BERKA DATA ANALYSIS

Vinay Rajagopalan
Swasthik Vellingiri
Sai Sumanth Sripada
Financial Programming
By Prof.Minh Phan

Table of Contents

Introduction :

Banks deal with a vast amount of data inside their organizations, but their ability to interpret and extract value from it is the area where they have struggled over the years. Data in banking provides an inbuilt advantage to them over their smaller competitors.

A leading Bank Based out of Czech Republic which provides multiple services like Loan, Credit Card and Premium Accounts to the Customers wants to get the insights on their clients and the analysis of their operating patterns based on the past data they have. Managers wants to improve their services based on the insights from the data. They have the database of their client details and their operating patterns over the period of time like Client Region, Age their transaction Patterns

Based on the Provided data, we are obliged to Make the data Mart with all the details of the clients and their account operating patterns. Below is the Data Mart Followed by the Steps Followed to Process the Data

DataMart

Sl. No	Variable Name	Description	Data Type	Value (R = Range)	Categorical/Numerical
1	account_id	identification of the account	int	R: 1 to 11382	Categorical
	account_district_id	Location of the branch	int	R: 1 to 77	Categorical
	account_frequency	frequency of issuance of statements	Object	R: 1 to 77	Categorical
	account_year	Year of account creation	int	R: 1993 to 1996	Categorical
	account_month	Month of account creation	int	R: 1 to 12	Categorical
	account_day	Day of account creation	int	R: 1 to 30	Categorical
	account_lor	Account length of relationship (years)	int	R: 1 to 4	Numerical
	client_id	client identifier	int	R: 1 to 13998	Categorical
	client_birth_year	Client's birth year	int	R: 1918 to 1982	Categorical
	client_birth_month	Client's birth month	int	R: 1 to 12	Categorical
	client_birth_day	Client's birth day	int	R: 1 to 31	Categorical
	client_gender	Gender of the client	Object	F, M	Categorical
	client_age	Age of the client	int	R:15 to 79	Numerical

	client_age_group	Age group of the client (example age 57 = age group 50)	int	10, 20, 30, 40, 50, 60, 70	Categorical
	client_category	Category of the client based on their age	Object	Young(< 21), Adult(21-55),Senior Citizen(> 55)	Categorical
	District_Code	District code	int	R: 1 to 77	Categorical
	District_Name	Name of the district	Object	Benesov, Znojmo etc.	Categorical
	Region	Region	Object	R: Prague, West bohemia etc.	Categorical
	n_inhabitants	no. of inhabitants	int	R: 42821 to 1204953	Numerical
	municipalities_inh_499	no. of municipalities with inhabitants < 499	int	R: 0 to 151	Numerical
	municipalities_500_1999	no. of municipalities with inhabitants 500-1999	int	R: 0 to 70	Numerical
	municipalities_2000_9999	no. of municipalities with inhabitants 2000-9999	int	R: 0 to 20	Numerical
	municipalities_10000	no. of municipalities with inhabitants >10000	int	R: 0 to 5	Numerical
	n_cities	no. of cities	int	R: 1 to 11	Numerical

	ratio_urban_inh	ratio of urban inhabitants	float	R: 33 to 100	Numerical
	average_salary	average salary	int	R: 8110 to 12541	Numerical
	unemploymant_rate_95	unemployment rate '95	float	R: 0 to 7.34	Numerical
	unemploymant_rate_96	unemployment rate '96	float	R: 0.43 to 9.4	Numerical
	entrepreneurs_per_1000	no. of entrepreneurs per 1000 inhabitants	int	R: 81 to 167	Numerical
	committed_crimes_95	no. of commited crimes '95	int	R: 0 to 85677	Numerical
	committed_crimes_96	no. of commited crimes '96	int	R: 888 to 99107	Numerical
	increase_in_unemploymant_rate	Difference between the unemployment rate between the years 1995 and 1996 (if increased)	float	R: 0 to 7	Numerical
	increase_in_committed_crimes	Difference between the committed crimes between the years 1995 and 1996 (if increased)	float	R: 0 to 13430	Numerical
	decrease_in_unemploymant_rate	Difference between the unemployment rate between the years 1995 and 1996 (if decreased)	float	R: 0 to 0.35	Numerical
	decrease_in_committed_crimes	Difference between the	float	R: 0 to 974	Numerical

		committed crimes between the years 1995 and 1996 (if decreased)			
	total_municipalities	Total number of municipalities of the district	int	R: 1 to 197	Numerical
	number_of_employed_population_96	Number of employed population of the district in the year 1996	int	R: 39819 to 1199771	Numerical
	number_of_employed_urban_population_96	Number of employed urban population of the district in the year 1996	int	R: 38996 to 1187773	Numerical
	num_order	Total number of Orders made	float	R: 1 to 5	Numerical
	total_order_amount	Total Debited Amount	float	R: 312 to 21322	Numerical
	LEASING	characterization of the order payment = Leasing	float	R: 0 to 4975	Numerical
	Other	characterization of the order payment = Other Payments	float	R: 0 to 12531	Numerical
	POJISTNE	characterization of the order payment = Insurance payment	float	R: 0 to 12504	Numerical

	SIPO	characterization of the order payment = household payment	float	R: 0 to 14723	Numerical
	UVER	characterization of the order payment = loan payment	float	R: 0 to 9910	Numerical
	loan_id	Loan record identifier	int	R: 4959 to 7308	Categorical
	loan_amount	amount of money lent	float	R: 4980 to 590820	Numerical
	loan_duration	duration of the loan	float	R: 12 to 60	Numerical
	loan_payments	monthly payments	float	R: 304 to 9910	Numerical
	loan_year	Year when the loan was granted	int	R: 1993 to 1998	Categorical
	loan_month	Month when the loan was granted	int	R: 1 to 12	Categorical
	loan_day	Day when the loan was granted	int	R: 1 to 31	Categorical
	loan_category	Category of the loan based on the amount. Amount < 196940 = Low	Object	Low (<196940), Medium (<393880.0), High (>393880.0)	Categorical
	loan_granted	Whether the loan is granted or not (Dependent variable)	float	1, 0	Categorical

	card_id	Card record identifier	int	R: 1 to 1247	Categorical
	card_type	Type of card	Object	Gold, Classic, Junior	Categorical
	card_issued_year	Year of card issuance	int	R: 1993 to 1998	Categorical
	card_issued_month	Month of card issuance	int	R: 1 to 12	Categorical
	card_issued_day	Day of card issuance	int	R: 1 to 31	Categorical
	card_LOR	Length of relationship calculation since the card issued (days)	timedelta	R: 1 to 1151 days	Numerical
	card_issued_97	Whether the card is issued or not (Dependent variable)	float	1,0	Categorical
	last_trans_date	Date of the latest transaction	Object	R: 1996/08/25 To 1998-12-31	Numerical
	trans_recency	Days since last transaction	Int	R: 0 tp 858	Numerical
	trans_frequency	Frequency of the transactions	int	R: 9 to 675	Numerical
	monetary	Total Amount of the transactions	float	R: 29400 to 7619102	Numerical
	rfm_Score	RFM score (Calculated using recency, frequency and monetary)	float	Range = 0 to 100	Numerical

	account_segment	Segmenting accounts based on their RFM score	object	Low Value(0-30), Gold(30-70), Platinum(70-100)	Categorical
	total_trans_amount	Total transaction amount	float	Range = 29400 - 7619102	Numerical
	cash_withdrawal	Total amount Spent in withdrawal	Float	R: 6952 to 3655226	Numerical
	trans_avgbalance	Average Transaction Balance	Float	R: 5711 to 81193	Numerical
	num_of_oldage_pension	Total number of transactions for old age pensions	Float	R: 1 to 12	Numerical
	num_of_other	Total number of transactions for other not specified categories	Float	R: 1 to 24	Numerical
	num_of_insurrance_payment	Total number of transactions for insurance payment	Float	R: 1 to 12	Numerical
	num_of_interest_negativebalance	Total number of transactions for Interest if negative balance	Float	R: 1 to 13	Numerical
	num_of_household_payment	Total number of transactions for household payment	Float	R: 1 to 13	Numerical
	num_of_statement_payment	Total number of	Float	R: 1 to 12	Numerical

		transactions for payment for statement			
	num_of_interest_credited	Total number of transactions for interest credited	Float	R: 1 to 36	Numerical
	num_of_loan_payment	Total number of transactions for loan payment	Float	R: 1 to 12	Numerical
	amount_of_oldage_pension	Total amount (sum) of transactions for old age pensions	Float	R: 4182 to 83844	Numerical
	amount_of_other	Total amount (sum) of transactions for other not specified categories	Float	R: 12 to 150372	Numerical
	amount_of_insurrance_payment	Total amount (sum) of transactions for insurance payment	Float	R: 8 to 112536	Numerical
	amount_of_interest_negativebalance	Total amount (sum) of transactions Interest if negative balance	Float	R: 0.1 to 454	Numerical
	amount_of_household_payment	Total amount (sum) of transactions for household payment	Float	R: 6 to 195400	Numerical
	amount_of_statement_payment	Total amount (sum) of transactions payment for statement	Float	R:14 to 1200	Numerical

	amount_of_interest_credited	Total amount (sum) of transactions interest credited	Float	R:2 to 6653	Numerical
	amount_of_loan_payment	Total amount (sum) of transactions for loan payment	Float	R: 878 to 116268	Numerical
	num_of_card_withdrawal	Total Number of withdrawals in credit card	Float	R: 1 to 20	Numerical
	num_of_cash_withdrawal	Total Number of withdrawals in cash	Float	R: 1 to 66	Numerical
	num_of_credit_cash	Total Number of credit in cash	Float	R:1 to 34	Numerical
	num_of_credit_otherbank	Total number of collection transaction from other bank	Float	R: 1 to 12	Numerical
	num_of_transfer_otherbank	Total number of remittance to another bank	Float	R: 1 to 60	Numerical
	amount_of_card_withdrawal	Total amount (sum) of withdrawals in credit card	Float	R:500 to 44800	Numerical
	amount_of_cash_withdrawal	Total amount (sum) of withdrawals in cash	Float	R: 14 to 991818	Numerical
	amount_of_credit_cash	Total amount (sum) of credit in cash	Float	R: 200 to 1036762	Numerical
	amount_of_credit_otherbank	Total amount (sum) of collection	Float	R: 2910 to 648010	Numerical

		transaction from other bank			
	amount_of_transfer_otherbank	Total amount (sum) of remittance to another bank	Float	R: 139 to 255866	Numerical
	num_of_credits	Total number of Credit transactions	Float	R: 1 to 58	Numerical
	num_of_withdrawals	Total number of Withdrawal transactions	Float	R: 1 to 101	Numerical
	amount_of_credit	Total amount (sum) of Credit transactions	Float	R: 200 to 1040784	Numerical
	amount_of_withdrawals	Total amount (sum) of Withdrawal transactions	Float	R: 14 to 1020970	Numerical

Data Correction and Transformation

The above-mentioned variables were created after cleaning the data, merging tables such as disp, client, account, loans, cards, district, transactions and orders. A lot of the variables were created from the existing tables using grouping and aggregation. All the steps followed are mentioned below. The methods followed are listed based on the tables following the same order from which the above base table was created.

Custom Functions Created

- checkEmpty(df)
  This function checks for empty cells in all columns for dataframe and returns a dictionary containing column name and list of booleans containing empty rows.
  Parameters: dataframe

- fillEmpty(df, empty)
  This function fills the Empty rows with Value 'Other'.
  Parameters: dataframe, dict (obtained from the function checkEmpty())

- renameColumn(table, df)
  This function renames the column names with the prefix "table name".
  Parameters: name of the table, dataframe

Step 1: Preprocessing Client Table

- Renamed the columns in the client table using the function renameColumn().
- Extracted birth month, birth year and the day of birth from the birth number column of the clients table. Methods used were .astype(), .str, .isna(), .sum()
- Extracted the gender of the client from the birth month extracted from the birth number column.
- Calculated the age of the client by finding the difference between the birth year and 1997.
- Segmented clients into different age groups based on their age.
- Segmented clients into different categories based on their age such as young if the age is lesser than 21, Adult if age lesser than 55 and Senior Citizen if age higher than 55.
- Checked if there are any empty values in the table using the function checkEmpty(df) and filled the values using the function fillEmpty().

Step 2: Preprocessing Account Table

- Renamed the columns in the account table using the function renameColumn().
- Extracted account created month, account created year and the day of account created from the birth_date column of the accounts table. Methods used were .astype(), .str, .isna(), .sum()
- Calculated the length of relationship of the client by subtracting the year of account creation from 1997.
- Checked if there are any empty values in the table using the function checkEmpty(df) and filled the values using the function fillEmpty().

Step 3: Preprocessing Disposition Table

- Renamed the columns in the disposition table using the function renameColumn().
- Checked if there are any empty values in the table using the function checkEmpty(df) and filled the values using the function fillEmpty().

Step 4: Merging Account, Client and Disposition Table

- Merged the account table with the disposition table using left join with account_id as the primary key (account_disp_merge).
- Merged the account_disp_merge with the client table using left join (client_disp_merge).
- Created a new column called "acc_has_disp" with value as 1 if the disposition type is the disponent else 0.
- Checked for accounts with more than one owner and found that there are no accounts with more than one owner.

Step 5: Aggregating And Merging Disposition Data Based On Account

- Grouped the merged dataframe (client_disp_merge) from the above step using the account_id and found the sum of the variable "acc_has_disp". New dataframe created as account_client_count.
- Merged the dataframe account_client_count with client_disp_merge using left join using the account ID as the primary key. New dataframe created as "client_disp_merge"
- Subset the new dataframe such that the disposition type is owner.
- Dropped a few columns such as birth number and Client ID (duplicated).

Step 6: Processing District Table

- Checked if there are any missing columns in the table.
- Checked values in A12, A15 column prior to converting to float.
- Replaced the cells that had "?" as values with 0.
- Converted A12 and A15 to float.
- Since the columns in the district table are named from A1 to A16, renamed all the columns based on its meaning.
- Checked if there are any empty values in the table using the function checkEmpty(df) and filled the values using the function fillEmpty().
- Calculated the increase and decrease in the employment rate and the committed crimes between the years 1995 and 1996 and created four new variables.
- Calculated the total number of municipalities in a district and created a new variable.
- Calculated new variables such as number of employed population in the year 1996 and the number of employed urban population in the year 1996.

Step 7: Merging District Data To Client Account And Disposition Table

- Merged the client_disp_merge dataframe with the district table using left join on client's district ID. New dataframe client_disp_dist_merge is created.

Step 8: Processing Order Data

- Renamed the columns in the Orders table using the function renameColumn().
- Checked if there are any empty values in the table using the function checkEmpty(df) and filled the values using the function fillEmpty().

Step 9: Aggregating Results In Order Data

- Aggregated the orders table using the account ID and by using order k symbol to calculate the sum of order amount, count of orders and also calculated the same for individual k symbol values.
- Merged this aggregated result to the table client_disp_dist_merge using a left join using account ID and created a new table called “processed table”.
- Renamed the columns created using groupby with meaningful names.

Step 10: Processing Loan Table

- Renamed the columns in the loans table using the function renameColumn().
- Extracted loan issued month, loan issued year and the day of loan issued from the loan date column of the loan table. Methods used were .astype(), .str, .isna(), .sum()
- Created a new variable to replace the loan status codes with its meaning and to fill the missing values.
- Created a new variable called loan category to segment the loans based on the amount of loan as high, low and medium.
- Created the target variable “loan_granted” as 1.
- Checked if there are any empty values in the table using the function checkEmpty(df) and filled the values using the function fillEmpty().
- Merged the loan data to the “processed table” using left join with account ID as key.

Step 11: Preprocessing Card Table

- Renamed the columns in the Card table using the function renameColumn().
- Extracted card issued month, card issued year and the day of card issued from the loan card issued column of the card table. Methods used were .astype(), .str, .isna(), .sum()
- Calculated the length of relationship of the card by subtracting the year of card issuance from 1997.
- Created the target variable “card_issued_97” as 1.
- Checked if there are any empty values in the table using the function checkEmpty(df) and filled the values using the function fillEmpty().
- Merged the Card table with the processed table using left join.

Step 12: Preprocessing Transaction Table

- Renamed the columns in the transaction table using the function renameColumn().
- Checked if there are any empty values in the table using the function checkEmpty(df) and filled the values using the function fillEmpty().
- Renamed the columns to make the names more meaningful.
- Replaced the values of the columns trans_type and trans_operation with the meanings of the codes.
- Extracted transaction month, transaction year and the day of transaction from the loan trans_date column of the transaction table. Methods used were .astype(), .str, .isna(), .sum()

- Calculated the recency, frequency and monetary values of the transactions by grouping the values to the account ID and using methods such as max(), count(), sum() etc.
- Calculated the RFM score using the variables created from the above point and the formula.
- Segmented the accounts based on their RFM score such as low value if RFM < 30 and **Gold** if RFM is between 30 and 70 and Platinum if 70 to 100.
- Merged the transaction table and the aggregated values to the processed table.
- Created new columns such as cash_credit and cash_withdrawal by calculating the total amount of cash credited and withdrawn per account.
- Merged the above created variables with the processed table.
- Calculated the average transaction balance and average transaction amount per account and merged the columns to the processed table.
- Also calculated the count and sum for all the values in the trans operation, trans k symbol and transaction type by aggregating them with the account ID.
- The above created variables were merged with the processed table.

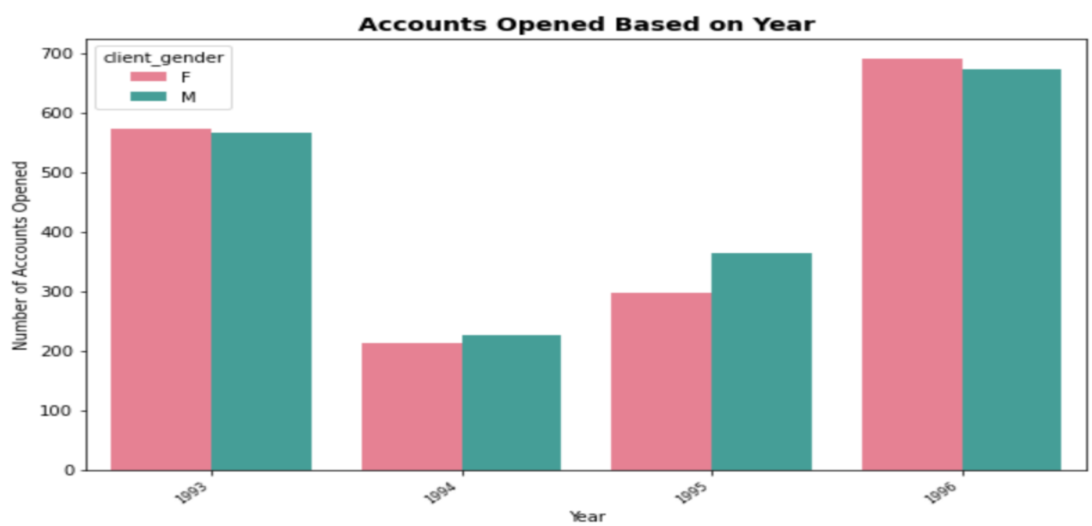
Step 13: Processed Table - Final Steps

- Renamed the variables created by aggregating columns with meaningful names.
- Exported the processed table as a .csv file.

Descriptive Analysis

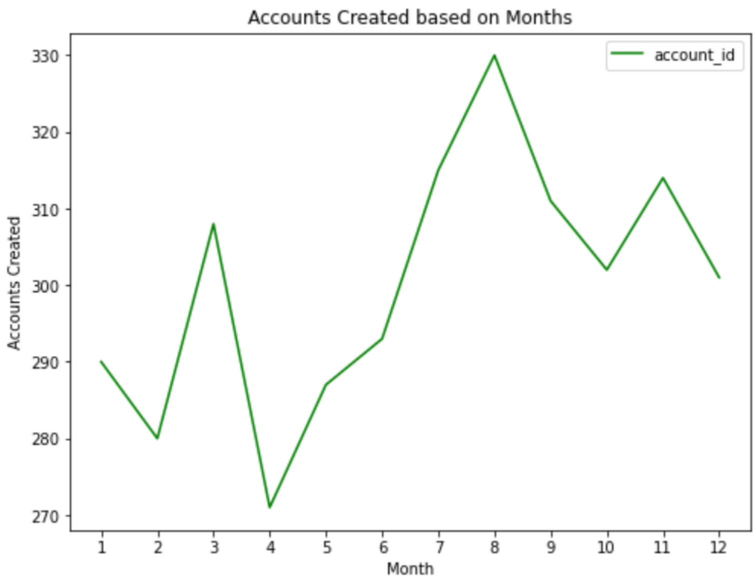
Account:

- Like any company, the customers are the heart of the banks as well. The accounts opened by clients are the first and foremost source for a bank to operate and make revenue. Hence it is important to analyze and find relatable information according to accounts and clients to improve the services and revenue.

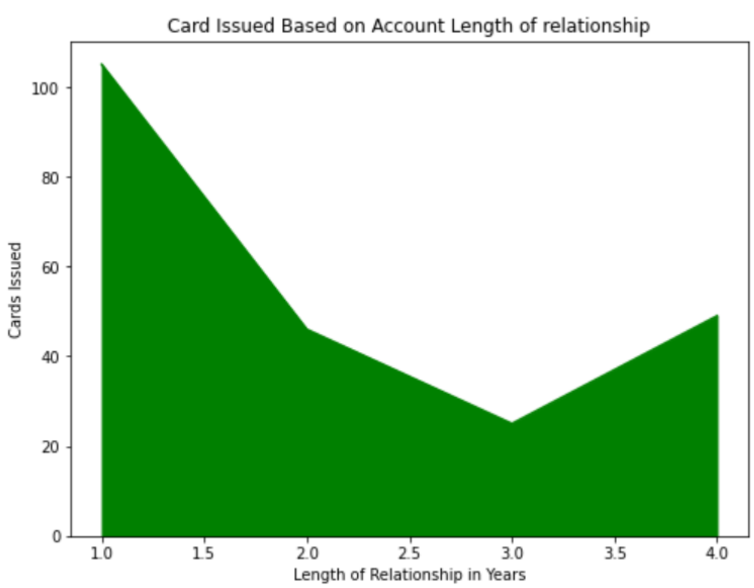


- Accounts, being one of the most important aspects of a bank, it is important to analyze its characteristics. The above plot displays the number of accounts opened in the bank in 4 years namely 1993, 1994, 1995 and 1996.

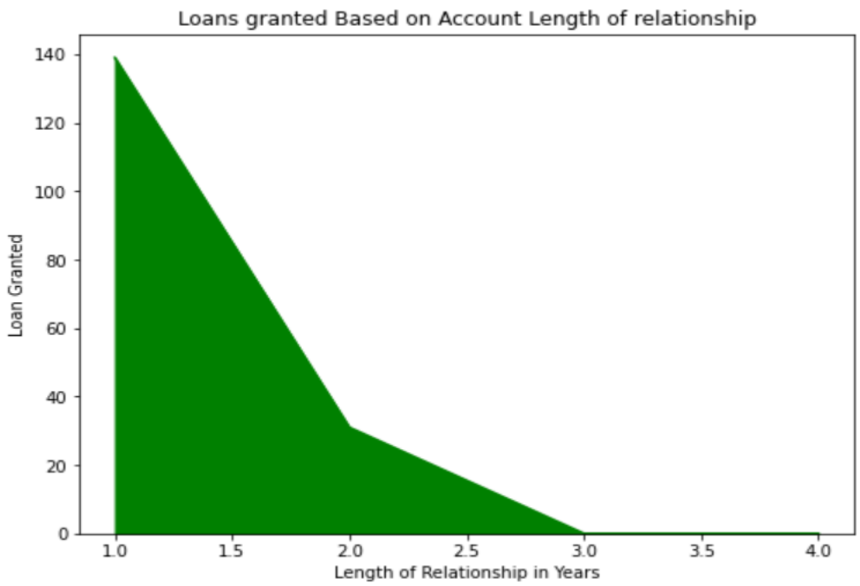
- It is evident that the number of accounts created declined for two consecutive years after 1993. However, the number of accounts created reached its peak in 1996.
- Also, the gender difference is not big and hence there are almost equal numbers of male and female accounts.



- The above chart compares the number of accounts created based on the month of account creation. The number of account creation is low during the month of April and it reaches its peak in the months of July and August. The third and fourth quarter of the year seems to be well performing when compared to the 1st and the 2nd quarter.
- The accounts created are the highest in July and August which are nothing but the early Summer Vacation period. People may consider opening their bank accounts during this period.
- July and August are also the months closer to the start of an academic year for students. Students may also be the reason behind the increase in account creation during this period.



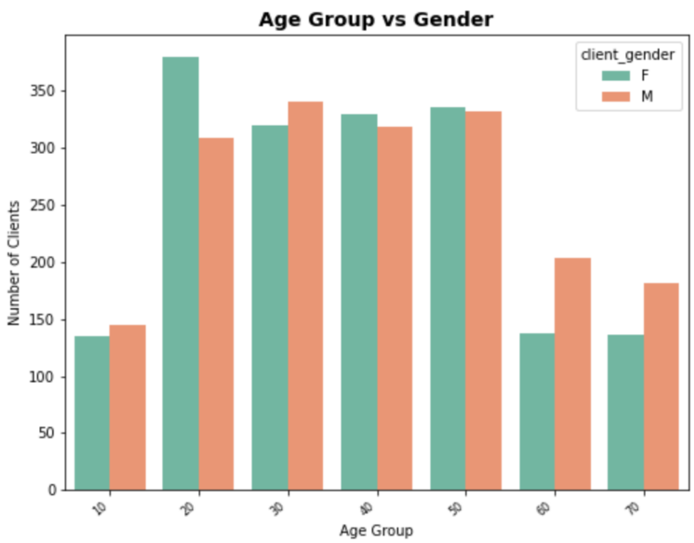
- This chart compares the amount of cards issued based on the length of relationship of the account. Most of the customers received their cards during the first year and the amount of cards issued declines in the next two years.
- There is a small rise in cards issued after 3 years of relationship. This may also be due to the card issued first getting old in three years and the client may request for the same or a different card again.



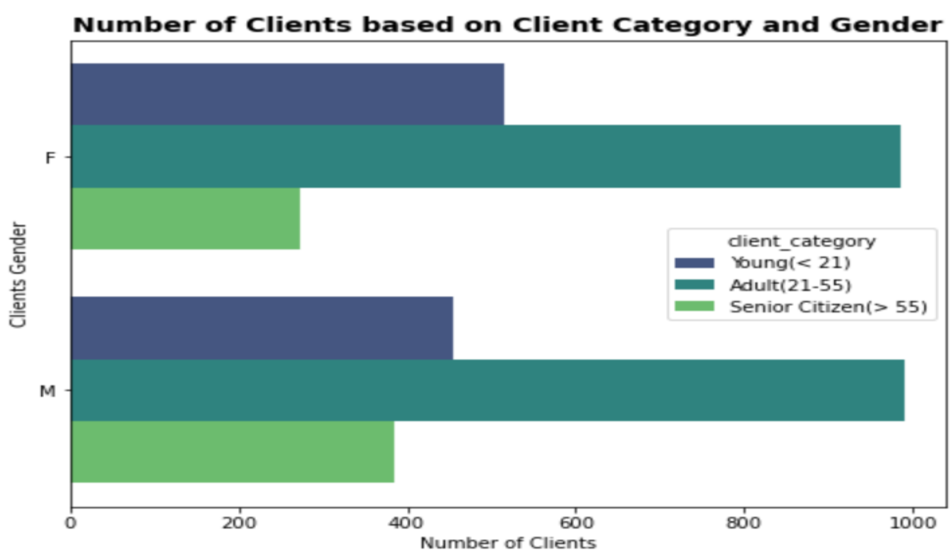
- This chart is similar to the above chart, but instead of comparing cards issued, it compares loans granted based on the length of the relationship. It is clearly visible that most of the loans were granted in the first two years of their relationship with the bank.
- There is also a possibility that these clients opened their account in this bank because of good loan plans available.

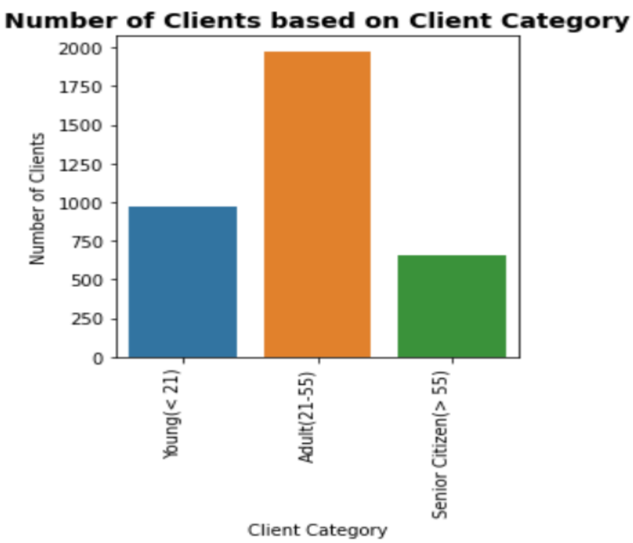
Clients

Clients are the customers of this business who open accounts with the bank. A client can have multiple accounts, or a single account can have multiple clients.

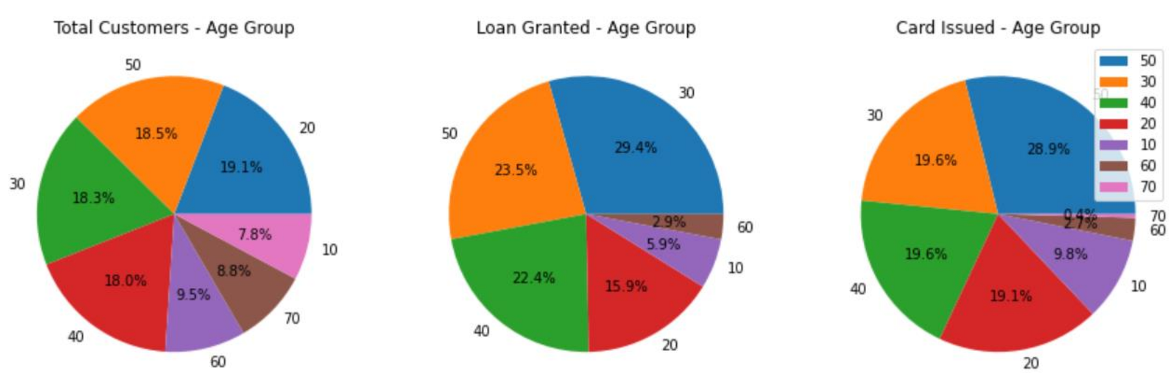


- The above bar graph displays the distribution of the clients based on their age groups. The graph shows that the majority of the clients are in the age groups 20,30,40 and 50 in almost equal distribution.
- There are more young female clients and elder male.





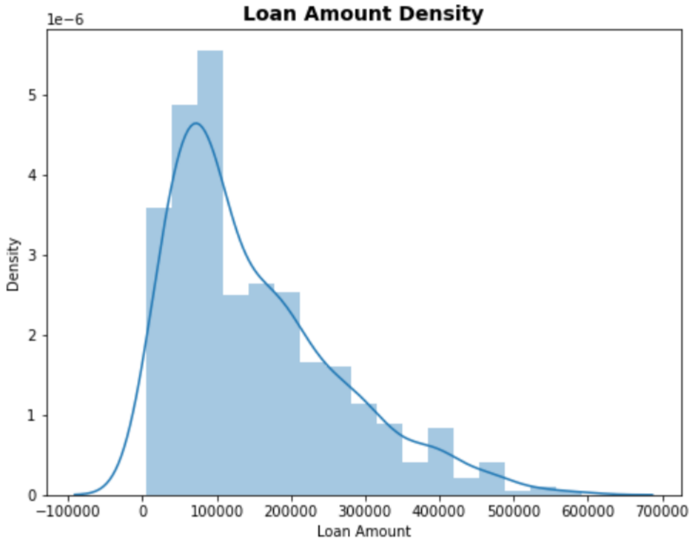
- Client category is one important subset that can be used for a variety of internal analysis within the bank. The above graph shows the number of clients based on the client category and the gender within these categories.
- Similar to the insights from the previous charts, it is evident that the adult category that is age greater than 21 and less than 55 is the largest group of clients.
- When compared with the client gender, the adult clients are almost equal in number, but there are more females in the young category and more males in the Senior Citizen category.



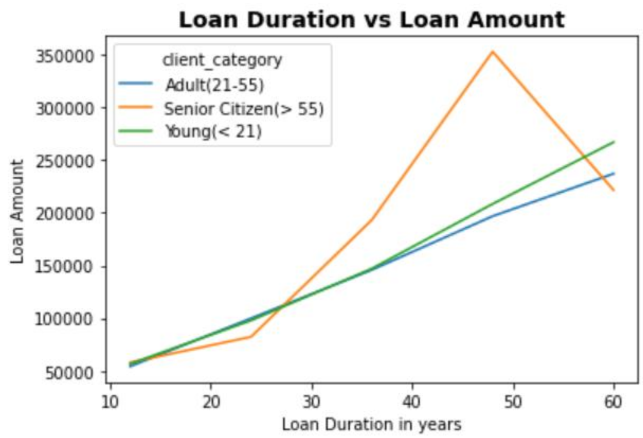
- These three pie charts indicate the total customers based on the age groups and the percentage of loans being granted and the cards being issued based on the groups.
- The first pie chart shows that the groups 20, 30, 40 and 50 are almost equal in terms of clients with minor deviations.
- The second chart shows that the clients in the group 30 were granted the most number of loans followed by 50, 40 and 20. The least groups were 60 and 10.
- The third chart shows that the group with the highest number of cards issued was 50. The lowest group was 70 followed by 60 and 10.

Loan:

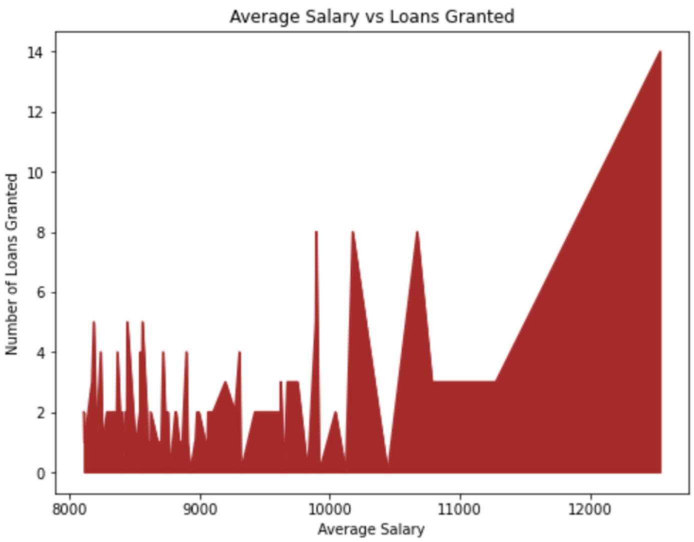
Loans are the most important revenue stream for banks. All companies sell their products to make money and the product for banks is the money. And hence, it is important to analyze the loans to predict, improve, and increase the efficiency in the business model.



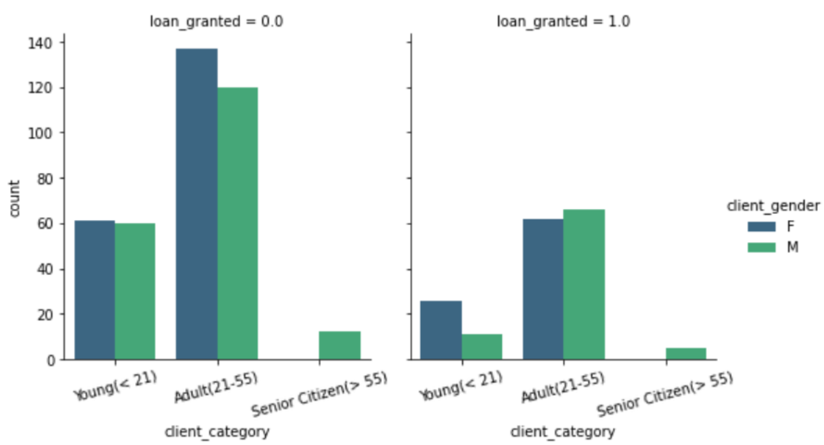
- The plot above provides the density of the loan amount being granted to the clients in this bank. From the plot, we can see that the amount of most of the loans granted were around 100000.
- Higher the loan amount, the lesser the number granted.



- This line plot compares the duration provided to repay the loan and the loan amount that was granted with respect to the client category.
- The loan duration increases with increase in the loan amount and the graph is almost similar for the young and adult client category.
- For the senior citizen category, the data suggests that the loan duration was smaller around 50 years for a loan of 350000.



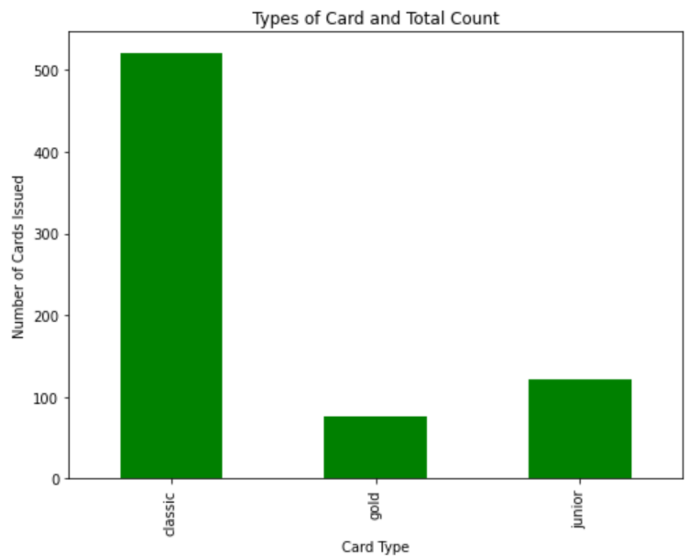
- In order for the bank to trust the clients to provide loans, the salary of the client plays a major role.
- The above area plot compares the number of loans granted and the average salary of the clients.
- From the plot it is evident that the number of loans granted are higher for salaries above 11200.
- The loans were also granted for salaries ranging from 8200 but the number of loans granted are lesser. It proves that higher the average salary, most likely is the loan being granted.



- It is also important to see the customers who have received loans from the bank. The above diagram compares the category of clients who have received loans and not received loans.
- Adult category clients are the clients that have received more loans. The male clients receiving loans are higher than the female clients receiving the loans.
- Adult category clients are followed by the young category in which the female clients are higher than the male clients receiving loans.
- The Senior citizen category is the lowest among the clients receiving loans from the bank.

Cards

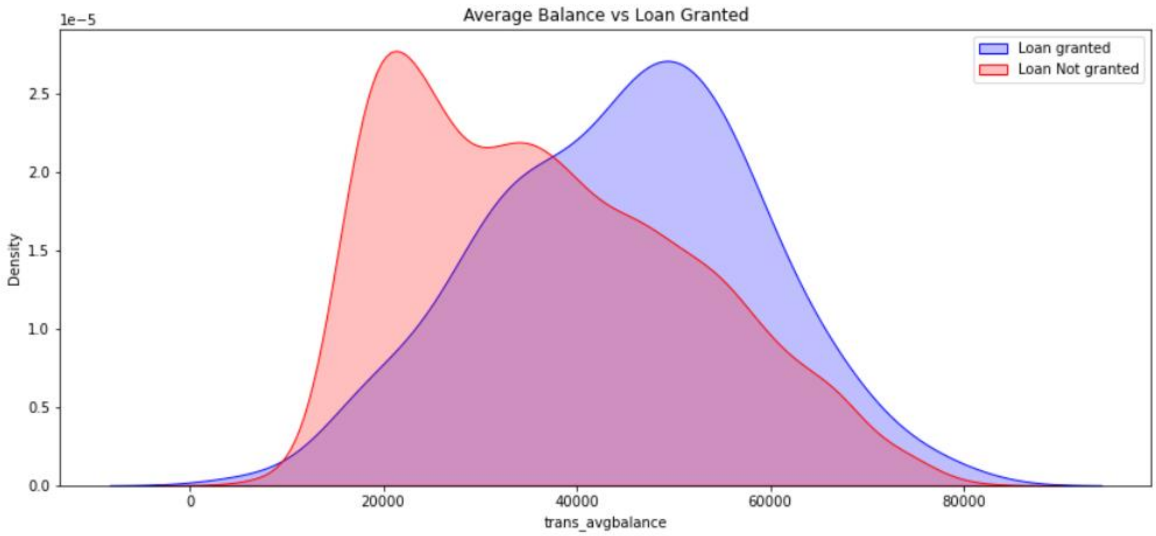
Credit cards and debit cards that banks issue are also a good stream of revenue for banks. Interest fees, credit card fee, merchant fees are some of the few revenue streams generated using cards. To get a credit card, the bank should approve the client and agree to let the client use its money to make purchases on the promise that they pay it back. Hence it is important to analyze the clients to predict the potential clients to trust and issue the cards.



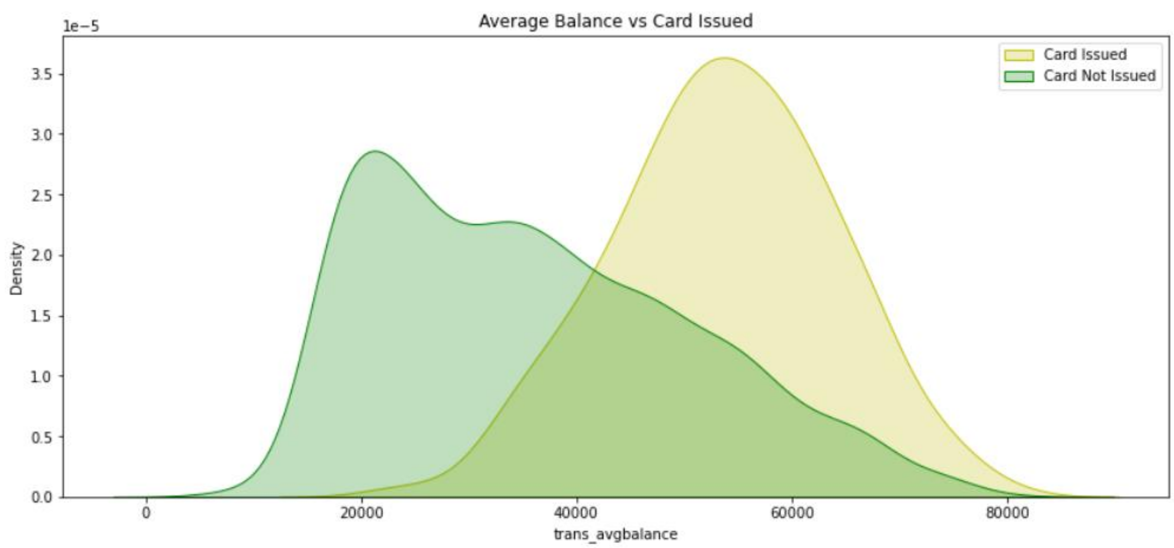
Transactions

Transaction data gives the business, few major customer details and the usage and monetary details and helps us with deeper insights which can be considered as decision makers while Granting Loan and taking any customer centric business decisions.

- **Transaction Balance** has impact on client getting their Loan Approved and get the credit card issued.
- The below two plots suggests, as the average balance decreases, the chance of client getting their Loan granted decreases and vice-versa
- More the average balance, higher the probability of getting the credit card issued.



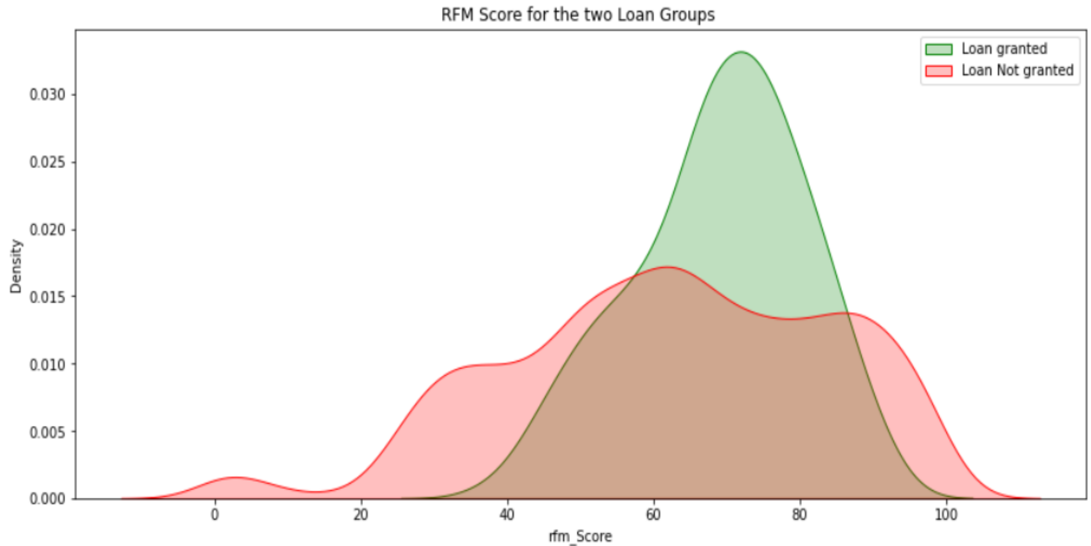
- More the average balance, greater is the chance of getting a credit card.
- It can be a parameter for managers or for loan.



- For the accounts having transaction balance ore than 40000, credit card was never rejected, and this can be a parameter for the business to consider while issuing the credit card.

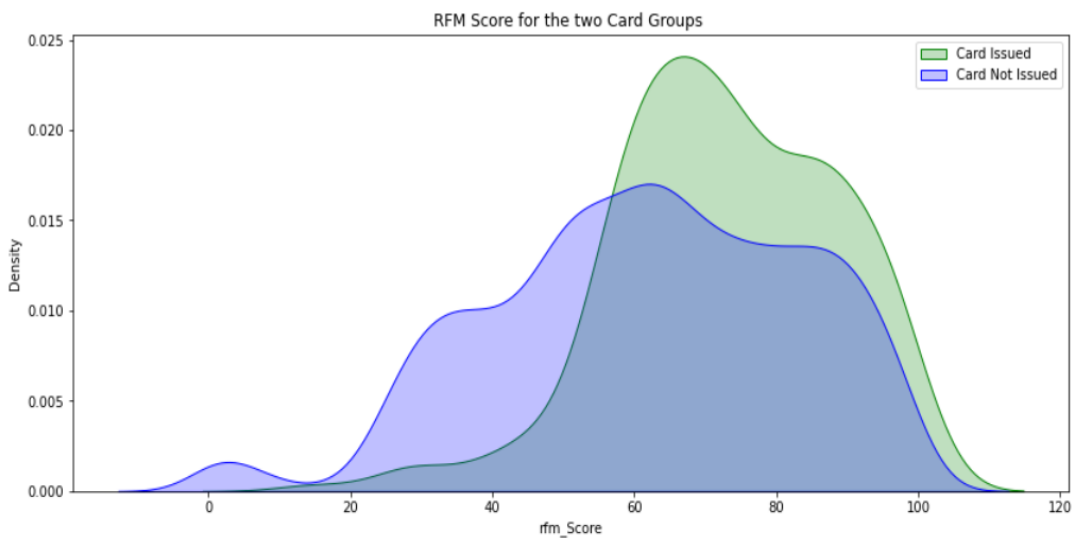
Recency, Frequency & Monetary (rfm_score) : This score is calculated on how recently & Frequently the client has made the transactions and what is the average monetary value of the account after transactions.

- RFM Score gives major meaningful insights on the customer operation patterns and the monetary status, which in turn helps the business to consider and decide while granting the loan or issuing the credit card.
- Accounts having RFM Score above

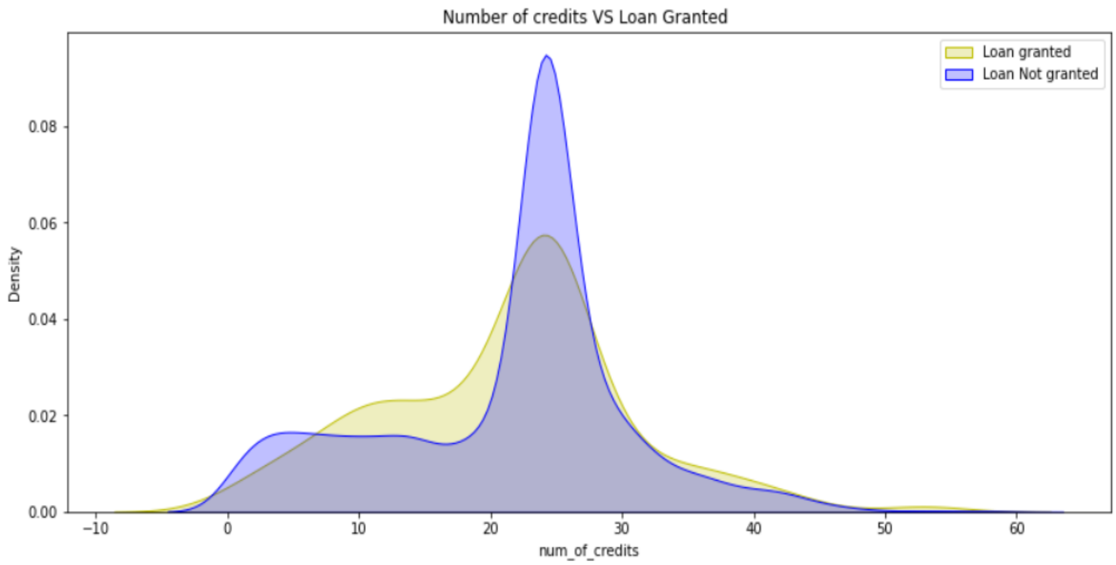


- It's evident from the plot that as the as the RFM increases there is a higher chance of getting Loan granted

- Similarly, for the Credit cards, as the Score increases, there is a higher chance of issuing the credit card.

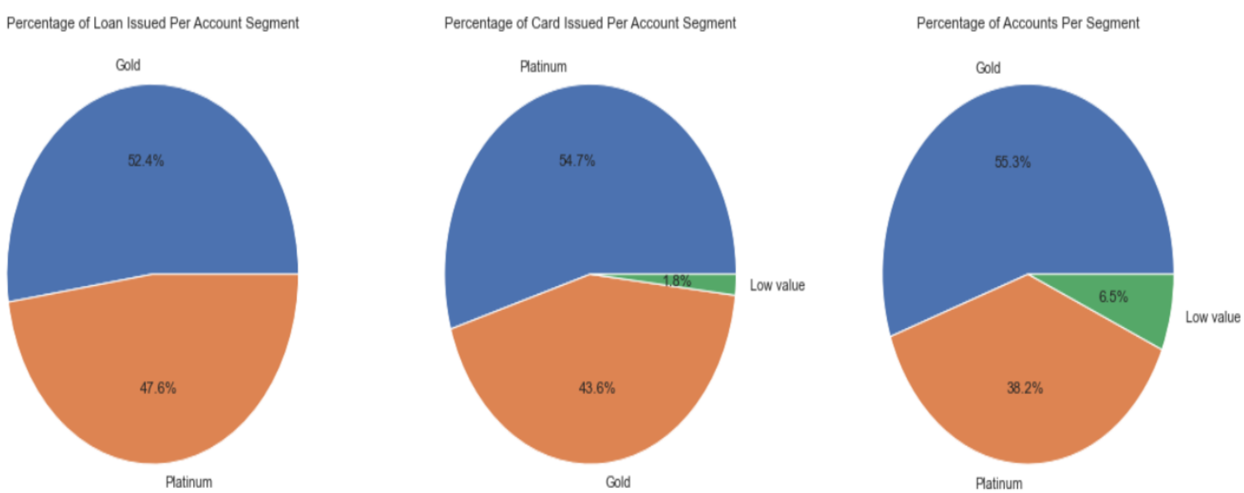


- Number of credits to the bank account has average impact on whether a customer will get Moan granted or not.

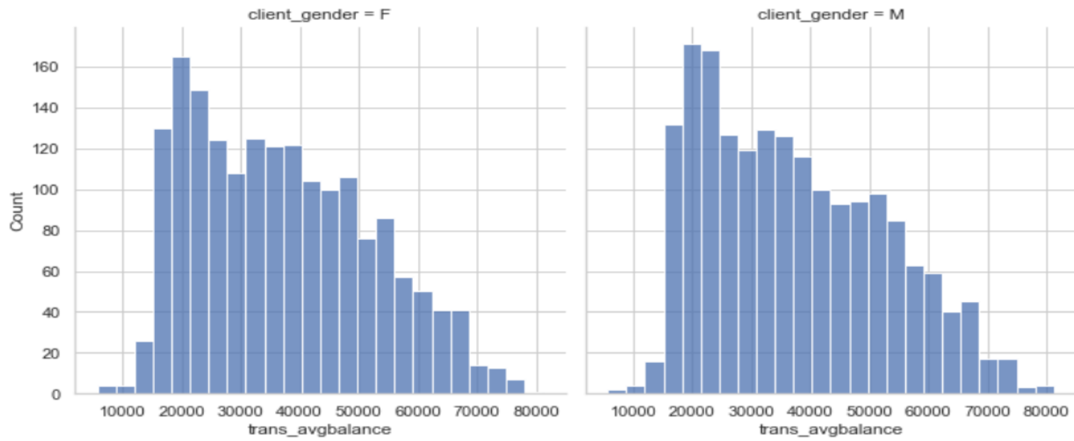


Account Segment : Account segment gives us the info on which segment the client falls into based on rfm Score. Accounts were classified into three segments , Premium (RFM > 70) , Gold (RFM 30-70) and Low Value (rfm <30).

- we can see from the below pie plots, Low value customers (whose RFM Score is less than 30 were never granted a Loan
- Major part of clients who got their Loan Approved and the card issued were from the premium Segment.



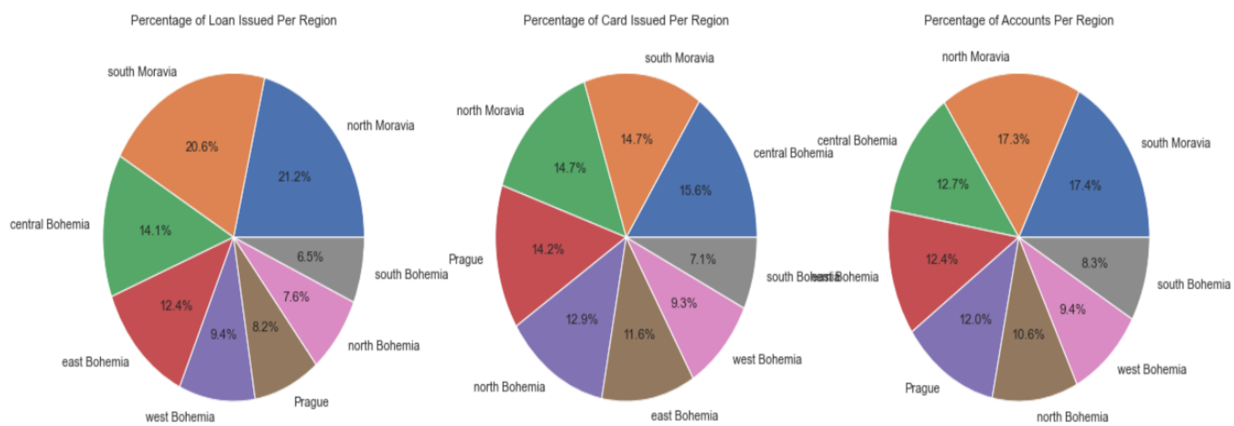
- We can see the gender distribution from the below chart against their transaction balances. On average both Male and female have equal amounts after transactions in their accounts.



District:

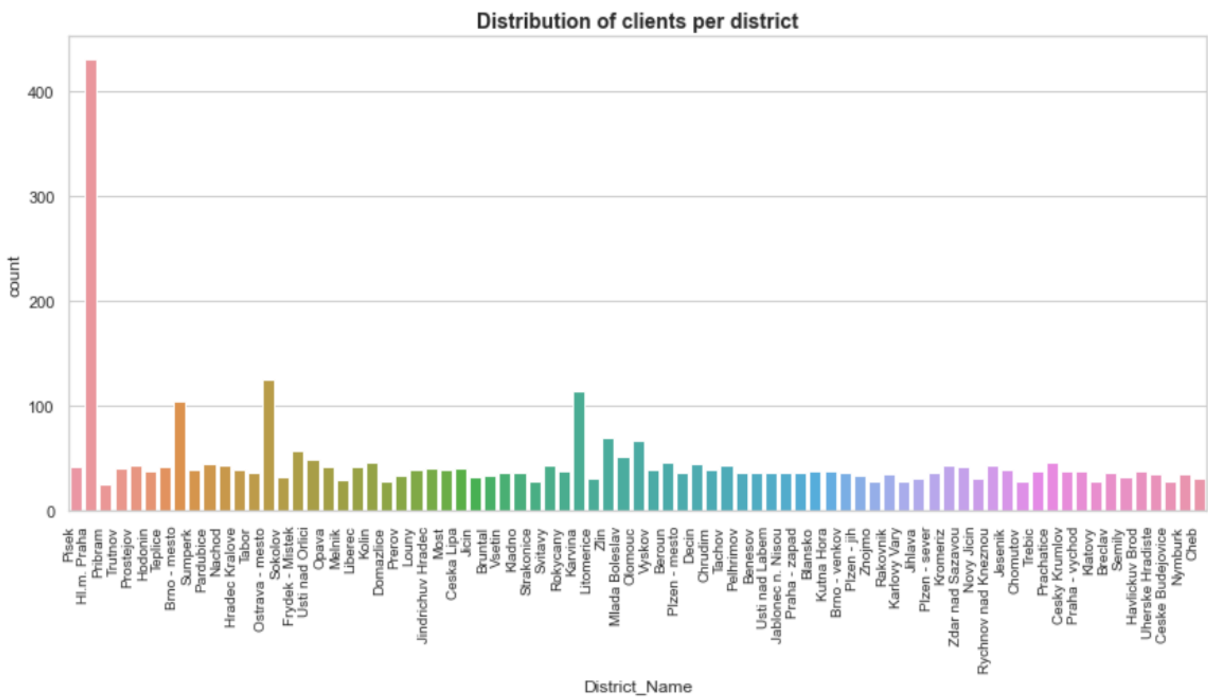
Region : Region wise data, helps us to understand the performance of the business as well as the impacting parameters based on the Region.

- South Moravia and North Moravia regions has majority of the Accounts and also major percentage of the Loans Approved and Cards Issued
- South Boehmia would be the least Performing region in terms of All the three factors



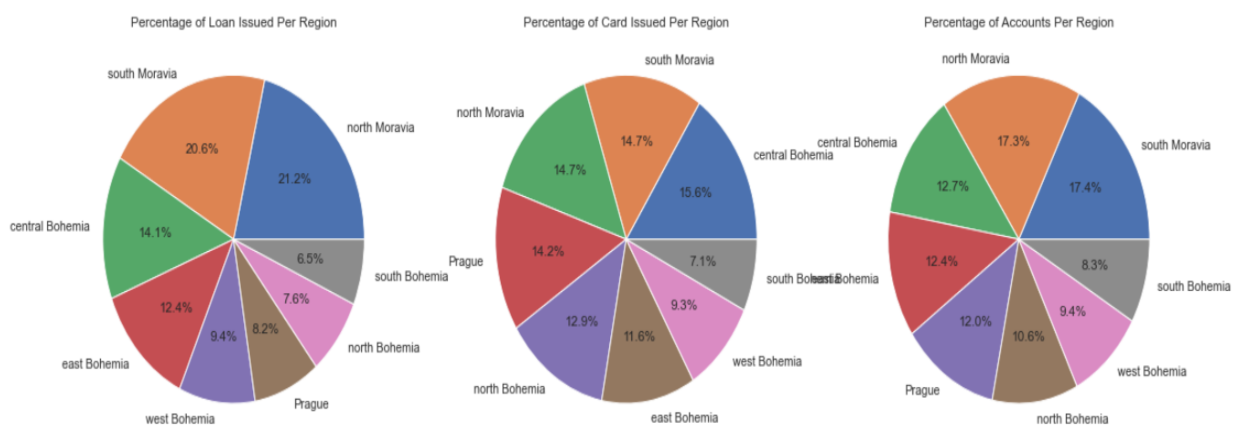
District Data of client gives us the one of the important data that any business needs which is the geography-based statistics and customer details. District data also gives us

- Praha region has the highest number of clients and seems the major percentage or clients comes from this region too.

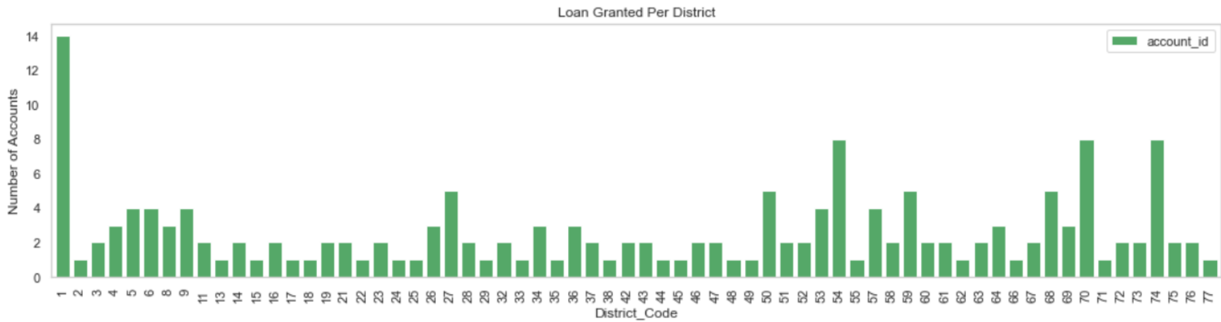


Region : Region wise data, helps us to understand the performance of the business as well as the impacting parameters based on the Region.

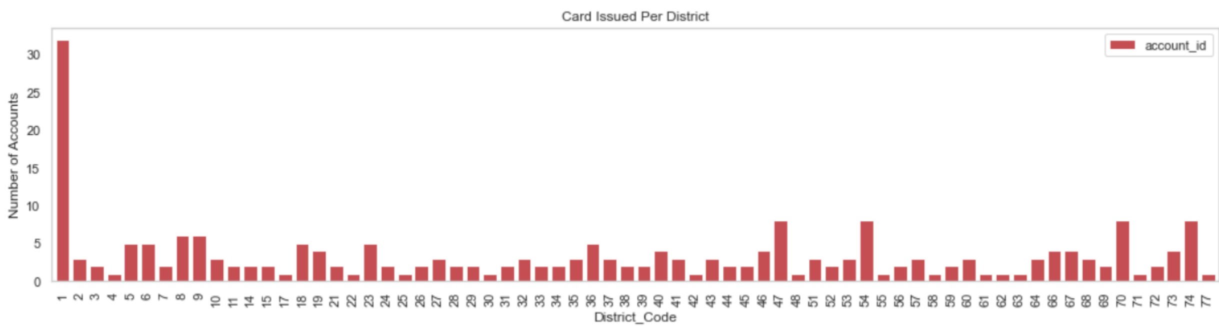
- South Moravia and North Moravia regions has majority of the Accounts and also major percentage of the Loans Approved and Cards Issued
- South Boehmia would be the least Performing region in terms of All the three factors



- Number of accounts per district can be seen from the below plot, where highest number of loans approved per district is 14



- Similarly, district can be seen from the below plot, where highest number of cards issued per district is 14



Summary for the Berka Employees:

While Approving Loan:

- Higher the loan amount, the lesser the number granted.
- Clients having Average salary were always granted the Loan and above 11000 was never Rejected
- Maximum approved Loan Amount is 100000
- For Senior citizens, the duration of Loan Amount is always Low compared to others
- Ideal RFM Score is the Range of 60 -80 as clients fall in this range were never rejected for their Loans
- Clients who have average Balance greater than 20000 were granted Loans as per the past data
- South Moravia is the Region with the Most number of Loans Approved while least being South Bohemia which is worth Noting while issuing Loans to the clients from this region

While Issuing Cards:

- Classic cards are the most issued type of cards followed by Gold
- Clients in the Age group 20 – 40 were the ideal age group likely to get the credit cards
- Ideal RFM Score for to issue credit cards to the client is 40 – 75
- Clients who have Average Balance less than 10000 are less likely to get the credit card based on the statistics
- Clients having RFM Score Less than 20 were hardly given Cards and hence can have a look at the profile before issuing Credit cards.
- South Moravia is the top region with most number of cards issued followed by north Moravia and South Moravia while least being the

BERKA ANALYSIS