-0.128	0.110
total_credit_per_account	-2.946e-06	2.79e-05	-0.106	0.916	-5.76e-05	5.17e-05
total_withdrawal_per_account	-3.434e-06	2.74e-05	-0.125	0.900	-5.71e-05	5.03e-05
opening_balance	-2.767e-07	3.5e-07	-0.791	0.429	-9.63e-07	4.1e-07
closing_balance	2.414e-06	5.22e-07	4.627	0.000	1.39e-06	3.44e-06
total_withdrawal_cash	-1.294e-07	1.34e-07	-0.968	0.333	-3.91e-07	1.33e-07
total_credit_cash	-2.589e-08	4.63e-08	-0.559	0.576	-1.17e-07	6.49e-08
total_miscellaneous	9.468e-06	1.24e-05	0.765	0.444	-1.48e-05	3.37e-05
monetaryvalue	3.817e-06	2.74e-05	0.139	0.889	-4.99e-05	5.75e-05
avg_monthly_balance	1.069e-06	1.84e-07	5.803	0.000	7.08e-07	1.43e-06
average_salary_account	-1.49e-05	2.2e-05	-0.676	0.499	-5.81e-05	2.83e-05
M	0.0355	0.024	1.461	0.144	-0.012	0.083
RFM_Score	-0.0133	0.013	-1.029	0.304	-0.039	0.012

Omnibus:	1278.555	Durbin-Watson:	1.938
Prob(Omnibus):	0.000	Jarque-Bera (JB):	13105.991
Skew:	3.451	Prob(JB):	0.00
Kurtosis:	14.313	Cond. No.	1.20e+17