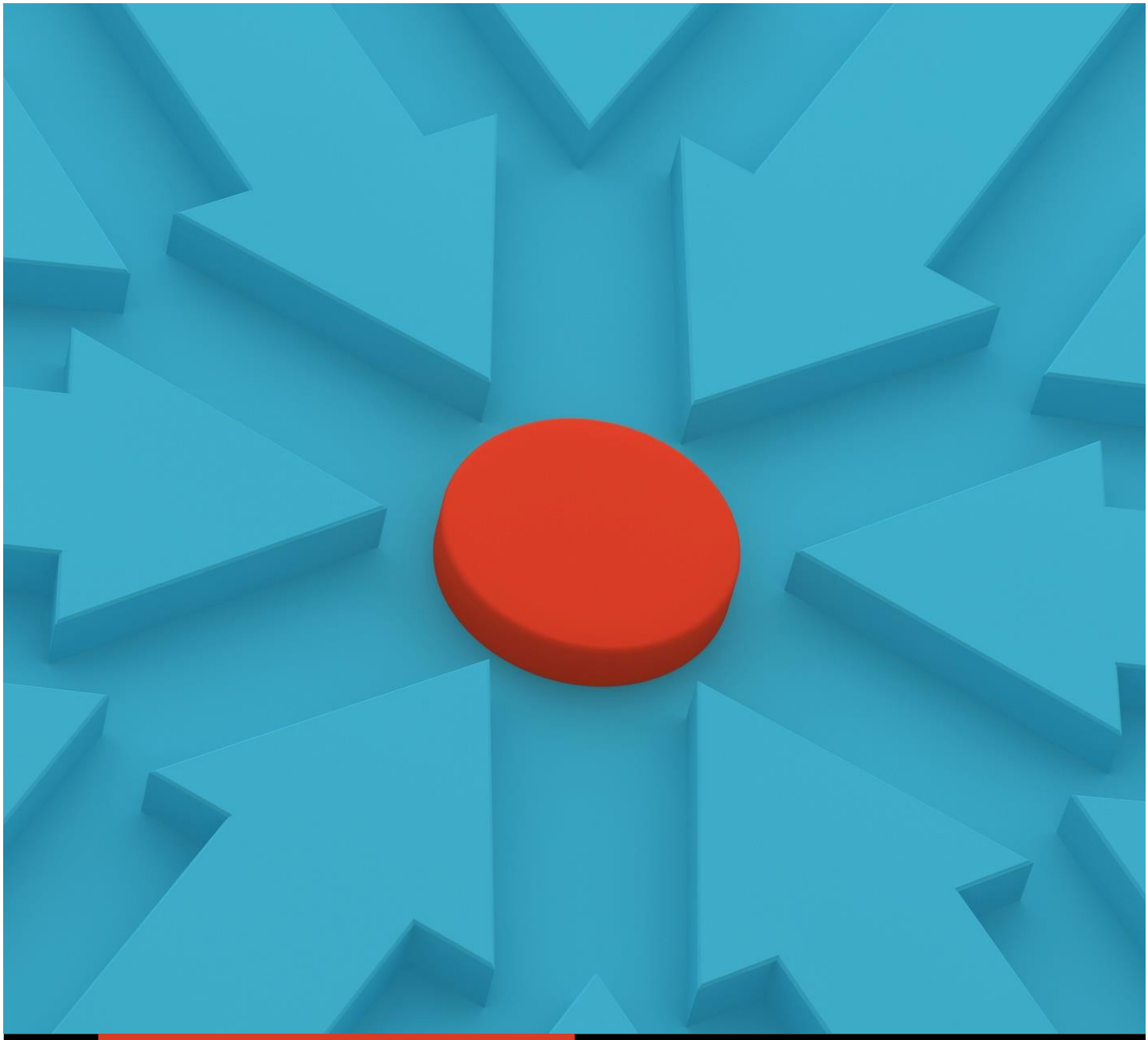


CAPSTONE BUSINESS REPORT

MRUDHULAA P V

PGP DSBA Online – March 22

12/02/2023

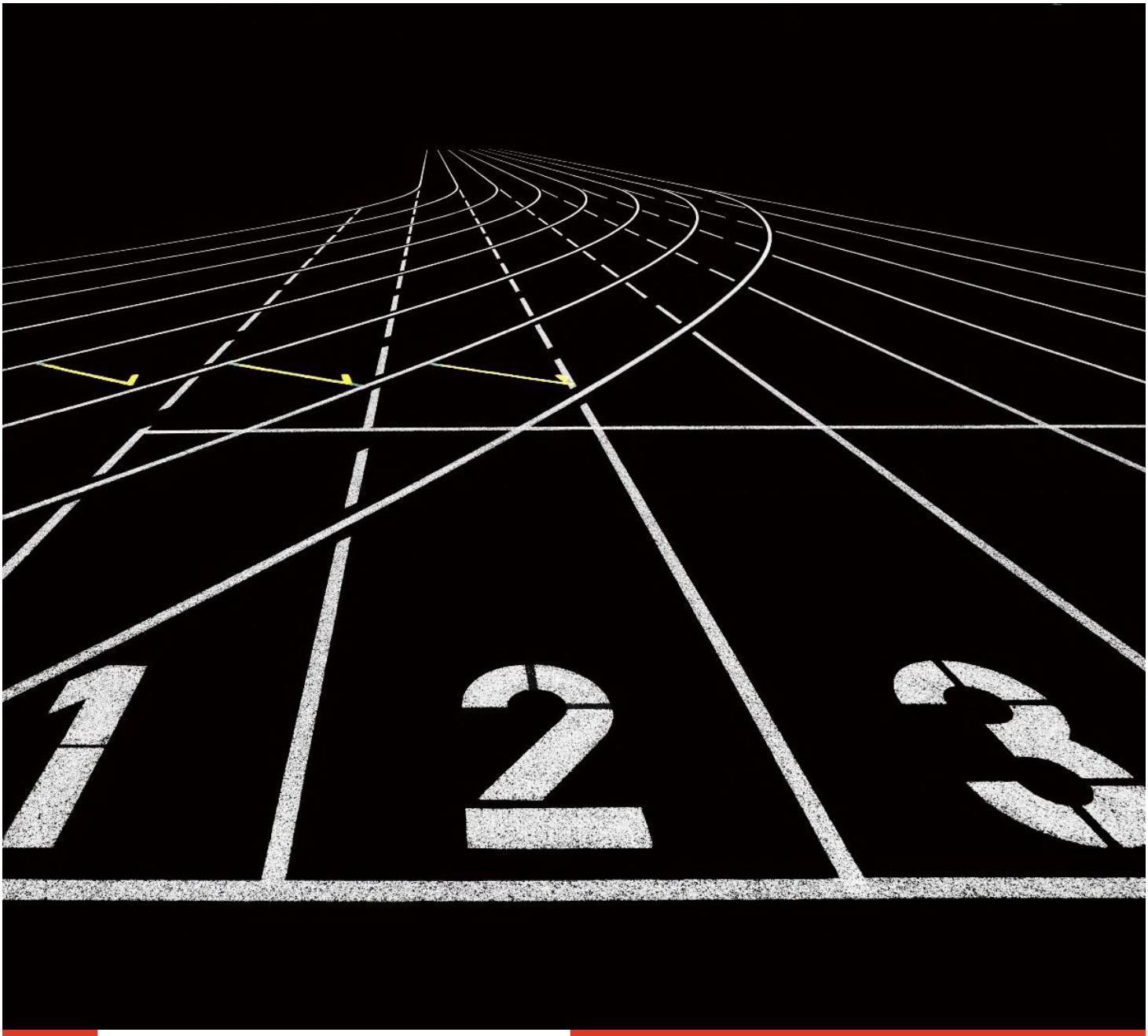


TABLE OF CONTENTS

List of figures

List of tables

Customer Churn

1) Problem Understanding

Problem Statement

Due to the intense rivalry now present in the industry, it is difficult for DTH service providers to keep their existing clientele. As a result, the DTH provider wants to create a model that would allow them to segment their offers and do account churn prediction. Because one account can have several customers, account churn is a significant issue with this organization. Therefore, by losing one account, the business could lose multiple clients.

We were tasked with creating a churn prediction model for this business and making business suggestions regarding the campaign. The campaign or model must be distinct and well-defined when offers are put forth. In order for the firm to avoid revenue losses and, on the other hand, be able to keep clients, the suggested offers should have a win-win situation for both the company and the customers.

Figure 1. Customer churn

Need of the study/project

For the client to plan for the future in terms of product creation, sales, or rolling out new offerings for different client segments, this study/project is absolutely necessary. The project's results will make it clear where the company is at this point and how much risk it is capable of accepting. It will also indicate the organization's potential for the future, how to improve it, how to prepare for it better, and how to assist them keep customers over the long term.

Understanding business/social opportunity

This case study features a DTH provider where each client is given a special account ID. A single account ID can hold several customers (such as those on a family plan), regardless of gender or marital status, and customers have a choice of several payment methods. Customers are once again divided into different plan categories based on their usage and the type of device they use (computer or mobile). In addition, customers receive cashbacks when paying their bills.

The success of the company depends on the loyalty and retention of its clients, which are attained by offering high-caliber services with added benefits. Additionally, implementing different promotional and holiday deals may assist a company in attracting new clients while retaining existing ones.

We can draw the conclusion that a customer retained is a consistent source of revenue for an organisation, a customer added is a new source of revenue for an organisation, and a customer lost will have a negative impact because a single account ID holds multiple customers, meaning that closing one account ID results in the loss of multiple customers.

Since nearly every household needs a DTH connection, there is a big opportunity for the business as well as increased demand and competition. The question of how a business may differentiate itself from rivals and what factors are essential to winning over customers' loyalty and keeping them on board emerges. The top player in the market will be determined by all these social duties.

2) Data Report

Dataset of problem: Customer Churn Data

Data Dictionary:

Description	Variable
account unique identifier	AccountID
account churn flag (Target)	Churn
Tenure of account	Tenure
Tier of primary customer's city	City_Tier
How many times all the customers of the account has contacted customer care in last 12months	CC_Contacted_L12m
Preferred Payment mode of the customers in the account	Payment
Gender of the primary customer of the account	Gender
Satisfaction score given by customers of the account on service provided by company	Service_Score
Number of customers tagged with this account	Account_user_count
Account segmentation on the basis of spend	account_segment
Satisfaction score given by customers of the account on customer care service provided by company	CC_Agent_Score
Marital status of the primary customer of the account	Marital_Status
Monthly average revenue generated by account in last 12 months	rev_per_month
Any complaints has been raised by account in last 12 months	Complain_l12m
revenue growth percentage of the account (last 12 months vs last 24 to 13 month)	rev_growth_yoy
How many times customers have used coupons to do the payment in last 12 months	coupon_used_l12m
Number of days since no customers in the account has contacted the customer care	Day_Since_CC_connect
Monthly average cashback generated by account in last 12 months	cashback_l12m
Preferred login device of the customers in the account	Login_device

Figure 2. Data dictionary

Dataset sample

	0	1	2	3	4
AccountID	20000	20001	20002	20003	20004
Churn	1	1	1	1	1
Tenure	4	0	0	0	0
City_Tier	3.0	1.0	1.0	3.0	1.0
CC_Contacted_LY	6.0	8.0	30.0	15.0	12.0
Payment	Debit Card	UPI	Debit Card	Debit Card	Credit Card
Gender	Female	Male	Male	Male	Male
Service_Score	3.0	3.0	2.0	2.0	2.0
Account_user_count	3	4	4	4	3
account_segment	Super	Regular Plus	Regular Plus	Super	Regular Plus
CC_Agent_Score	2.0	3.0	3.0	5.0	5.0
Marital_Status	Single	Single	Single	Single	Single
rev_per_month	9	7	6	8	3
Complain_ly	1.0	1.0	1.0	0.0	0.0
rev_growth_yoy	11	15	14	23	11
coupon_used_for_payment	1	0	0	0	1
Day_Since_CC_connect	5	0	3	3	3
cashback	159.93	120.9	NaN	134.07	129.6
Login_device	Mobile	Mobile	Mobile	Mobile	Mobile

Table 1. Data description

Dataset has 11260 rows and 19 columns in total. From the data, there are 5 columns which are of type float, 2 columns of type integer and 12 columns of type string. Among these 19 columns, 1 column is known to be its target variable

Understanding how data was collected in terms of time, frequency and methodology

- For a random sample of 11,260 distinct account IDs, information about gender and marital status has been gathered.
- We can infer that the data has been collected over the past 12 months by looking at the

variables "CC Contacted L12m," "rev per month," "Complain L12m," "rev growth yoy," "coupon Used L12m," "Day Since CC Connect," and "cashback L12m."

- 19 variables make up the data: 18 independent variables and the target variable, which indicates whether or not a customer churned.
- The information consists of the services that clients use, their preferred method of payment, and also their basic personal information.

Renaming variables:

Variable renaming is done. Following are the changes made:

AccountID ----------→ account_id,

churn ----------→ churn,

Tenure ----------→ account_tenure,

City_Tier ----------→ city_tier,

CC_Contacted_LY ----------→ cust_care_contacts_12m,

Payment ----------→ payment_method,

Gender ----------→ gender,

Service_Score ----------→ service_score,

Account_user_count ----------→ customers_per_account,

CC_Agent_Score ----------→ cc_agent_score,

Marital_Status ----------→ marital_Status,

rev_per_month ----------→ revenue_per_month,

Complain_ly ----------→ account_complaints_12m,

coupon_used_for_payment ----------→ coupons_used,

Day_Since_CC_connect ----------→ days_since_cc_contact,

Login_device ----------→ login_device

Checking data info:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11260 entries, 0 to 11259
Data columns (total 19 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   account_id              11260 non-null  int64
 1   Churn                   11260 non-null  int64
 2   account_tenure          11158 non-null  object
 3   city_tier               11148 non-null  float64
 4   cust_care_contacts_12m  11158 non-null  float64
 5   payment_method          11151 non-null  object
 6   gender                  11152 non-null  object
 7   service_score           11162 non-null  float64
 8   customers_per_account   11148 non-null  object
 9   account_segment         11163 non-null  object
 10  cc_agent_score          11144 non-null  float64
 11  marital_Status          11048 non-null  object
 12  revenue_per_month       11158 non-null  object
 13  account_complaints_12m  10903 non-null  float64
 14  rev_growth_yoy          11260 non-null  object
 15  coupons_used            11260 non-null  object
 16  days_since_cc_contact   10903 non-null  object
 17  cashback                10789 non-null  object
 18  login_device            11039 non-null  object
dtypes: float64(5), int64(2), object(12)
memory usage: 1.6+ MB
```

Fig.3 Details of the dataset columns

OBSERVATION:

Dataset has 11260 rows and 19 columns in total. From the data, there are 5 columns which are of type float, 2 columns of type integer and 12 columns of type string. Among these 19 columns, 1 column is known to be its target variable

Data Description

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
account_id	11260.0	NaN	NaN	NaN	25629.5	3250.62635	20000.0	22814.75	25629.5	28444.25	31259.0
Churn	11260.0	NaN	NaN	NaN	0.168384	0.374223	0.0	0.0	0.0	0.0	1.0
account_tenure	11158.0	38.0	1.0	1351.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
city_tier	11148.0	NaN	NaN	NaN	1.653929	0.915015	1.0	1.0	1.0	3.0	3.0
cust_care_contacts_12m	11158.0	NaN	NaN	NaN	17.867091	8.853269	4.0	11.0	16.0	23.0	132.0
payment_method	11151	5	Debit Card	4587	NaN	NaN	NaN	NaN	NaN	NaN	NaN
gender	11152	4	Male	6328	NaN	NaN	NaN	NaN	NaN	NaN	NaN
service_score	11162.0	NaN	NaN	NaN	2.902526	0.725584	0.0	2.0	3.0	3.0	5.0
customers_per_account	11148.0	7.0	4.0	4569.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
account_segment	11163	7	Super	4062	NaN	NaN	NaN	NaN	NaN	NaN	NaN
cc_agent_score	11144.0	NaN	NaN	NaN	3.066493	1.379772	1.0	2.0	3.0	4.0	5.0
marital_Status	11048	3	Married	5860	NaN	NaN	NaN	NaN	NaN	NaN	NaN
revenue_per_month	11158.0	59.0	3.0	1746.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
account_complaints_12m	10903.0	NaN	NaN	NaN	0.285334	0.451594	0.0	0.0	0.0	1.0	1.0
rev_growth_yoy	11260.0	20.0	14.0	1524.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
coupons_used	11260.0	20.0	1.0	4373.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
days_since_cc_contact	10903.0	24.0	3.0	1816.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
cashback	10789.0	5693.0	155.62	10.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
login_device	11039	3	Mobile	7482	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Fig. 4 Summary of the data

This illustrates how different statistical measurements across variables vary, indicating that each variable is distinct and diverse.

Checking null values

```
account_id                  0
Churn                       0
account_tenure            102
city_tier                 112
cust_care_contacts_12m    102
payment_method            109
gender                    108
service_score              98
customers_per_account     112
account_segment            97
cc_agent_score            116
marital_Status            212
revenue_per_month         102
account_complaints_12m    357
rev_growth_yoy              0
coupons_used                0
days_since_cc_contact     357
cashback                  471
login_device              221
dtype: int64
```

Fig. 5 Showing null values in dataset

All other variables have null values except for "account_id," "Churn," "rev_growth_yoy," and "coupon_used"

Data does not contain any duplicates as of now. But, on dropping account_id variable we will see duplicates to be 259, which is being covered in the later part of this report.

Understanding of attributes (variable info, renaming if required)

There are 18 attributes in this project that affect the target variable. Let's look into those variables individually:

1. account_id - This variable denotes a unique ID that denotes a unique consumer. This data is of the integer data type and contains no null values.
2. churn – It is the target variable which indicates whether or not a customer has left. This has no null values and is categorical in nature. "0" stands for "NO," and "1" for "YES."
3. account_tenure - This shows how long the account has been open overall. There are 102 null values in this continuous variable.
4. city_tier - Based on the city where the major client resides, this variable divides the consumer into three groups. There are 112 null values in this category variable.
5. cust_care_contacts_12m - This variable shows how many times all of the account's customers have been in touch with customer service over the past 12 months. There are 102 null values for this

continuous variable.
6. payment_method: This variable indicates the customer's preferred method of paying their bills. This has 109 null values and is categorical in nature.
7. gender - The primary account holder's gender is indicated by this attribute. This has 108 null values and is categorical in nature.
8. service_score - Customer ratings based on the quality of the company's service. There are 98 null values for this category variable.
9. customers_per_account - This variable indicates how many customers have an account_id associated to them. This is ongoing and contains 112 null values.
10. account segment - This variable divides client into various segments based on how much money they spend and how much money they generate. This has 97 null values and is categorical in nature.
11. cc_agent_score – Customer ratings based on the company's customer service representative's performance. This variable's 116 null values make it a categorical one.
12. marital_Status - This displays the primary account holder's marital status. This has 212 null values and is categorical in nature.
13. revenue_per_month - This shows the average monthly revenue for each account ID for the previous 12 months. There are 102 null values for this continuous variable.
14. account_complaints_12m - Indicates if a customer has filed a complaint in the last 12 months. This has 357 null values and is categorical in nature.
15. rev growth yoy - This variable compares revenue growth over 24 to 13 months to that over 12 months. This has no null values and is continuous in nature.
16. coupons_used - This counter shows how frequently customers have used promotional codes to pay their bills. This has no null values and is continuous in nature.
17. days_since_cc_contact - This shows how many days have passed since the consumer last contacted customer service. The service is better when the number of days is more. This has 357 null values and is continuous in nature.
18. cashback l12m - This variable shows how much cash back the consumer received after paying their bill. This has 471 null values and is continuous in nature.
19. login_device - This variable indicates if a consumer is using a phone or a computer to access the services. This has 221 null values and is a category variable.

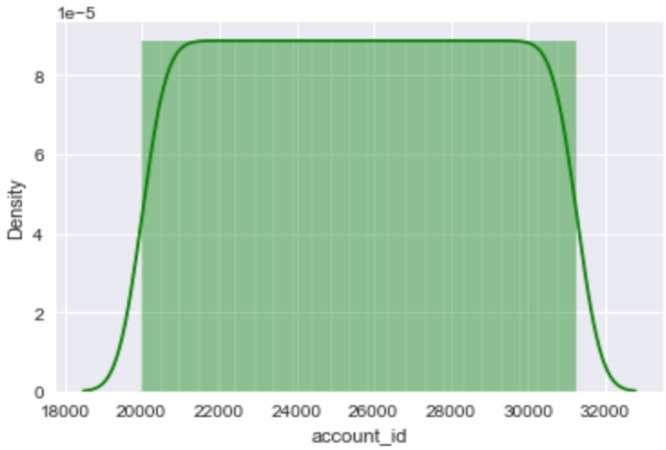
3) Exploratory Data Analysis

Univariate analysis:

account_id:

```
Description of account_id
--------------------------------------
count    11260.00000
mean     25629.50000
std       3250.62635
min      20000.00000
25%      22814.75000
50%      25629.50000
75%      28444.25000
max      31259.00000
Name: account_id, dtype: float64
--------------------------------------
```

```
Distribution of account_id
--------------------------------------
```




```
Boxplot of account_id
--------------------------------------
```

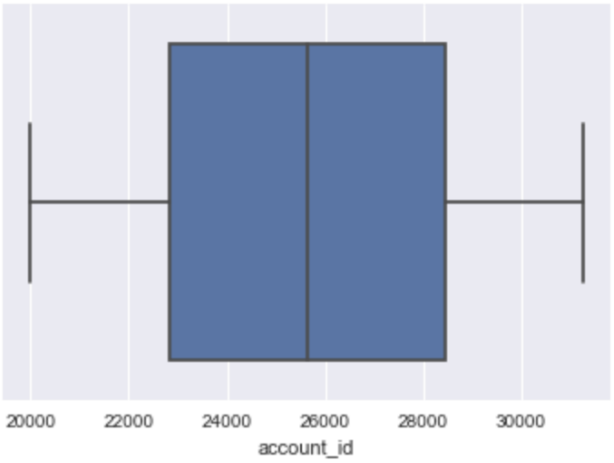
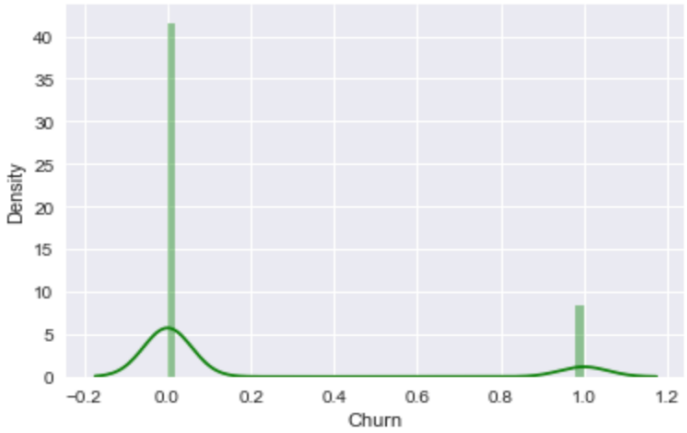


Fig. 6 Description, Distribution and boxplot of account_id

churn

```
Description of Churn
--------------------------------
count    11260.000000
mean         0.168384
std          0.374223
min          0.000000
25%          0.000000
50%          0.000000
75%          0.000000
max          1.000000
Name: Churn, dtype: float64
```

```
Distribution of Churn
-------------------------------------
```




```
Boxplot of Churn
-------------------------------------
```

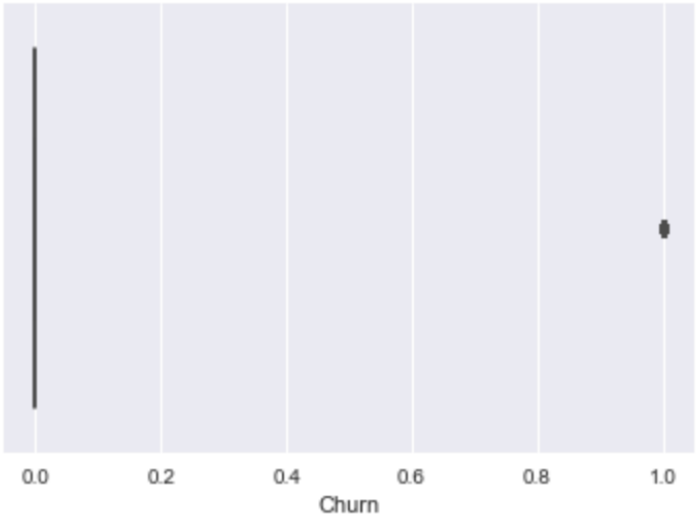
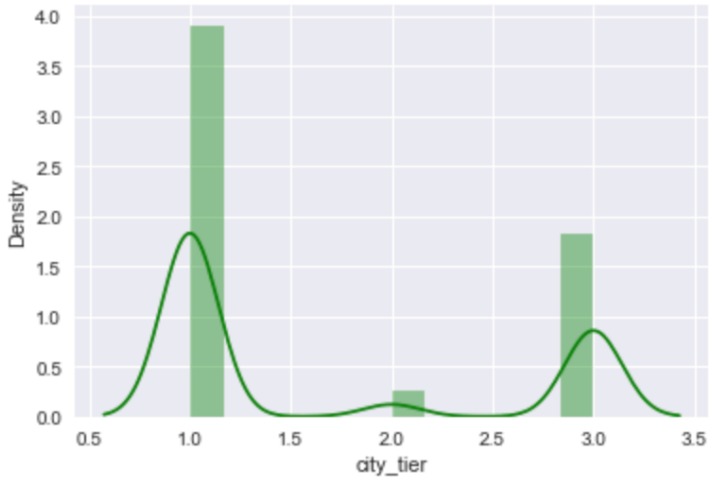


Fig. 7 Description, Distribution and boxplot of Churn

city_tier:

```
Description of city_tier
---------------------------------
count    11148.000000
mean         1.653929
std          0.915015
min          1.000000
25%          1.000000
50%          1.000000
75%          3.000000
max          3.000000
Name: city_tier, dtype: float64
```

Distribution of city_tier



Boxplot of city_tier

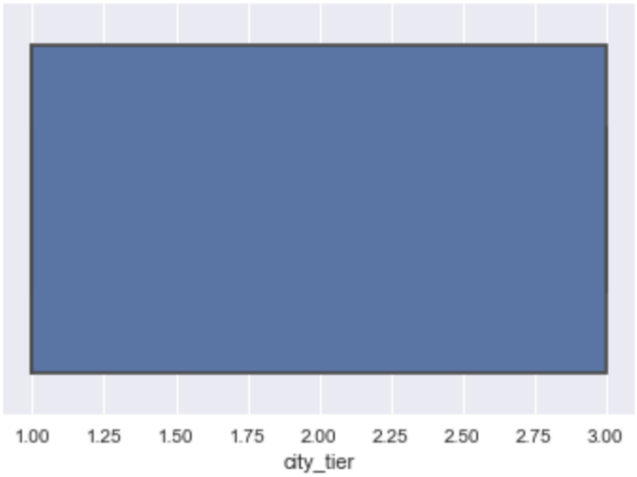
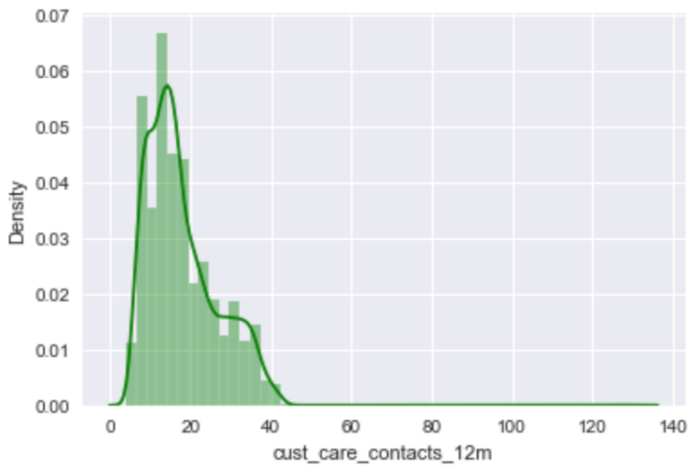


Fig. 8 Description, Distribution and boxplot of city_tier

cust_care_contacts_12m:

```
Description of cust_care_contacts_12m
--------------------------------------
count    11158.000000
mean        17.867091
std          8.853269
min          4.000000
25%         11.000000
50%         16.000000
75%         23.000000
max        132.000000
Name: cust_care_contacts_12m, dtype: float64
```

Distribution of cust_care_contacts_12m



Boxplot of cust_care_contacts_12m

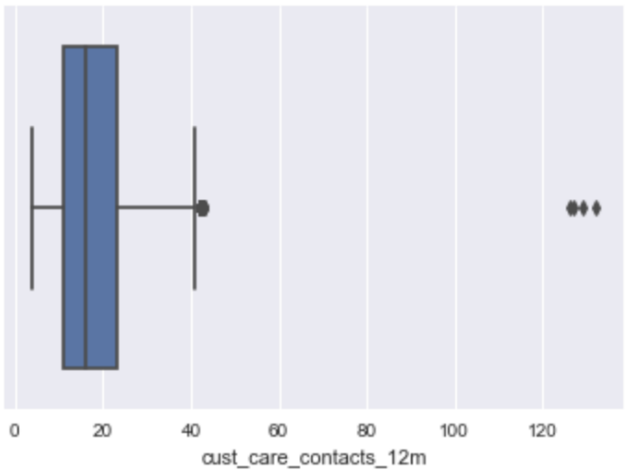
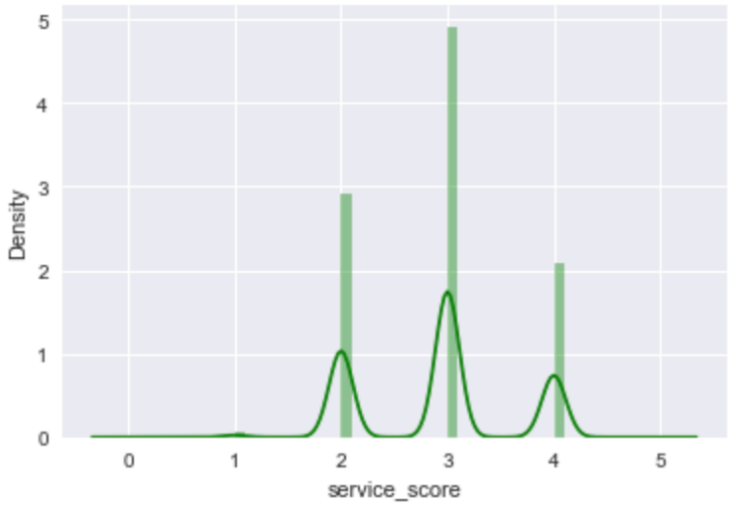


Fig. 9 Description, Distribution and boxplot of cust_care_contacts_12m

service_score:

```
Description of service_score
-------------------------------------
count    11162.000000
mean         2.902526
std          0.725584
min          0.000000
25%          2.000000
50%          3.000000
75%          3.000000
max          5.000000
Name: service_score, dtype: float64
```

```
Distribution of service_score
-------------------------------------
```




```
Boxplot of service_score
-------------------------------------
```

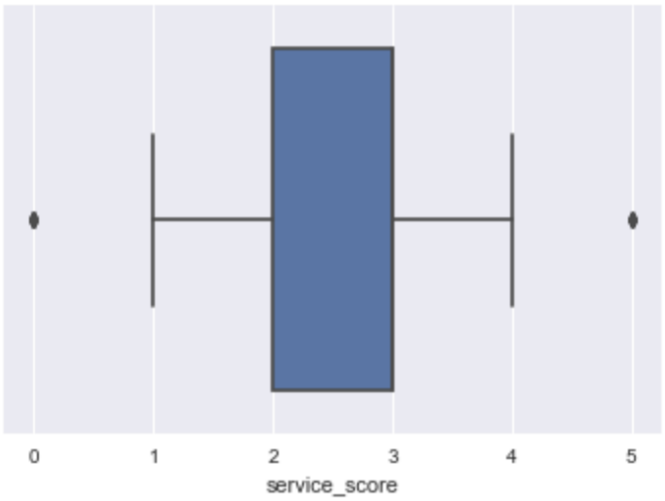
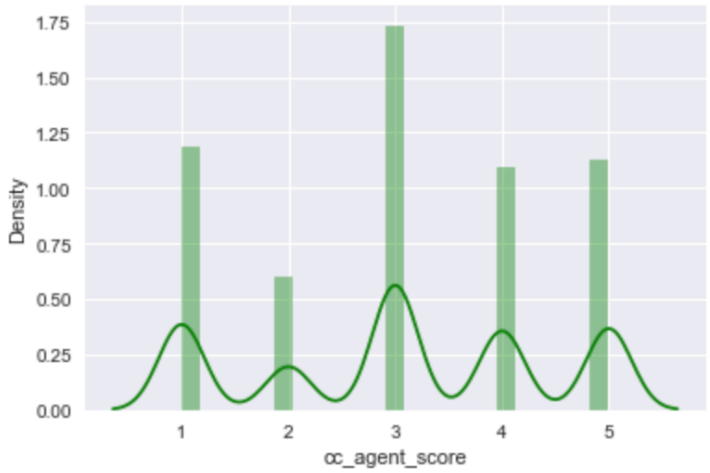


Fig. 10 Description, Distribution and boxplot of service_score

cc_agent_score:

```
Description of cc_agent_score
-------------------------------------
count    11144.000000
mean         3.066493
std          1.379772
min          1.000000
25%          2.000000
50%          3.000000
75%          4.000000
max          5.000000
Name: cc_agent_score, dtype: float64
```

```
Distribution of cc_agent_score
-------------------------------------
```




```
Boxplot of cc_agent_score
-------------------------------------
```

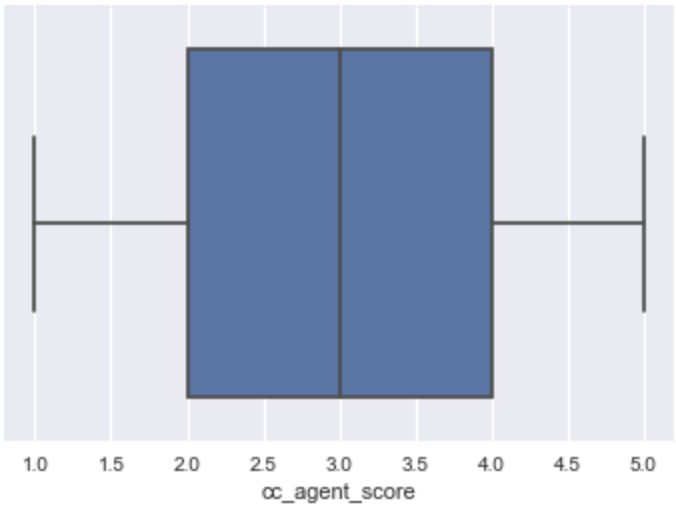
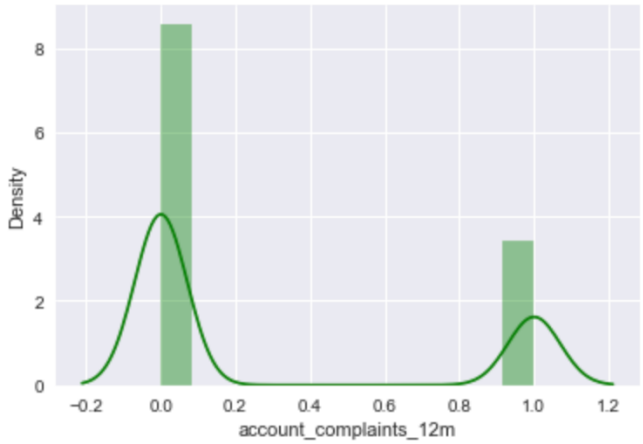


Fig. 11 Description, Distribution and boxplot of cc_agent_score

account_complaints_12m:

```
Description of account_complaints_12m
--------------------------------------
count    10903.000000
mean         0.285334
std          0.451594
min          0.000000
25%          0.000000
50%          0.000000
75%          1.000000
max          1.000000
Name: account_complaints_12m, dtype: float64
```

Distribution of account_complaints_12m



Boxplot of account_complaints_12m

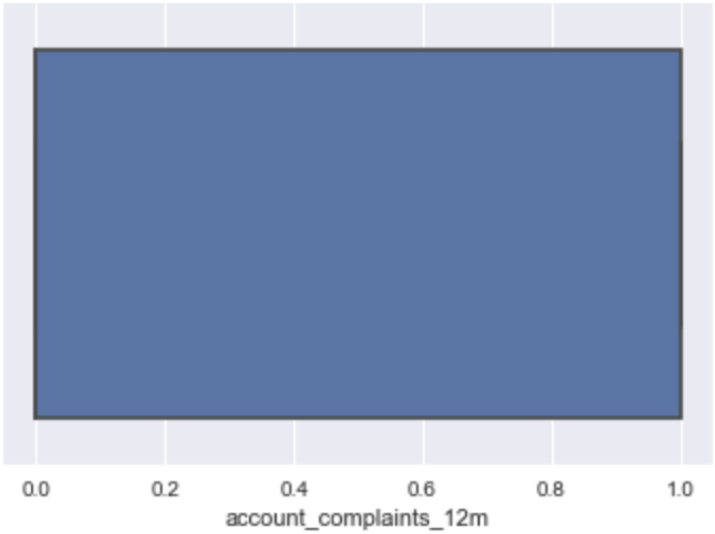


Fig. 12 Description, Distribution and boxplot of account_complaints_12m

Skewness of dataset:

	Skewness
Churn	1.77
city_tier	0.74
cust_care_contacts_12m	1.42
service_score	0.00
cc_agent_score	-0.14
account_complaints_12m	0.95

Fig. 13 Skewness of numerical variables

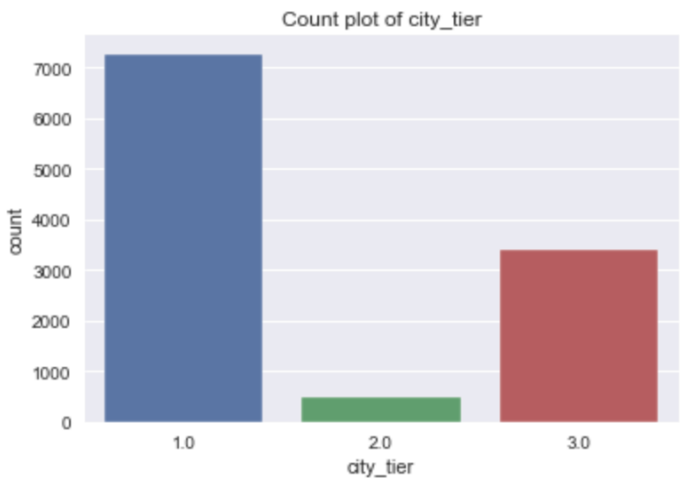
Conclusion:

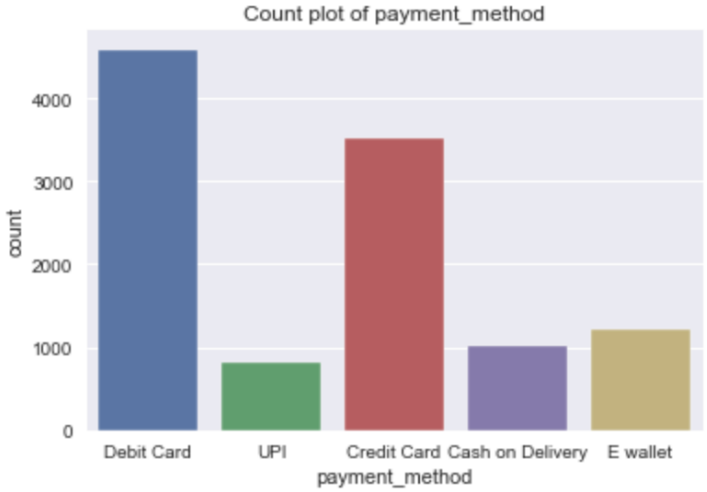
- Variable churn is highly skewed. For churn, values with 0 are in lead compared to those with 1.
- The mean and median of a variable account_id are the same, indicating that the data has a symmetrical distribution. The mean is also greater than the standard deviation, which shows that the data values are relatively close to the average. This suggests that the data is heavily concentrated around the mean, with relatively no outliers.
- Variable cust_care_contacts_12m is moderately skewed.The mean of the variable is 17, with a standard deviation of 8.85. The median of the data is 16, which is lower than the mean. This indicates that the data is skewed to the right, with a few high values that increase the mean. The standard deviation of 8.85 is relatively large compared to the mean, indicating that the data values are spread out, with some values far from the average.
- The mean of the "city_tier" variable is 1.653929, with a standard deviation of 0.915015. The median of the data is 1, which is lower than the mean. This suggests that the data is skewed to the right, with a few higher values that increase the mean. The standard deviation of 0.915015 indicates that the data values are relatively spread out, with some values deviating significantly from the average. Overall, the mean and median provide valuable information about the central tendency and variability of the "city_tier" variable.
- The "service_score" variable has a total of 11162 observations, with a mean value of 2.90 and a standard deviation of 0.73. The minimum value is 0 and the maximum value is 5. The median of the data is 3, which is also the 75th percentile value. This suggests that the majority of the data values are centered around 3, with relatively few values below or above this value. The standard deviation of 0.73 is relatively low compared to the mean, indicating that the data values are relatively close to the average. In conclusion, the mean, median, and standard deviation provide valuable information about the central tendency and variability of the "service_score" variable.
- The "cc_agent_score" variable has 11144 observations, with a mean value of 3.07 and a standard deviation of 1.38. The minimum value is 1 and the maximum value is 5. The median of the data is 3, while the 75th percentile value is 4. This indicates that the majority of the data values are

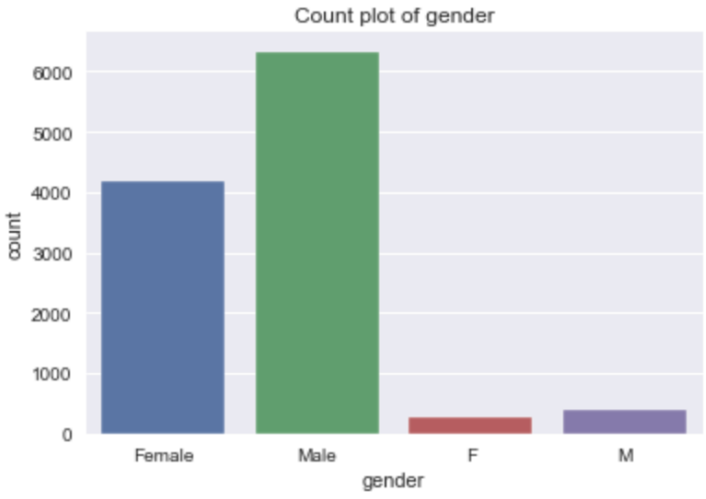
centered around 3, but with a larger spread than the median, with some values deviating significantly from the average. The standard deviation of 1.38 is relatively high compared to the mean, indicating that the data values are spread out, with some values far from the average. In conclusion, the mean, median, and standard deviation provide valuable information about the central tendency and variability of the "cc_agent_score" variable.

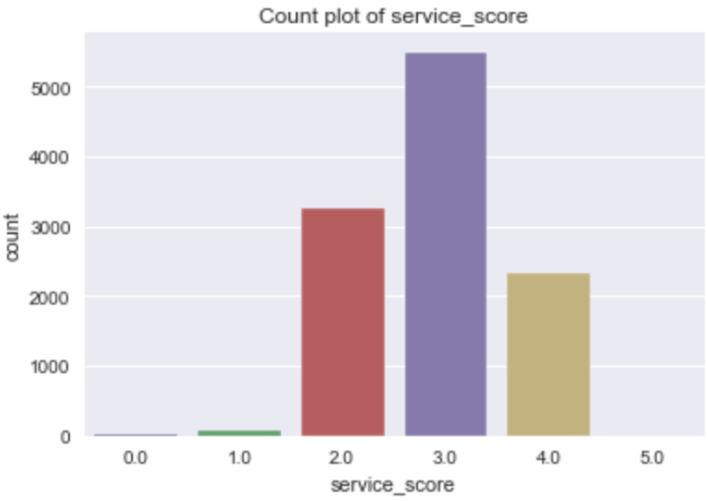
- The "account_complaints_12m" variable has 10903 observations, with a mean value of 0.29 and a standard deviation of 0.45. The minimum value is 0 and the maximum value is 1. The median of the data is 0, with the 25th and 50th percentile values also equal to 0. This indicates that the majority of the data values are centered around 0, with relatively few values above this value. The standard deviation of 0.45 is relatively large compared to the mean, indicating that the data values are spread out, with some values deviating significantly from the average. In conclusion, the mean, median, and standard deviation provide valuable information about the central tendency and variability of the "account_complaints_12m" variable. The mean gives a sense of the average value, while the standard deviation provides information about the spread of the data, and the median gives a measure of central tendency that is not affected by outliers or skewness in the data.

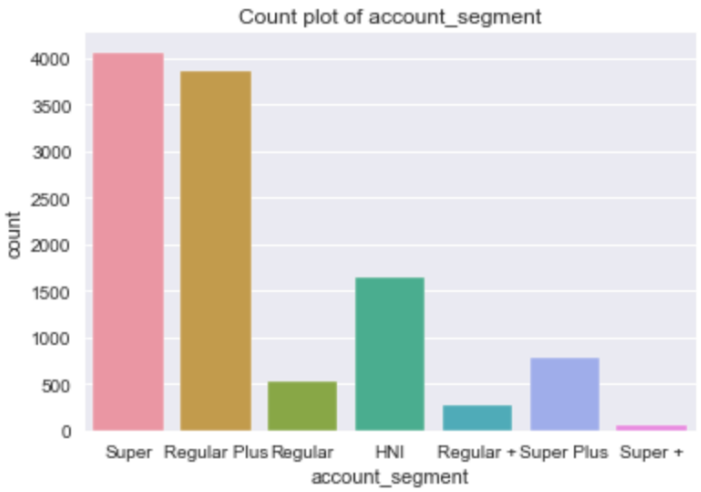
Univariate analysis for categorical variables:

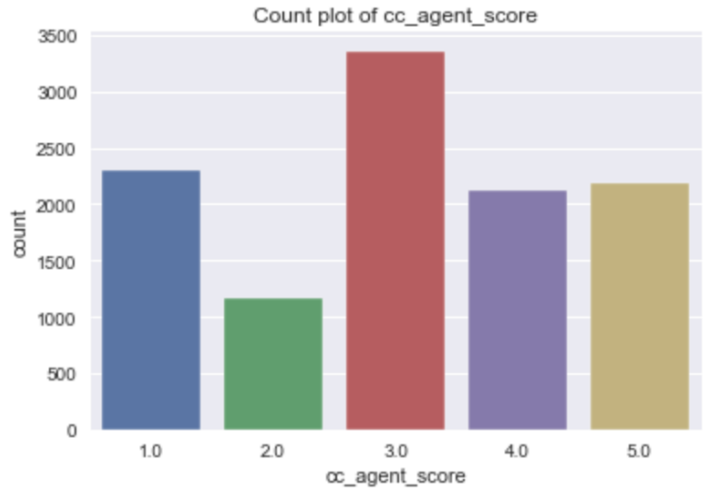


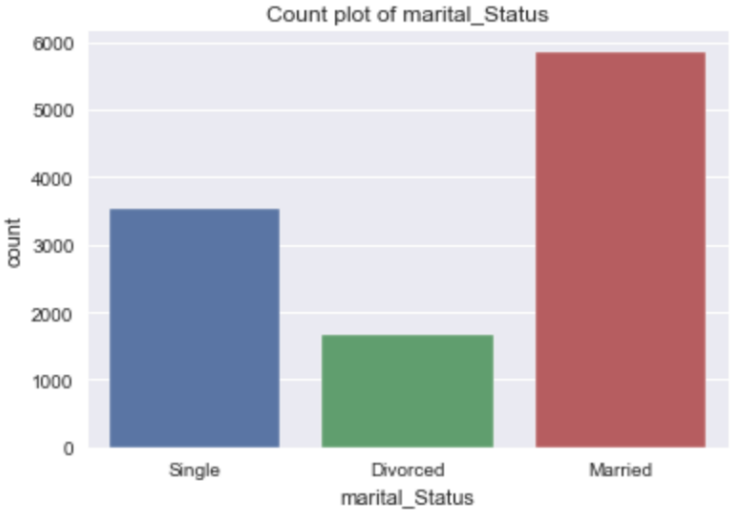












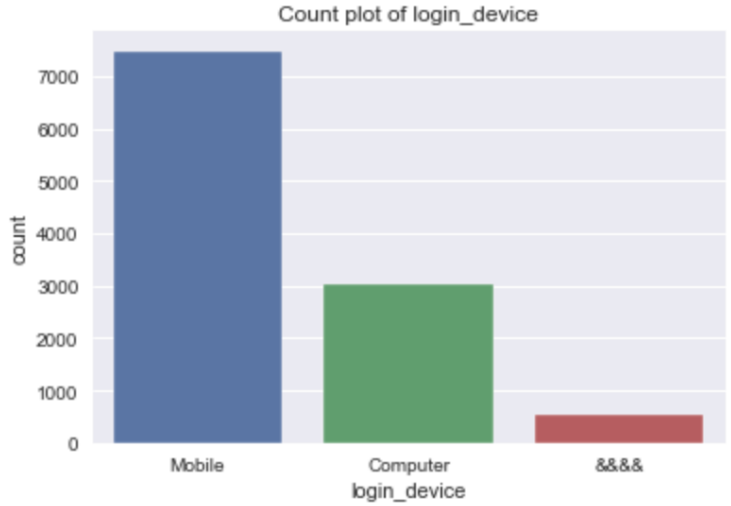


Fig. 14 Count plot of all categorical variables

Conclusion:

- City Tier Type 1 has the highest number of customers, reflecting the high population density in this city type.
- Debit and credit cards are the most preferred mode of payment among customers.
- The ratio of male customers is higher compared to female customers.
- The average service score given by customers is around 3, which suggests that there is room for improvement in the service provided.
- The majority of customers belong to the "Super+" segment, while the least number of customers belong to the "Regular" segment.
- Most of the customers availing services are married.
- Mobile devices are the preferred choice among customers for availing services.

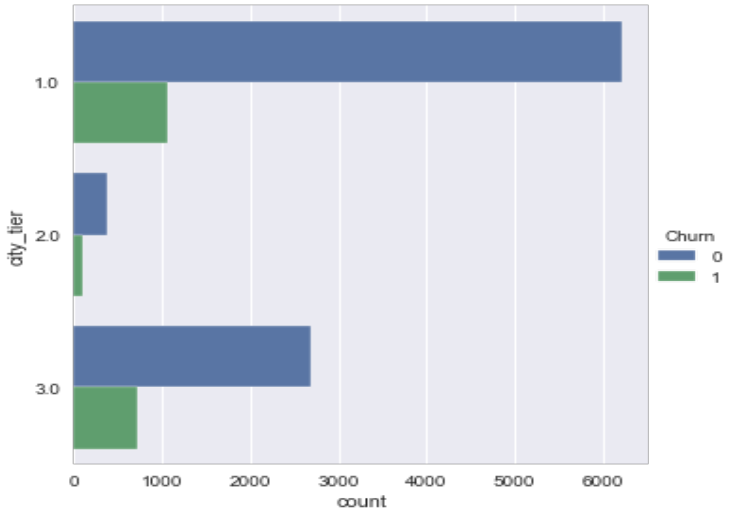
Bivariate analysis:

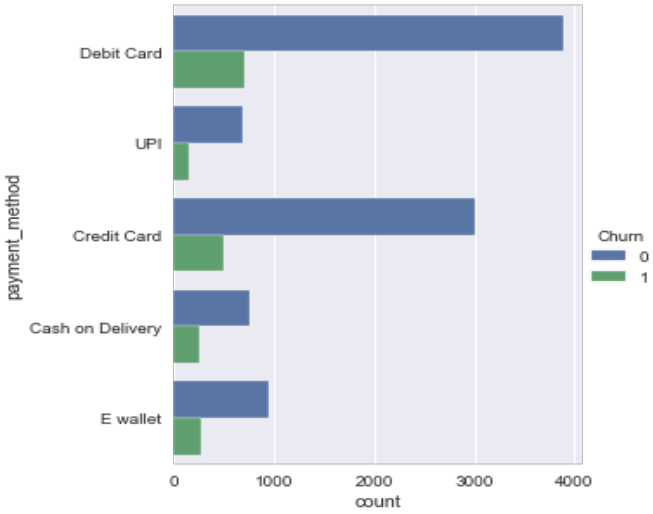
Pair plot:

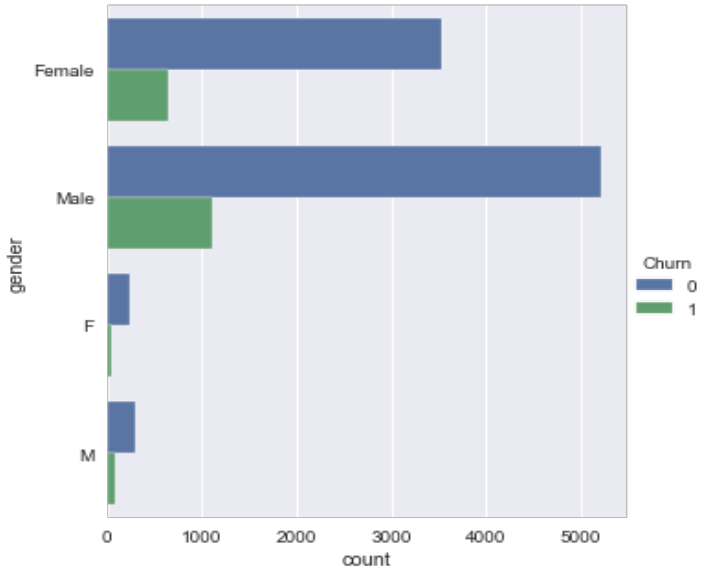
Fig. 15 Pair plot of all numerical variables

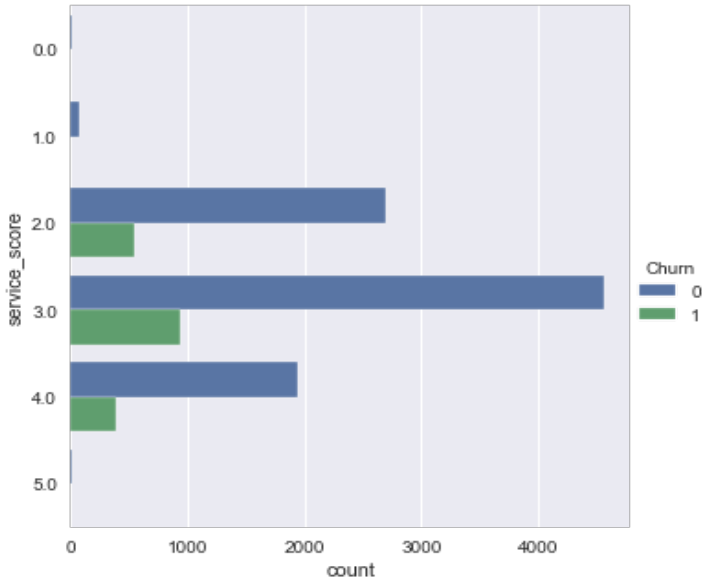
Conclusion:

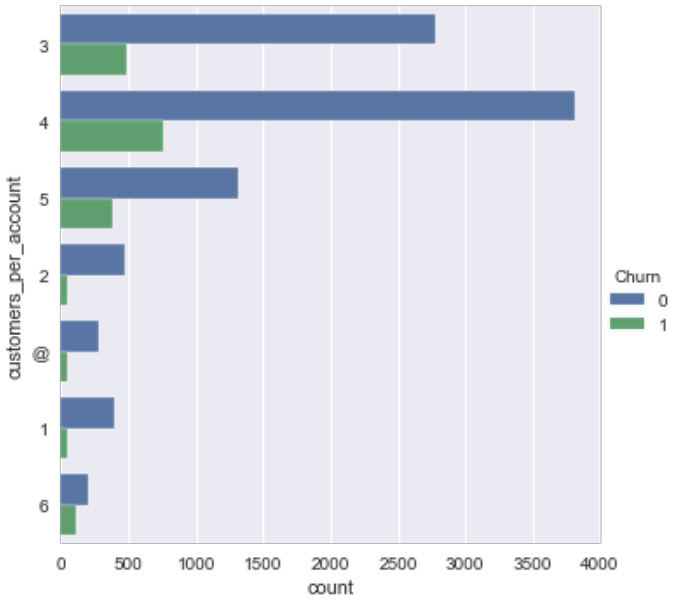
The pair-plot depicted above indicates that the independent variables are insufficient or poor predictors of the target variable since their densities overlap.

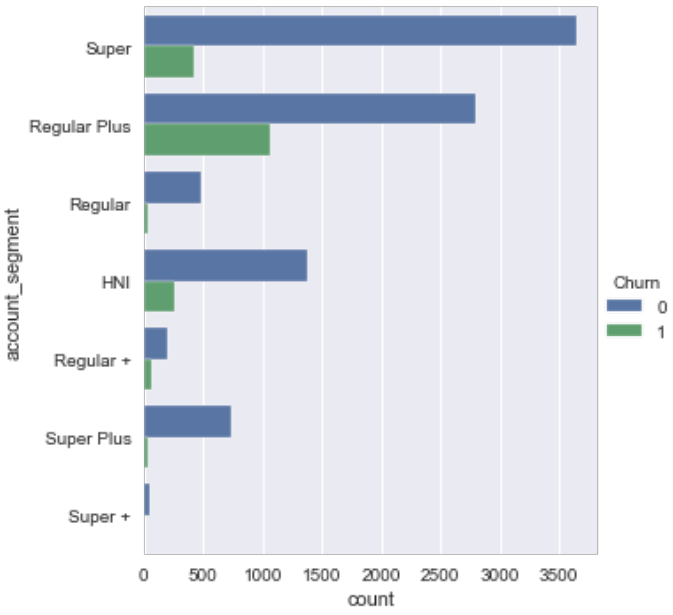

city_tier
1.0
2.0
3.0
0
1000
2000
3000
4000
5000
6000
count
Churn
0
1


payment_method
Debit Card
UPI
Credit Card
Cash on Delivery
E wallet
0
1000
2000
3000
4000
count
Churn
0
1


gender
Female
Male
F
M
0
1000
2000
3000
4000
5000
count
Churn
0
1


service_score
0.0
1.0
2.0
3.0
4.0
5.0
0
1000
2000
3000
4000
count
Churn
0
1


customers_per_account
3
4
5
2
@
1
6
0
500
1000
1500
2000
2500
3000
3500
4000
count
Churn
0
1


account_segment
Super
Regular Plus
Regular
HNI
Regular +
Super Plus
Super +
0
500
1000
1500
2000
2500
3000
3500
count
Churn
0
1

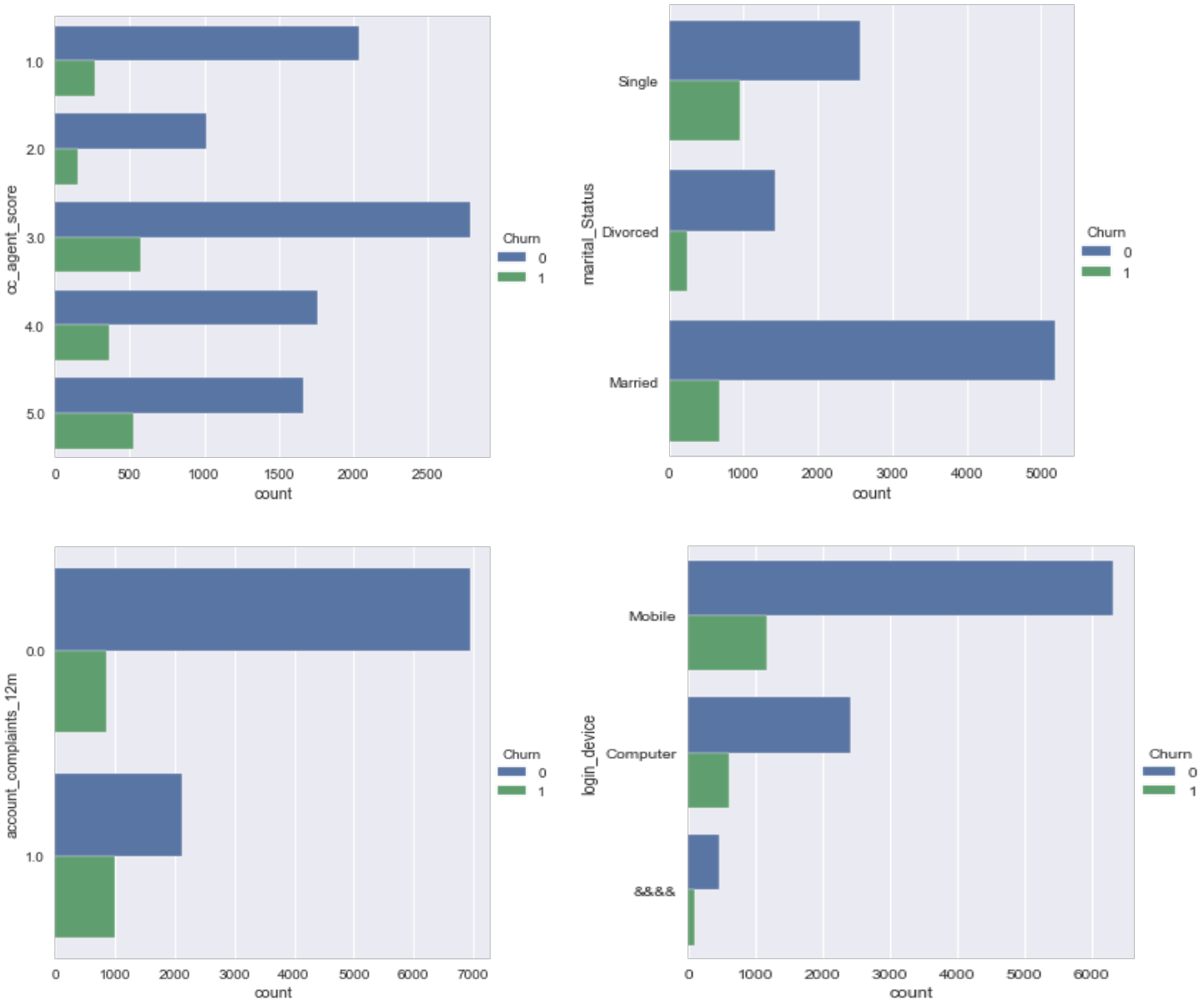


Fig. 16 Contribution of categorical variable toward churn

Conclusion:

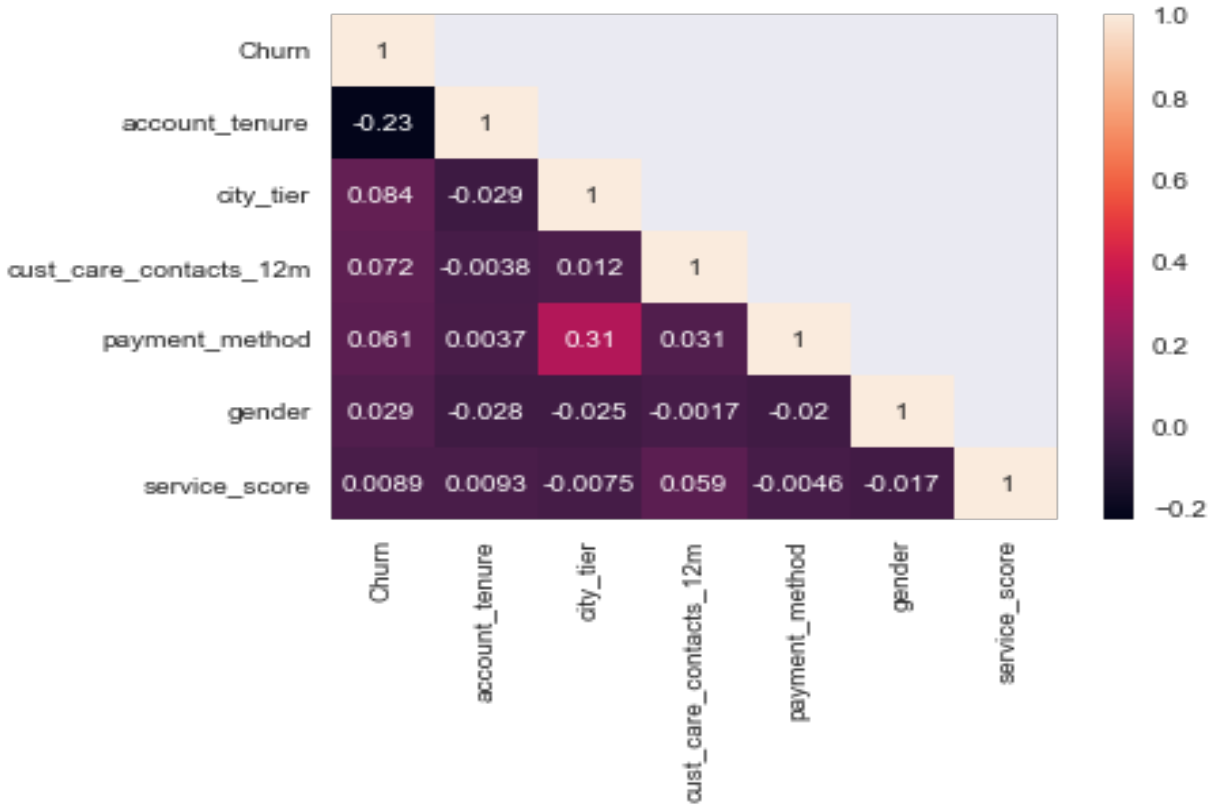
- City tier type 1 shows the highest rate of churn compared to city tier types 2 and 3.
- Customers who prefer debit cards or credit cards as their mode of payment are more likely to churn.
- Male customers have a higher churn rate compared to female customers.
- Customers in the "Regular Plus" segment are showing a higher churn rate.
- Single customers are more prone to churning compared to divorced or married customers.
- Customers who use the service over a mobile device show a higher rate of churn.
- It is important to note that these findings should be considered in the context of the overall

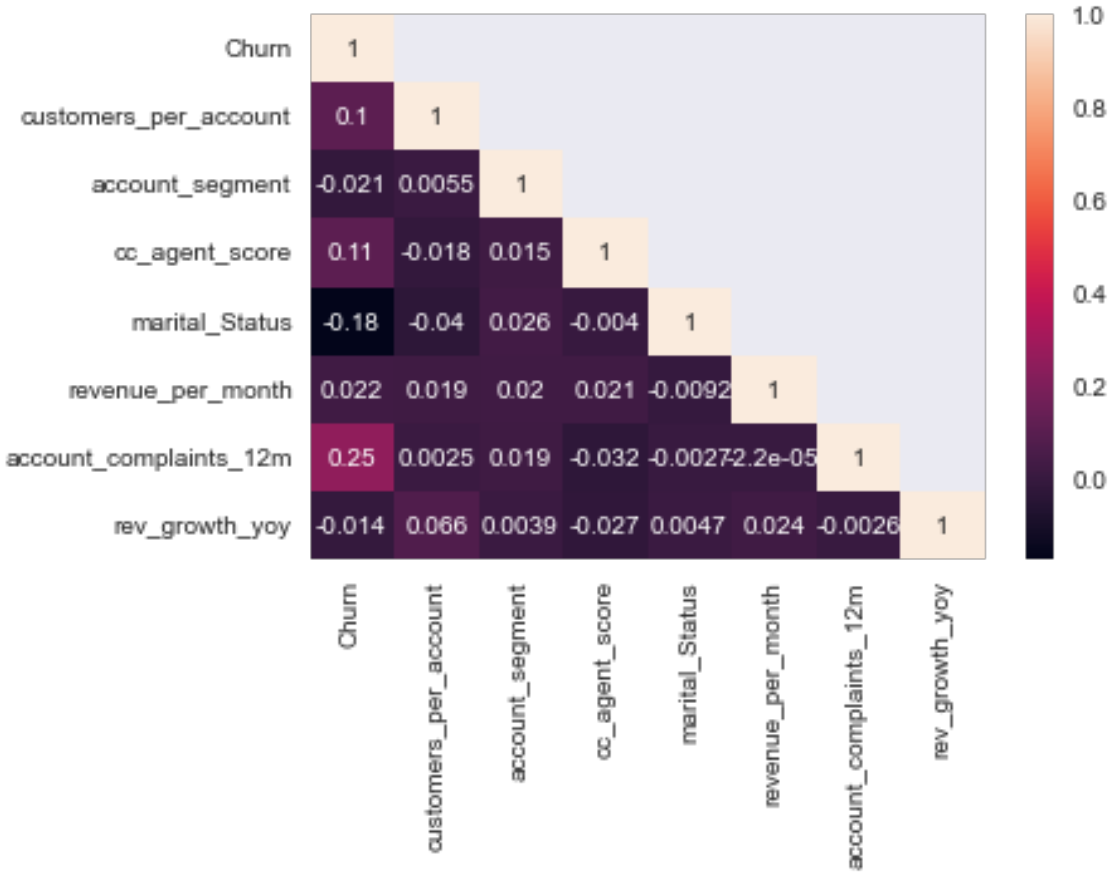
customer base and the other factors that may influence churn. Further analysis, such as multivariate analysis, may be necessary to establish the strength and direction of relationships between the variables and churn.

- Additionally, it would also be beneficial to understand the reasons behind the higher churn rates in certain segments or demographics to develop targeted retention strategies. This could involve surveying customers or looking at additional data sources such as call center logs or customer service reports.

Correlation among variable:

Following the treatment of flawed data and missing values, correlation analysis was done between the variables. We also transformed into integer data types to test for correlation because categorical data won't display in the following images.




Churn
customers_per_account
account_segment
cc_agent_score
marital_Status
revenue_per_month
account_complaints_12m
rev_growth_yoy
1
0.1 1
-0.021 0.0055 1
0.11 -0.018 0.015 1
-0.18 -0.04 0.026 -0.004 1
0.022 0.019 0.02 0.021 -0.0092 1
0.25 0.0025 0.019 -0.032 -0.0027 2.2e-05 1
-0.014 0.066 0.0039 -0.027 0.0047 0.024 -0.0026 1
1.0
0.8
0.6
0.4
0.2
0.0

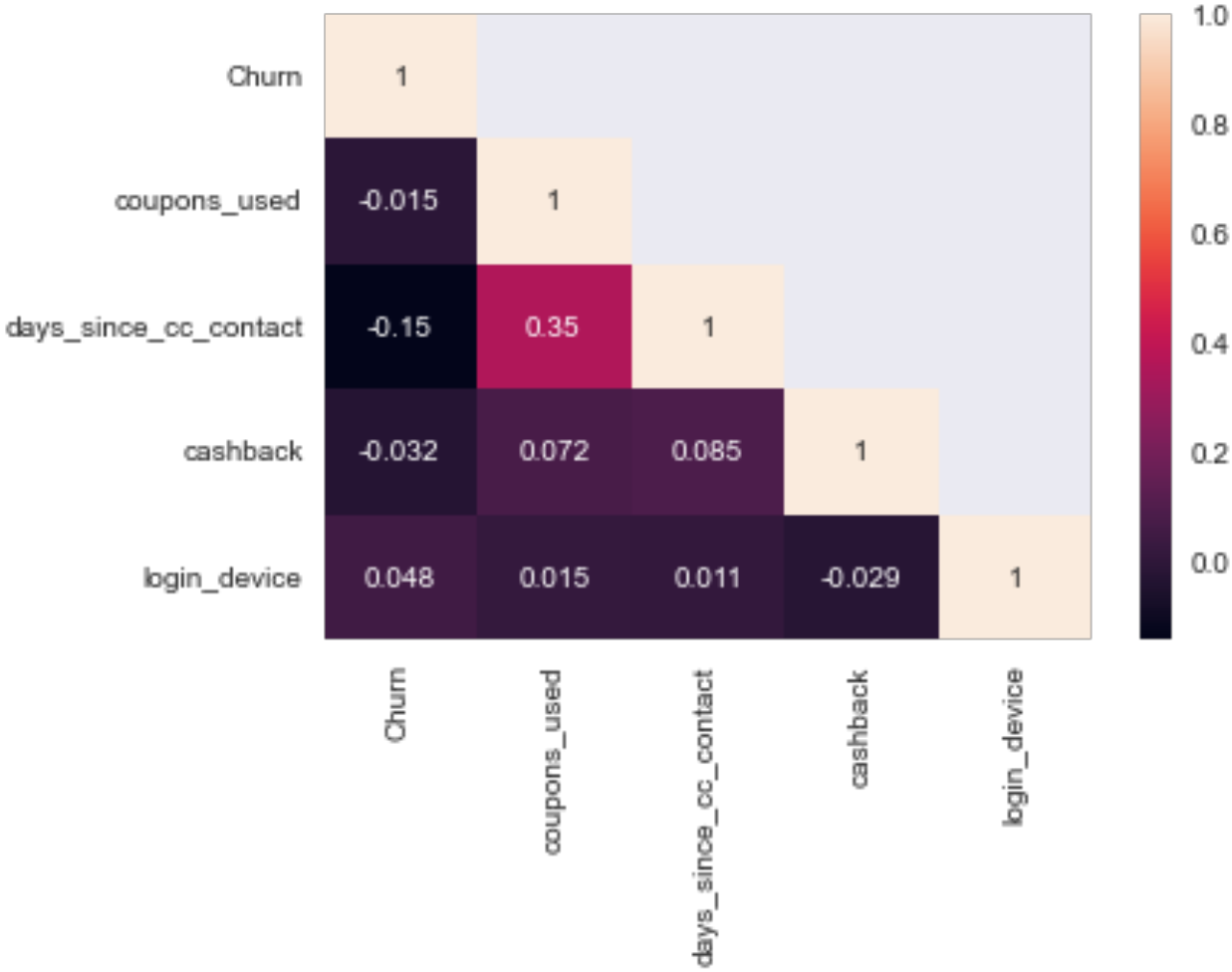

Churn
coupons_used
days_since_cc_contact
cashback
login_device
1
-0.015 1
-0.15 0.35 1
-0.032 0.072 0.085 1
0.048 0.015 0.011 -0.029 1
1.0
0.8
0.6
0.4
0.2
0.0

Fig. 17 Correlation among variables

Inferences from correlation:

- Variable “account_tenure” shows high co-relation with Churn.
- Variable “marital_Status” shows high co-relation with churn.
- Variable “account_complaints_2m” shows high- correlation with churn.

Removal of unwanted variables

After carefully analysing the data, we come to the conclusion that at this point in the project, removing variables is not necessary. We can get rid of the variable "AccountID," which stands for a special ID given to special clients. But doing so will result in 8 duplicate rows. Looking at the univariate and bi-variate analyses, the rest of the factors appear to be significant.

Outlier treatment

This dataset combines categorical and continuous variables. Because each category represents a certain customer type, applying outlier treatment to categorical variables is completely illogical. Therefore, we only apply outlier treatment to variables that are continuous in nature.

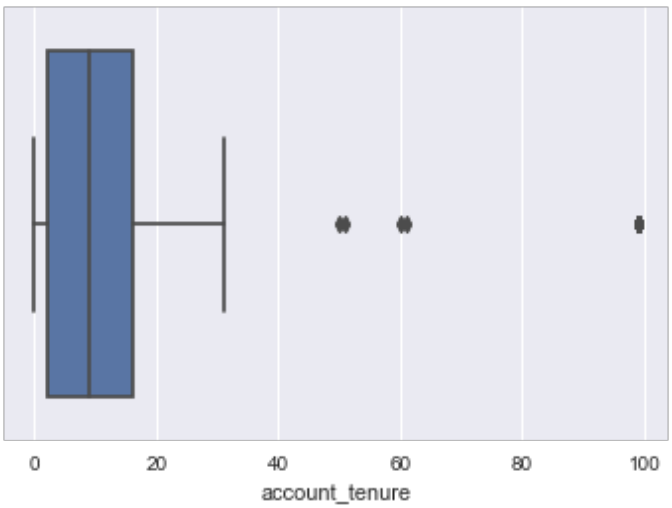
- Box plot was used to identify whether an outlier was present in a variable.
- The outlier in the variable is represented by the dots outside a quantile's upper bound.
- Eight continuous variables, including "account_tenure," " cust_care_contacts_12m," " customers_per_account," "cashback," " revenue_per_month," " days_since_cc_contact," "coupons_used" and "rev_growth_yoy," are included in the dataset.
- To eliminate outliers, we employed upper limit and lower limit. The visual depiction of variables both before and after outlier correction is shown below

Proportion of outliers present before outlier treatment are shown below:

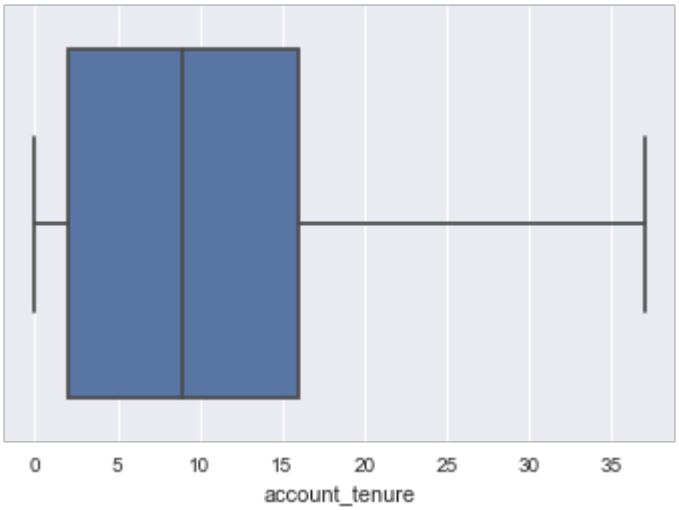
	outlier %
Churn	16.83
account_tenure	1.26
city_tier	0.00
cust_care_contacts_12m	0.38
payment_method	0.00
gender	0.00
service_score	0.12
customers_per_account	6.73
account_segment	14.68
cc_agent_score	0.00
marital_Status	0.00
revenue_per_month	1.68
account_complaints_12m	0.00
rev_growth_yoy	0.00
coupons_used	12.42
days_since_cc_contact	1.16
cashback	8.59
login_device	0.00

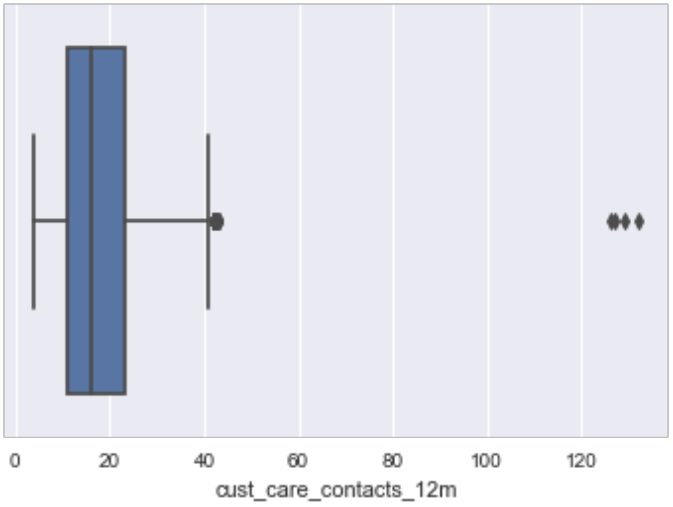
Fig. 18 Proportion of outliers present in the data set

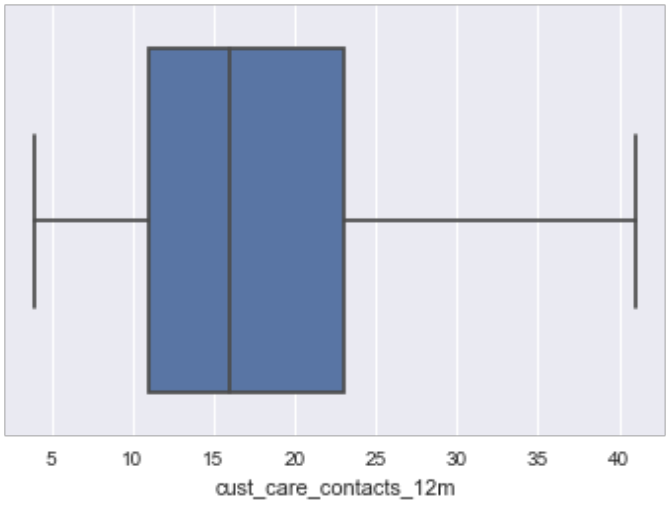
BEFORE

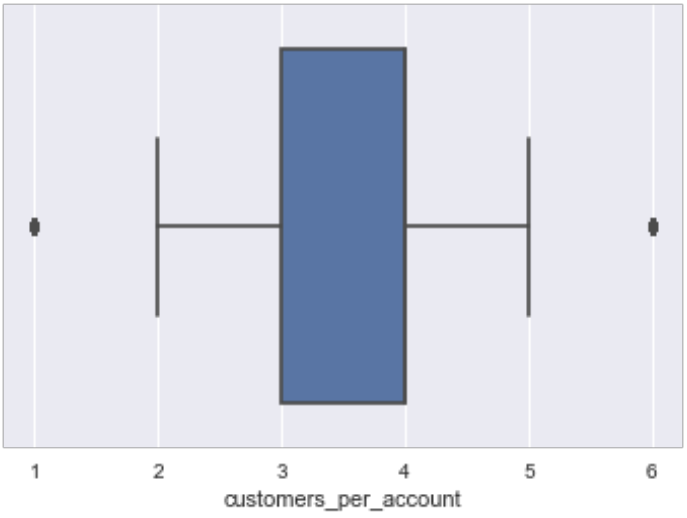


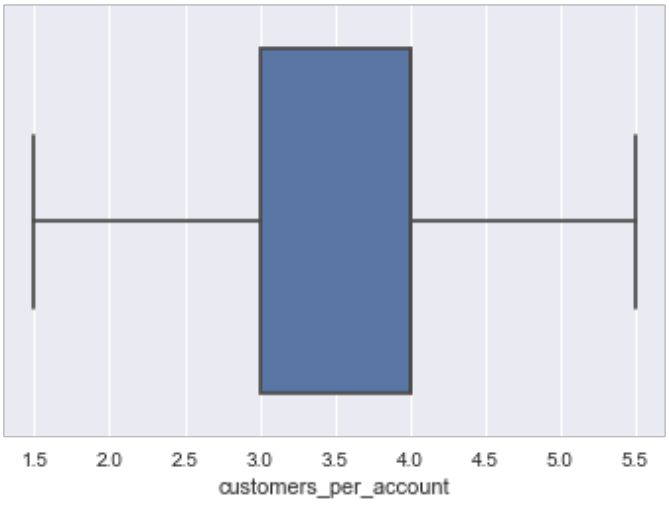
AFTER

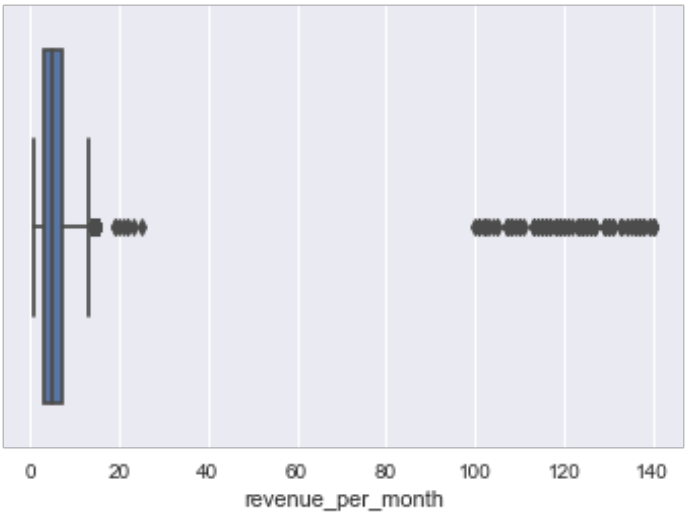


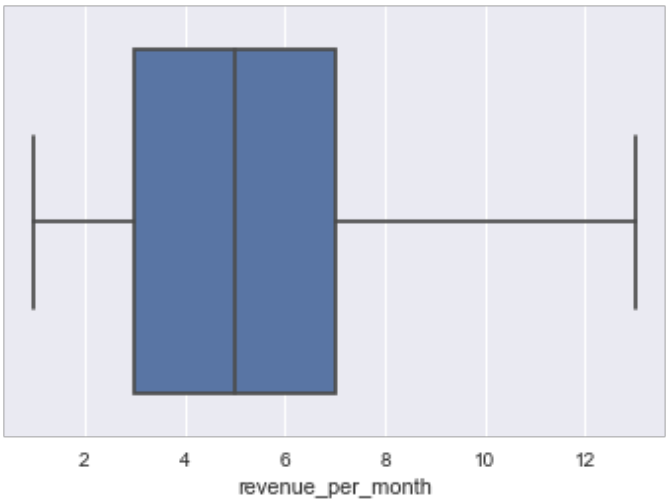

0
20
40
60
80
100
120
cust_care_contacts_12m

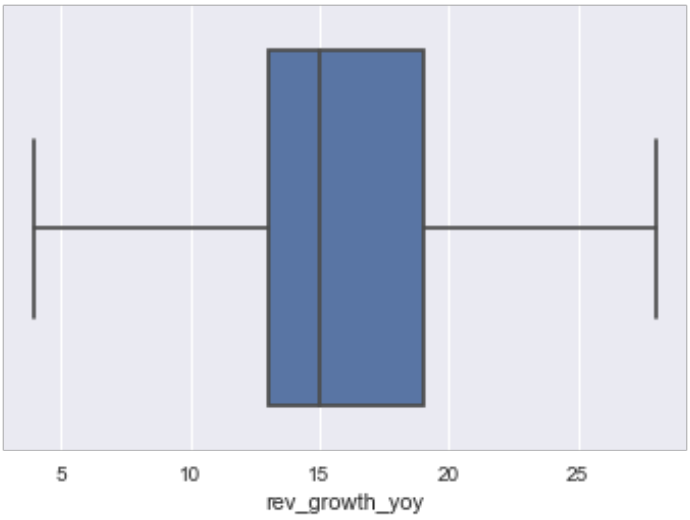

5
10
15
20
25
30
35
40
cust_care_contacts_12m

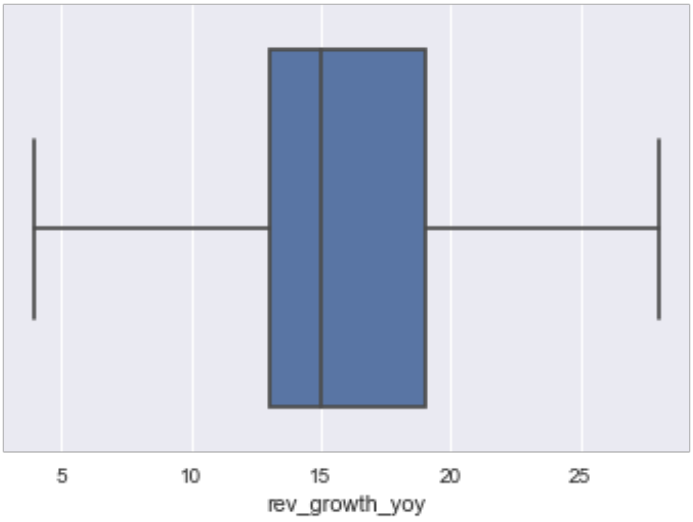

1
2
3
4
5
6
customers_per_account

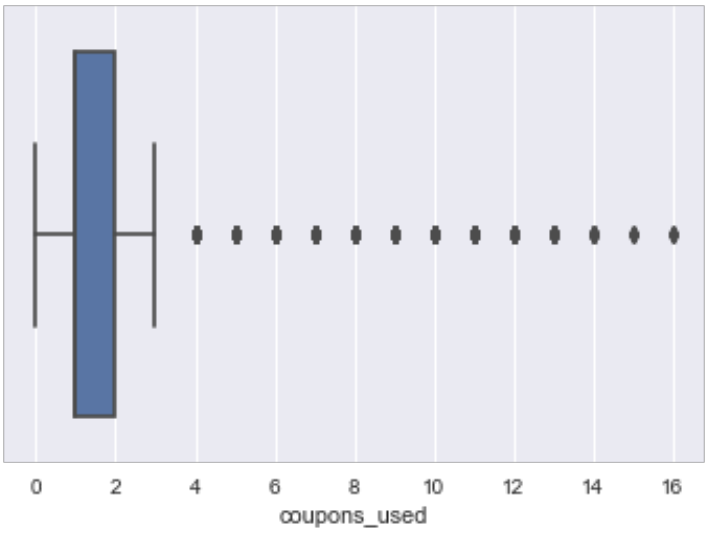

1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
customers_per_account

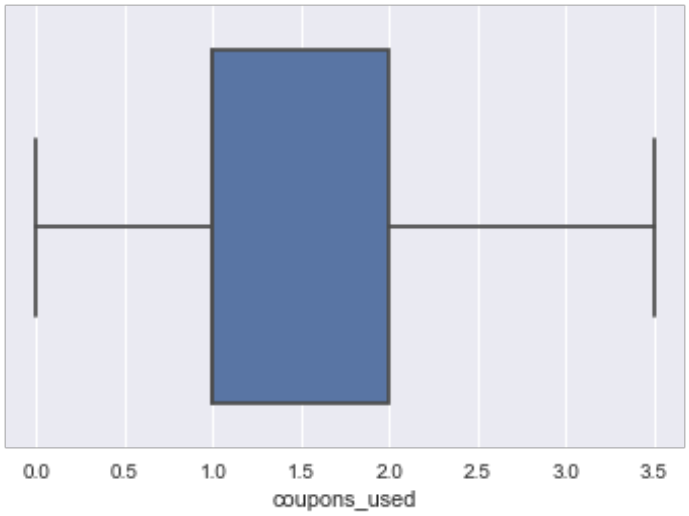

0
20
40
60
80
100
120
140
revenue_per_month

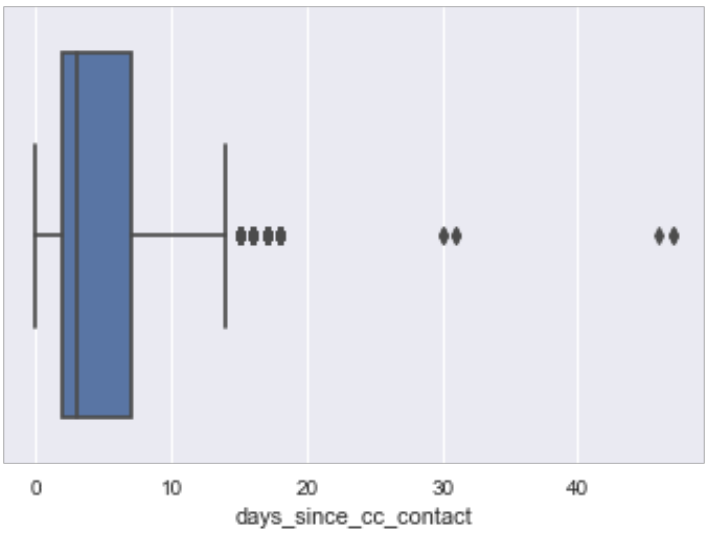

2
4
6
8
10
12
revenue_per_month

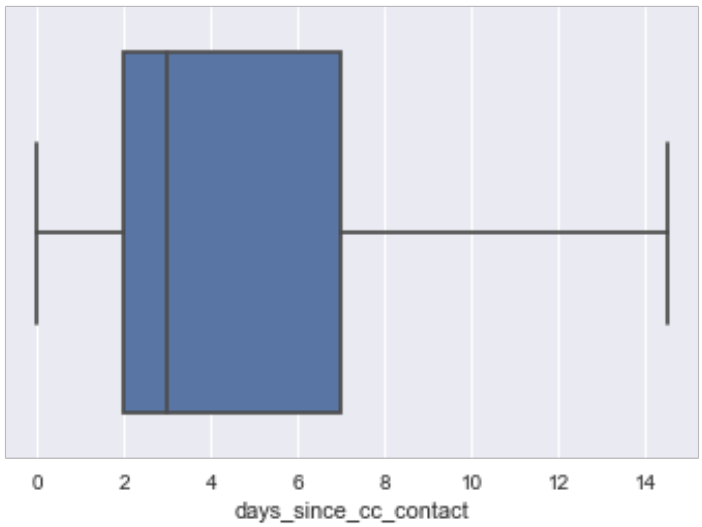

5
10
15
20
25
rev_growth_yoy


5
10
15
20
25
rev_growth_yoy


0
2
4
6
8
10
12
14
16
coupons_used


0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
coupons_used


0
10
20
30
40
days_since_cc_contact


0
2
4
6
8
10
12
14
days_since_cc_contact

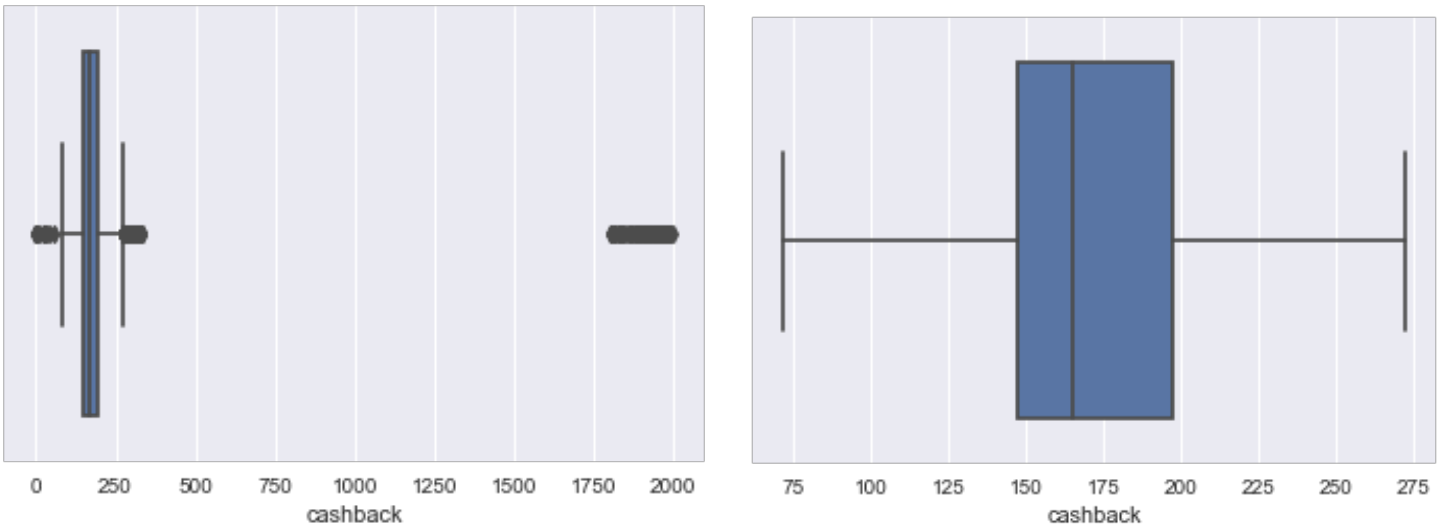


Fig. 19 Before and after outlier treatment

Removing duplicates

There are 259 rows with duplicate values. Hence, we can drop them

```
Number of duplicate rows: 259
```

After dropping the duplicates,

```
Number of duplicate rows = 0
```

And the total rows of the dataset got decreased to 11001 from 11260

Missing value treatment

- We have data anomalies in 17 of the 19 variables, and null values in 15 of the variables.
- Using the median instead of the mean when the variable is continuous because the median is less likely to contain outliers than the mean.
- In situations where variables are categorical in nature, using "Mode: to impute null values.
- We have handled null values separately for each variable because each one is different in its own way.

Treating variable “Tenure”

We can see that "#" and "nan" are present in the data by looking at the variable's unique observations. Where "nan" stands for a null value and "#" indicates an anomaly.

```
array([4, 0, 2, 13, 11, '#', 9, 99, 19, 20, 14, 8, 26, 18, 5, 30, 7, 1,
       23, 3, 29, 6, 28, 24, 25, 16, 10, 15, 22, nan, 27, 12, 21, 17, 50,
       60, 31, 51, 61], dtype=object)
```

By replacing "#" with "nan" and then "nan" with the calculated variable's median, we no longer detect any incorrect data or null values.

Because the IDE identified it as an object data type due to the presence of incorrect data, the data type was converted to integer.

```
[4, 0, 2, 13, 11, 9, 99, 19, 20, 14, 8, 26, 18, 5, 30, 7, 1, 23, 3, 29, 6, 28, 24, 25, 16
, 10, 15, 22, 27, 12, 21, 17, 50, 60, 31, 51, 61]
Length: 37, dtype: Int64
```

Treating variable "City_Tier"

When we examine the variable's singular observations, we can observe that the null value is there, as seen below.

```
array([ 3.,  1., nan,  2.])
```

Since "nan" has been replaced with the variable's computed mode, there are no longer any null values present.

Data type was changed from object to integer because the existence of incorrect data caused the IDE to identify it as such.

```
array([3., 1., 2.])
```

Treating variable "cust_care_contacts_12m"

When we examine the variable's singular observations, we can observe that the null value is there, as seen below.

```
array([  6.,   8.,  30.,  15.,  12.,  22.,  11.,   9.,  31.,  18.,  13.,
        20.,  29.,  28.,  26.,  14.,  10.,  25.,  27.,  17.,  23.,  33.,
        19.,  35.,  24.,  16.,  32.,  21.,  nan,  34.,   5.,   4., 126.,
         7.,  36., 127.,  42.,  38.,  37.,  39.,  40.,  41., 132.,  43.,
       129.])
```

Now that "nan" has been replaced with the variable's computed Median, there are no longer any null values present.

Data type was changed from object to integer because the existence of incorrect data caused the IDE to identify it as such.

```
array([  6.,   8.,  30.,  15.,  12.,  22.,  11.,   9.,  31.,  18.,  13.,
        20.,  29.,  28.,  26.,  14.,  10.,  25.,  27.,  17.,  23.,  33.,
        19.,  35.,  24.,  16.,  32.,  21.,  34.,   5.,   4., 126.,   7.,
        36., 127.,  42.,  38.,  37.,  39.,  40.,  41., 132.,  43., 129.])
```

Treating variable "payment_method"

When we examine the variable's singular observations, we can observe that the null value is there, as seen below.

```
array(['Debit Card', 'UPI', 'Credit Card', 'Cash on Delivery', 'E wallet',
       nan], dtype=object)
```

Now that "nan" has been replaced with the variable's computed Mode, there are no longer any null values present.

Additionally, label encoding for the observations was done. where (1, 2, 3, 4, and 5) are debit cards, UPI, credit cards, cash on delivery, and electronic wallets. After that, they should be converted to integer data types because further model development will require them.

```
array(['1', '2', '3', '4', '5'], dtype=object)
```

Treating variable "gender"

When we examine the variable's distinct observations, we find that there are numerous abbreviations for the same observation as shown below, along with the presence of a null value.

```
array(['Female', 'Male', 'F', nan, 'M'], dtype=object)
```

Now that "nan" has been replaced with the variable's computed Mode, there are no longer any null values present.

Additionally, label encoding for the observations was done. where card 1 is a female and card 2 is a male. After that, they should be converted to integer data types because further model development will require them.

```
array(['1', '2'], dtype=object)
```

Treating variable "service_score"

When we examine the variable's singular observations, we can observe that the null value is there, as seen below.

```
array([ 3.,  2.,  1., nan,  0.,  4.,  5.])
```

Now that "nan" has been replaced with the variable's computed Mode, there are no longer any null values present.

After that, they were converted to integer data types because later model development would require them.

```
array([3., 2., 1., 0., 4., 5.])
```

Treating variable "customers_per_account"

When we look at the variable's singular observations, we can see that it contains null values and the character "@," which is invalid data.

```
array([3, 4, nan, 5, 2, '@', 1, 6], dtype=object)
```

Once we have eliminated the presence of incorrect data and null values by replacing "@" with "nan" and "nan" with the variable's estimated median, we can convert the data to integer data type, which will be utilised for further model construction.

```
array([3., 4., 5., 2., 1., 6.])
```

Treating variable “account_segment”

We look at the unique observations in the variable and see presence of null value as well different denotations for the same type of observations, shown below.

```
array(['Super', 'Regular Plus', 'Regular', 'HNI', 'Regular +', nan,
       'Super Plus', 'Super +'], dtype=object)
```

Replacing “nan” with calculated Mode of the variable and also labelled different account segments, where in 1 = Super, 2 = Regular Plus, 3 = Regular, 4 = HNI and 5 = Super Plus and now we don’t see any presence of bad data and null values.

Then converting them to integer data type as it will be used for further model building.

```
array(['1', '2', '3', '4', '5'], dtype=object)
```

Treating variable “cc_agent_score”

When we examine the variable's singular observations, we can observe that the null value is there, as seen below.

```
array([ 2.,  3.,  5.,  4., nan,  1.])
```

We no longer observe any invalid data or null values after replacing "nan" with the estimated Mode of the variable and after converting them to integer data type, which will be used for subsequent model construction.

```
array([2., 3., 5., 4., 1.])
```

Treating variable “marital_Status”

When we examine the variable's singular observations, we can observe that the null value is there, as seen below.

```
array(['Single', 'Divorced', 'Married', nan], dtype=object)
```

Using the estimated mode of the variable in place of "nan" and labelling the observations. We no longer notice any instances of invalid data or null values where 1 = Single, 2 = Divorced, and 3 = Married.

After that, they were converted to integer data types because later model development would require

them.

```
array(['Single', 'Divorced', 'Married'], dtype=object)
```

Treating variable "revenue_per_month"

When we examine the variable's singular observations, we can see that it contains null values as well as the indicator "+" for flawed data. seen beneath.

```
array([9, 7, 6, 8, 3, 2, 4, 10, 1, 5, '+', 130, nan, 19, 139, 102, 120,
       138, 127, 123, 124, 116, 21, 126, 134, 113, 114, 108, 140, 133,
       129, 107, 118, 11, 105, 20, 119, 121, 137, 110, 22, 101, 136, 125,
       14, 13, 12, 115, 23, 122, 117, 131, 104, 15, 25, 135, 111, 109,
       100, 103], dtype=object)
```

By replacing "+" with "nan" and then "nan" with the variable's estimated median, we no longer detect any incorrect data or null values.

After that, they were converted to integer data types because later model development would require them.

```
array([  9.,   7.,   6.,   8.,   3.,   2.,   4.,  10.,   1.,   5., 130.,
        19., 139., 102., 120., 138., 127., 123., 124., 116.,  21., 126.,
       134., 113., 114., 108., 140., 133., 129., 107., 118.,  11., 105.,
        20., 119., 121., 137., 110.,  22., 101., 136., 125.,  14.,  13.,
        12., 115.,  23., 122., 117., 131., 104.,  15.,  25., 135., 111.,
       109., 100., 103.])
```

Treating variable "account_complaints_12m"

When we examine the variable's singular observations, we can observe that the null value is there, as seen below.

```
array([ 1.,  0., nan])
```

We no longer notice any null values after replacing "nan" with the calculated Mode of the variable and converting the results to integer data type, which will be used for further model construction.

```
array([1., 0.])
```

Treating variable "rev_growth_yoy"

When we examine the variable's singular observations, we can see that "$," which stands for flawed data, is present. seen beneath.

```
array([11, 15, 14, 23, 22, 16, 12, 13, 17, 18, 24, 19, 20, 21, 25, 26,
       '$', 4, 27, 28], dtype=object)
```

By replacing "$" with "nan" and then "nan" with the variable's estimated median, we no longer detect any incorrect data or null values.

After that, they were converted to integer data types because later model development would require them.

```
array([11., 15., 14., 23., 22., 16., 12., 13., 17., 18., 24., 19., 20.,
       21., 25., 26.,  4., 27., 28.])
```

Treating variable “coupons_used”

When we examine the unique observations for the variable, we can see that there are invalid data symbols like "$", "*," and "#" present. seen beneath.

```
array([1, 0, 4, 2, 9, 6, 11, 7, 12, 10, 5, 3, 13, 15, 8, '#', '$', 14,
       '*', 16], dtype=object)
```

Then we convert them to integer data type since it would be utilised for further model development, replacing "$", "*" and "#" with "nan" and further replacing "nan" with calculated median of the variable. Now we don't see any presence of faulty data and null values.

```
array([ 1.,  0.,  4.,  2.,  9.,  6., 11.,  7., 12., 10.,  5.,  3., 13.,
       15.,  8., 14., 16.])
```

Treating variable “days_since_cc_contact”

When we examine the variable's singular observations, we can see that "$," which stood for "poor data," and null values are both present. seen beneath.

```
array([5, 0, 3, 7, 2, 1, 8, 6, 4, 15, nan, 11, 10, 9, 13, 12, 17, 16, 14,
       30, '$', 46, 18, 31, 47], dtype=object)
```

By replacing "$" with "nan" and then "nan" with the variable's estimated median, we no longer detect any incorrect data or null values.

After that, they were converted to integer data types because later model development would require them.

```
array([ 5.,  0.,  3.,  7.,  2.,  1.,  8.,  6.,  4., 15., 11., 10.,  9.,
       13., 12., 17., 16., 14., 30., 46., 18., 31., 47.])
```

Treating variable “cashback”

When we examine the variable's singular observations, we can see that "$," which stood for "poor data," and null values are both present. seen beneath.

```
array([159.93, 120.9, nan, ..., 227.36, 226.91, 191.42], dtype=object)
```

By replacing "$" with "nan" and then "nan" with the variable's estimated median, we no longer detect any incorrect data or null values.

After that, they were converted to integer data types because later model development would require them.

```
array([159.93, 120.9 , 165.25, ..., 227.36, 226.91, 191.42])
```

Treating variable "login_device"

When we examine the individual observations for the variable, we can see that "&&&&"—a symbol for incorrect data—as well as null values are present. seen beneath.

```
array(['Mobile', 'Computer', '&&&&', nan], dtype=object)
```

Substituting "nan" for "&&&&" and then "nan" for the calculated Mode of the variable. Additionally, the observations were given labels with the values 1 for mobile and 2 for computers, ensuring that there were no errors or null values, and they were then converted to integer data types for use in future model construction.

```
array(['Mobile', 'Computer'], dtype=object)
```

Count of null values after null value treatment

```
Churn                     0
account_tenure            0
city_tier                 0
cust_care_contacts_12m    0
payment_method            0
gender                    0
service_score             0
customers_per_account     0
account_segment           0
cc_agent_score            0
marital_Status            0
revenue_per_month         0
account_complaints_12m    0
rev_growth_yoy            0
coupons_used              0
days_since_cc_contact     0
cashback                  0
login_device              0
dtype: int64
```

Fig. 20 After null value treatment

Assumptions:

We are assuming the categorical variables as given below:

payment_method:
Debit Card - 1
UPI – 2
Credit Card – 3
Cash on Delivery – 4
E Wallet – 5

gender:
F/Female – 1
M/Male – 2

account_segment:
Regular – 1
Regular + / Regular Plus - 2
Super – 3
Super + / Super Plus – 4
HNI - 5

marital_Status:
Single – 1
Divorced – 2
Married – 3

login_devices:
Mobile – 1
Computer – 2

account_tenure is considered in months
revenue_per_month is average revenue per month which is considered in Rupees
rev_growth_yoy is considered in percentage (%)
cashback is considered in Rupees

	0	1	2	3	4
Churn	1	1	1	1	1
account_tenure	4	0	0	0	0
city_tier	3	1	1	3	1
cust_care_contacts_12m	6	8	30	15	12
payment_method	1	2	1	1	3
gender	1	2	2	2	2
service_score	3	3	2	2	2
customers_per_account	3.0	4.0	4.0	4.0	3.0
account_segment	3	2	2	3	2
cc_agent_score	2	3	3	5	5
marital_Status	1	1	1	1	1
revenue_per_month	9.0	7.0	6.0	8.0	3.0
account_complaints_12m	1	1	1	0	0
rev_growth_yoy	11.0	15.0	14.0	23.0	11.0
coupons_used	1.0	0.0	0.0	0.0	1.0
days_since_cc_contact	5.0	0.0	3.0	3.0	3.0
cashback	159.0	120.0	165.0	134.0	129.0
login_device	1	1	1	1	1

Table. 2 Dataset sample after encoding

Variable transformation

- We can observe that the various variables have various dimensions. Similar to how the variable "cashback" signifies money and "cc_agent_score" denotes customer ratings. They also have different statistical ratings as a result.
- This data collection might benefit from scaling, which will equalize the data and bring the standard deviation nearly to zero.
- Data normalization using the MinMax scaling.

	0	1	2	3	4
Churn	1.000000	1.000000	1.000000	1.000000	1.000000
account_tenure	0.108108	0.000000	0.000000	0.000000	0.000000
city_tier	1.000000	0.000000	0.000000	1.000000	0.000000
cust_care_contacts_12m	0.054054	0.108108	0.702703	0.297297	0.216216
payment_method	0.000000	0.250000	0.000000	0.000000	0.500000
gender	0.000000	1.000000	1.000000	1.000000	1.000000
service_score	0.600000	0.600000	0.400000	0.400000	0.400000
customers_per_account	0.375000	0.625000	0.625000	0.625000	0.375000
account_segment	0.500000	0.250000	0.250000	0.500000	0.250000
cc_agent_score	0.250000	0.500000	0.500000	1.000000	1.000000
marital_Status	0.000000	0.000000	0.000000	0.000000	0.000000
revenue_per_month	0.666667	0.500000	0.416667	0.583333	0.166667
account_complaints_12m	1.000000	1.000000	1.000000	0.000000	0.000000
rev_growth_yoy	0.291667	0.458333	0.416667	0.791667	0.291667
coupons_used	0.285714	0.000000	0.000000	0.000000	0.285714
days_since_cc_contact	0.344828	0.000000	0.206897	0.206897	0.206897
cashback	0.430000	0.235000	0.460000	0.305000	0.280000
login_device	0.000000	0.000000	0.000000	0.000000	0.000000

Table 2 Dataset sample after minmax scaling

Standard scaling method is also implemented for scaling and producing clusters thereafter. The below image shows the data set after standard scaling

	0	1	2	3	4
Churn	2.222626	2.222626	2.222626	2.222626	2.222626
account_tenure	-0.704189	-1.153115	-1.153115	-1.153115	-1.153115
city_tier	1.479979	-0.709243	-0.709243	1.479979	-0.709243
cust_care_contacts_12m	-1.380526	-1.147274	1.418500	-0.330891	-0.680769
payment_method	-1.013677	-0.288991	-1.013677	-1.013677	0.435694
gender	-1.237690	0.807957	0.807957	0.807957	0.807957
service_score	0.135956	0.135956	-1.247623	-1.247623	-1.247623
customers_per_account	-0.769108	0.312718	0.312718	0.312718	-0.769108
account_segment	0.090912	-0.818288	-0.818288	0.090912	-0.818288
cc_agent_score	-0.770012	-0.041708	-0.041708	1.414899	1.414899
marital_Status	-1.376157	-1.376157	-1.376157	-1.376157	-1.376157
revenue_per_month	1.295405	0.601043	0.253861	0.948224	-0.787683
account_complaints_12m	1.617520	1.617520	1.617520	-0.618230	-0.618230
rev_growth_yoy	-1.384988	-0.321004	-0.587000	1.806965	-1.384988
coupons_used	-0.436018	-1.341042	-1.341042	-1.341042	-0.436018
days_since_cc_contact	0.126480	-1.305301	-0.446232	-0.446232	-0.446232
cashback	-0.414182	-1.300636	-0.277805	-0.982422	-1.096070
login_device	-0.602939	-0.602939	-0.602939	-0.602939	-0.602939

Table 3 Dataset after Standard scaling

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 11001 entries, 0 to 11259
Data columns (total 18 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Churn                   11001 non-null  float64
 1   account_tenure          11001 non-null  float64
 2   city_tier               11001 non-null  float64
 3   cust_care_contacts_12m  11001 non-null  float64
 4   payment_method          11001 non-null  float64
 5   gender                  11001 non-null  float64
 6   service_score           11001 non-null  float64
 7   customers_per_account   11001 non-null  float64
 8   account_segment         11001 non-null  float64
 9   cc_agent_score          11001 non-null  float64
 10  marital_Status          11001 non-null  float64
 11  revenue_per_month       11001 non-null  float64
 12  account_complaints_12m  11001 non-null  float64
 13  rev_growth_yoy          11001 non-null  float64
 14  coupons_used            11001 non-null  float64
 15  days_since_cc_contact   11001 non-null  float64
 16  cashback                11001 non-null  float64
 17  login_device            11001 non-null  float64
dtypes: float64(18)
memory usage: 1.8 MB
```

Fig. 20 Data information after minmax scaling

As we can see above, all the columns have been converted into float data type as part of minmax scaling

Standard Deviation Before and After Normalization:

Before

```
Standard deviation of variables
Churn                      0.374223
city_tier                  0.915015
cust_care_contacts_12m     8.853269
service_score              0.725584
cc_agent_score             1.379772
account_complaints_12m     0.451594
dtype: float64
```

After

```
Standard deviation of variables
Churn                      0.374192
account_tenure             0.240826
city_tier                  0.456804
cust_care_contacts_12m     0.231751
payment_method             0.344993
gender                     0.488865
service_score              0.144559
customers_per_account      0.231101
account_segment            0.274979
cc_agent_score             0.343279
marital_Status             0.446216
revenue_per_month          0.240039
account_complaints_12m     0.447297
rev_growth_yoy             0.156651
coupons_used               0.315712
days_since_cc_contact      0.240849
cashback                   0.219988
login_device               0.442208
dtype: float64
```

Fig. 21 Before and after normalization

We can see that the standard deviation of the variables is now very near to zero. We also transformed the variables to the int data type, which will aid in the process of creating the model in the future.

Addition of new variables:

At this time, we don't see the need to add any additional variables in the traditional sense. may be necessary at a later stage of model construction and can be made in that manner.

4) Business insights from EDA

a) Is the data unbalanced? If so, what can be done?

The presented data set is unbalanced. Our target variable, "Churn," has a considerable degree of variation in its categorical count. We have 9364 for "0" and 1896 for "1" in our count.

```
0    9149
1    1852
```

Fig. 21 Value counts of variable Churn

Using the SMOTE technique, it is possible to correct this dataset imbalance by generating more datapoints.

We must only use SMOTE on the train dataset; not the test dataset. As a standard business procedure, data were separated into train and test datasets in a 70:30 ratio (can be changed later as instructed).

Before SMOTE:

```
X_train (7700, 17)
X_test (3301, 17)
y_train (7700,)
y_test (3301,)
```

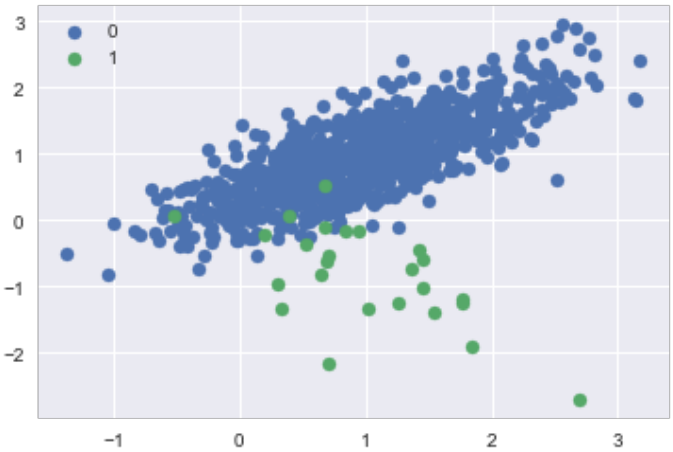


Fig. 22 Scatter plot before SMOTE

After SMOTE:

```
X_train_res (12812, 17)
y_train_res (12812,)
```

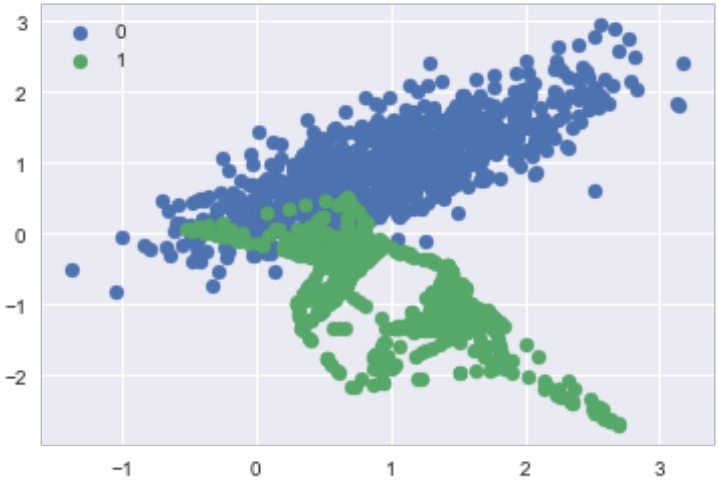


Fig. 23 Scatter plot after SMOTE

The increase in density of the green dots indicates the increase in data points.

b) Any business insights using clustering

- K-means cluster was used to create 3 clusters, and customers were divided into these 3 divisions.
- Based on the inertia value, 3 clusters were chosen.
- The second cluster has the highest customer counts, and the third cluster has the lowest.

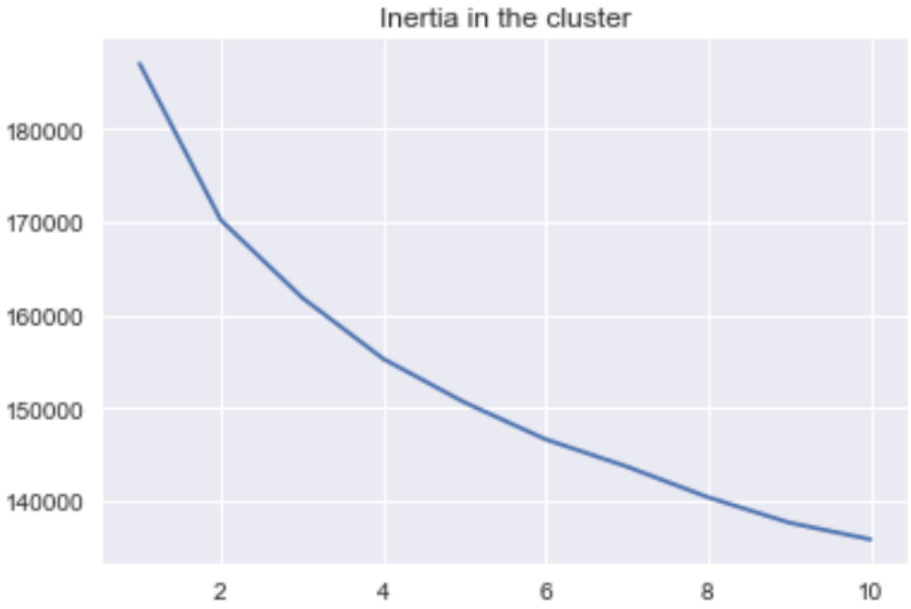


Fig. 24 Plotting clusters

Below table represents the clusters and the columns' count in each cluster

Clus_kmeans	0	1	2
account_tenure	2760	4976	3265
city_tier	2760	4976	3265
cust_care_contacts_12m	2760	4976	3265
payment_method	2760	4976	3265
gender	2760	4976	3265
service_score	2760	4976	3265
customers_per_account	2760	4976	3265
account_segment	2760	4976	3265
cc_agent_score	2760	4976	3265
marital_Status	2760	4976	3265
revenue_per_month	2760	4976	3265
account_complaints_12m	2760	4976	3265
rev_growth_yoy	2760	4976	3265
coupons_used	2760	4976	3265
days_since_cc_contact	2760	4976	3265
cashback	2760	4976	3265
login_device	2760	4976	3265

Fig. 25 K means clustering across all variables

Business insights

- The data collected shows a mix of services provided and customer ratings, indicating that there is room for improvement in the services offered.
- The business can increase its visibility in tier 2 cities to acquire new customers.
- By promoting payment methods such as standing instructions in bank accounts or UPI, the business can provide a hassle-free and secure experience for customers.
- The service scores show a lot of room for improvement and the business can conduct a survey to better understand customer expectations.
- Improving customer experience can be achieved by training customer care executives and providing them with the tools they need to deliver better service.
- By offering customized plans based on customer spend and tenure, the business can improve customer loyalty and retention.
- Offering family floater plans for married customers can be a good way to increase customer satisfaction and loyalty.
- These insights provide a starting point for the business to consider new strategies for improving customer experience and reducing churn. Further analysis and research may be necessary to determine the best approach for each business and customer segment.

- By understanding customer behavior and preferences, the business can tailor its services and marketing efforts to meet the needs and expectations of its target audience. This will not only improve customer satisfaction, but also drive business growth and success.
- Business can analyze the customer churning pattern based on gender and address the reasons why male customers are more likely to churn than female customers.
- Focus on improving the customer experience for the "Regular Plus" segment, as customers in this segment have shown a higher churn rate.
- Offer tailored packages for single customers, perhaps with added benefits or discounts to reduce churn rates among this customer segment.
- Conduct a detailed study of customer preferences for accessing services through mobile devices and see if there are any pain points that can be addressed to reduce churn rates among mobile users.
- Develop marketing strategies aimed at increasing customer retention and loyalty, such as loyalty programs or rewards for repeat business.
- Offer financial incentives or flexible payment options to reduce churn among customers who prefer debit or credit cards.
- Invest in technology to streamline customer service processes and provide faster and more efficient service, which could help to improve customer satisfaction and reduce churn.