CUSTOMER CHURN PREDICTION SOLUTION

Submitted By
Rejin CR
rejincr001@gmail.com

Introduction

Defining the Problem Statement

The e-commerce company is facing significant challenges in retaining its customer base due to intense market competition. Account churn, where an account (potentially representing multiple customers) ceases to engage with the company, is a critical issue. The primary goal is to develop a predictive model that identifies accounts at risk of churning, enabling the company to implement targeted retention strategies.

Need of the Study/Project

Retaining existing customers is more cost-effective than acquiring new ones. By predicting churn, the company can proactively address customer dissatisfaction, reduce attrition, and maintain revenue streams. This project aims to provide data-driven insights to create effective retention campaigns that balance customer satisfaction with financial viability.

Understanding Business/Social Opportunity

Addressing churn not only helps in sustaining revenue but also enhances customer loyalty and brand reputation. By understanding the factors contributing to churn, the company can improve its services and customer experience, leading to long-term business growth and a stronger market position.

Data Report

Understanding How Data Was Collected

The data was collected over a period of 12 months, capturing various customer interactions, account details, and service metrics. The frequency of data collection aligns with customer activities, such as service usage, customer care contacts, and payment transactions. The methodology involves aggregating data at the account level to provide a comprehensive view of customer behavior.

Visual Inspection of Data

The dataset comprises multiple variables, including account tenure, customer satisfaction scores, payment modes, and demographic details. Initial inspection reveals the presence of both continuous and categorical variables, with some potential missing values and outliers that need to be addressed during the analysis.

Understanding of Attributes

Key attributes include AccountID, Churn (target variable), Tenure, City_Tier, and various customer interaction metrics. Each variable has been defined clearly in the data dictionary, ensuring a proper understanding of the data structure and relevance to the churn prediction model.

Data Dictionary

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

Data Overview

```
Data - Structure

Shape           :(11260, 19)
Size            :213940
Dimension       :2
```

```
************************************************
Data – Info

Data columns (total 19 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   AccountID               11260 non-null  int64
 1   Churn                   11260 non-null  int64
 2   Tenure                  11158 non-null  object
 3   City_Tier               11148 non-null  float64
 4   CC_Contacted_LY         11158 non-null  float64
 5   Payment                 11151 non-null  object
 6   Gender                  11152 non-null  object
 7   Service_Score           11162 non-null  float64
 8   Account_user_count      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  rev_per_month           11158 non-null  object
 13  Complain_ly             10903 non-null  float64
 14  rev_growth_yoy          11260 non-null  object
 15  coupon_used_for_payment 11260 non-null  object
 16  Day_Since_CC_connect    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


Numerical Columns : 7
['AccountID', 'Churn', 'City_Tier', 'CC_Contacted_LY', 'Service_Score',
'CC_Agent_Score', 'Complain_ly']

Categorical Columns : 12
['Tenure', 'Payment', 'Gender', 'Account_user_count', 'account_segment',
'Marital_Status', 'rev_per_month', 'rev_growth_yoy',
'coupon_used_for_payment', 'Day_Since_CC_connect', 'cashback',
'Login_device']


************************************************
Data - Sample
```

Observations:

- The dataset has a total of **11,260 observations** and **19 features**.
- Of these 19 features:

- **7** are numerical datatypes.
- **12** are object types.
- However, this can be reconsidered lated based on the nature and distribution of the data

- The `AccountID` column can be considered as the unique identifier in the dataset.
- From the data sample, it is observed that there are **missing values** in the dataset.

Statistical Summary of Numerical Columns

	count	mean	std	min	25%	50%	75%	max
AccountID	11260	25629.5	3250.63	20000	22814.8	25629.5	28444.3	31259
Churn	11260	0.16838	0.37422	0	0	0	0	1
City_Tier	11148	1.65393	0.91502	1	1	1	3	3
CC_Contacted_LY	11158	17.8671	8.85327	4	11	16	23	132
Service_Score	11162	2.90253	0.72558	0	2	3	3	5
CC_Agent_Score	11144	3.06649	1.37977	1	2	3	4	5
Complain_ly	10903	0.28533	0.45159	0	0	0	1	1

Statistical Summary of Categorical Columns

	count	unique	top	freq
Tenure	11158	38	1	1351
Payment	11151	5	Debit Card	4587
Gender	11152	4	Male	6328
Account_user_count	11148	7	4	4569
account_segment	11163	7	Super	4062
Marital_Status	11048	3	Married	5860
rev_per_month	11158	59	3	1746
rev_growth_yoy	11260	20	14	1524
coupon_used_for_payment	11260	20	1	4373
Day_Since_CC_connect	10903	24	3	1816
cashback	10789	5693	155.62	10
Login_device	11039	3	Mobile	7482

Observations:

- `Churn` :
  - There are no missing values.

    - The mean churn rate is 16.84% (mean = 0.168384), indicating that approximately 16.84% of customers have churned.
- `City_Tier`:
    - The mean city tier is 1.65, with a standard deviation of 0.915.
    - The values range from 1 to 3, indicating three tiers of cities. The 50% of the data is from city tier 1
- `CC_Contacted_LY`:
    - Customers contacted the call center 17.87 times on an average last year, with a standard deviation of 8.85.
    - The range is from 4 to 132 contacts with 75% of the data lies in 23 indicating the data is strongly right skewed
    - Also there are high chances of outliers as well.
- `Service_Score`:
    - The average service score is 2.90 (which is not so good), with a standard deviation of 0.73. Scores range from 0 to 5.
- `CC_Agent_Score`:
    - The average call center agent score is 3.07, with a standard deviation of 1.38. Scores range from 1 to 5.
- `Complain_ly`
    - About 28.53% of customers complained last year.
- `Tenure` :
    - The Tenure column has 38 unique values, with the most frequent value being 1 (appearing 1,351 times).
- `Payment`:
    - There are 5 unique payment methods.
    - The most frequent payment method is Debit Card.
- `Gender`:
    - There are 4 unique gender categories.(However, this could be also due to some typing mistake as well)
    - The most frequent gender is Male.
- `Account_user_count`:
    - The user count within an account has 7 unique values, with the most frequent value being 4 (appearing 4569 times).
- `account_segment`:
    - The most common account segment is "Super," with 4,062 occurrences.
- `Marital_Status`:
    - There are 3 unique marital statuses. The majority of customers are Married, with 5,860 occurrences.
- `rev_per_month`:
- The most frequent monthly revenue value is 3, appearing 1,746 times. However, this might need to convert to numerical feature.
- `rev_growth_yoy`:
- The most frequent year-over-year revenue growth value is 14, appearing 1,524 times. This again might need to convert to numerical feature after further analysis.
- `coupon_used_for_payment`:
    - The most frequent value is 1, appearing 4,373 times. This suggests a common behavior in using coupons for payments.
- `Day_Since_CC_connect`:

- The most frequent value is 3, appearing 1,816 times. This indicates a common recency of customer service contact.
- `cashback`
- The most frequent cashback value is 155.62, appearing 10 times. This suggests a specific cashback amount is popular. However, this also might need to be converted to numerical type after detailed analysis
- `Login_device`:
    - There are 3 unique login devices.
    - The majority of logins are from Mobile devices, with 7,482 occurrences. This indicates a strong preference for mobile access.

Overall:

- The dataset is imbalanced, with 16.84% churn rate.
- Several columns have missing values and outliers, which need to be handled
- Categorical features like Payment, Gender, account_segment, and Marital_Status should be encoded
- Features like CC_Contacted_LY, Service_Score, CC_Agent_Score, and Complain_ly may strongly influence churn prediction.
- Tenure and rev_per_month could also be significant predictors.

Data Preprocessing

Checking the Duplicate Values in each columns

There are no duplicate records found in the data set

Checking the Unique Values in each columns

```
# @title
check_unique_values(data)
**************************************************
AccountID
20000    1
27510    1
27502    1
27503    1
27504    1
        ..
23754    1
23755    1
23756    1
23757    1
31259    1
Name: count, Length: 11260, dtype: int64

Column Name: AccountID
Data Type: int64
Total Count: 11260
```

```
Unique Count: 11260

*************************************************
Churn
0    9364
1    1896
Name: count, dtype: int64

Column Name: Churn
Data Type: int64
Total Count: 11260
Unique Count: 2

*************************************************
Tenure
1     1351
0     1231
8      519
9      496
7      450
10     423
3      410
5      403
4      403
11     388
6      363
12     360
13     359
2      354
14     345
15     311
16     291
19     273
18     253
20     217
17     215
21     170
23     169
22     151
24     147
28     137
30     137
27     131
99     131
26     122
#      116
25     114
29     114
31      96
```

```
50       2
60       2
51       2
61       2
Name: count, dtype: int64

Column Name: Tenure
Data Type: object
Total Count: 11260
Unique Count: 39

**************************************************
City_Tier
1.0    7263
3.0    3405
2.0     480
Name: count, dtype: int64

Column Name: City_Tier
Data Type: float64
Total Count: 11260
Unique Count: 4

**************************************************
CC_Contacted_LY
14.0     682
16.0     663
9.0      655
13.0     655
15.0     623
12.0     571
8.0      538
17.0     525
11.0     524
10.0     489
7.0      391
18.0     374
19.0     364
20.0     319
6.0      311
21.0     310
22.0     282
23.0     241
24.0     214
25.0     197
32.0     192
29.0     181
28.0     178
34.0     178
```

```
30.0     175
27.0     174
26.0     169
35.0     165
31.0     165
33.0     155
36.0     148
37.0      96
38.0      73
39.0      55
40.0      46
42.0      30
41.0      29
43.0       8
5.0        8
127.0      1
126.0      1
132.0      1
4.0        1
129.0      1
Name: count, dtype: int64

Column Name: CC_Contacted_LY
Data Type: float64
Total Count: 11260
Unique Count: 45

**************************************************
Payment
Debit Card          4587
Credit Card         3511
E wallet            1217
Cash on Delivery    1014
UPI                  822
Name: count, dtype: int64

Column Name: Payment
Data Type: object
Total Count: 11260
Unique Count: 6

**************************************************
Gender
Male      6328
Female    4178
M          376
F          270
Name: count, dtype: int64
```

```
Column Name: Gender
Data Type: object
Total Count: 11260
Unique Count: 5

**************************************************
Service_Score
3.0    5490
2.0    3251
4.0    2331
1.0      77
0.0       8
5.0       5
Name: count, dtype: int64

Column Name: Service_Score
Data Type: float64
Total Count: 11260
Unique Count: 7

**************************************************
Account_user_count
4    4569
3    3261
5    1699
2     526
1     446
@     332
6     315
Name: count, dtype: int64

Column Name: Account_user_count
Data Type: object
Total Count: 11260
Unique Count: 8

**************************************************
account_segment
Super           4062
Regular Plus    3862
HNI             1639
Super Plus       771
Regular          520
Regular +        262
Super +           47
Name: count, dtype: int64

Column Name: account_segment
Data Type: object
```

```
Total Count: 11260
Unique Count: 8

**************************************************
CC_Agent_Score
3.0    3360
1.0    2302
5.0    2191
4.0    2127
2.0    1164
Name: count, dtype: int64

Column Name: CC_Agent_Score
Data Type: float64
Total Count: 11260
Unique Count: 6

**************************************************
Marital_Status
Married     5860
Single      3520
Divorced    1668
Name: count, dtype: int64

Column Name: Marital_Status
Data Type: object
Total Count: 11260
Unique Count: 4

**************************************************
rev_per_month
3      1746
2      1585
5      1337
4      1218
6      1085
7       754
+       689
8       643
9       564
10      413
1       402
11      278
12      166
13       93
14       48
15       24
102       8
123       5
```

```
124       5
107       5
136       4
140       4
118       4
133       4
129       4
115       3
117       3
138       3
101       3
110       3
137       3
119       3
108       3
127       3
116       3
126       3
130       3
113       3
120       2
19        2
131       2
139       2
114       2
125       2
22        2
121       2
105       2
134       2
20        1
23        1
122       1
21        1
104       1
25        1
135       1
111       1
109       1
100       1
103       1
Name: count, dtype: int64

Column Name: rev_per_month
Data Type: object
Total Count: 11260
Unique Count: 60

**************************************************
```

```
Complain_ly
0.0    7792
1.0    3111
Name: count, dtype: int64

Column Name: Complain_ly
Data Type: float64
Total Count: 11260
Unique Count: 3

**************************************************
rev_growth_yoy
14    1524
13    1427
15    1283
12    1210
16     949
18     708
17     704
19     619
20     562
11     523
21     433
22     403
23     345
24     229
25     188
26      98
27      35
28      14
$        3
4        3
Name: count, dtype: int64

Column Name: rev_growth_yoy
Data Type: object
Total Count: 11260
Unique Count: 20

**************************************************
coupon_used_for_payment
1     4373
2     2656
0     2150
3      698
4      424
5      284
6      234
7      184
```

```
8       88
10      34
9       34
11      30
12      26
13      22
14      12
15       4
16       4
#        1
$        1
*        1
Name: count, dtype: int64

Column Name: coupon_used_for_payment
Data Type: object
Total Count: 11260
Unique Count: 20

**************************************************
Day_Since_CC_connect
3     1816
2     1574
1     1256
8     1169
0      964
7      911
4      893
9      622
5      479
10     339
6      229
11     183
12     146
13     117
14      74
15      37
17      34
16      26
18      26
30       2
31       2
47       2
$        1
46       1
Name: count, dtype: int64

Column Name: Day_Since_CC_connect
Data Type: object
```

```
Total Count: 11260
Unique Count: 25

**************************************************
cashback
155.62    10
149.36     9
154.73     9
145.08     9
149.68     9
          ..
131.55     1
245.64     1
130.78     1
299.72     1
191.42     1
Name: count, Length: 5693, dtype: int64

Column Name: cashback
Data Type: object
Total Count: 11260
Unique Count: 5694

**************************************************
Login_device
Mobile      7482
Computer    3018
&&&&         539
Name: count, dtype: int64

Column Name: Login_device
Data Type: object
Total Count: 11260
Unique Count: 4
```

Observations:

- **Invalid Entries to be Updated as `NaN`**

The following columns contain invalid entries that need to be replaced with `NaN`:

- **Tenure**: Invalid record (`#`)
- **Account_user_count**: Invalid record (`@`)
- **rev_per_month**: Invalid record (`+`)
- **rev_growth_yoy**: Invalid record (`$`)
- **coupon_used_for_payment**: Invalid records (`$`, `#`, `*`)
- **Day_Since_CC_connect**: Invalid record (`$`)
- **cashback**: Invalid record (`$`)

- **Login_device**: Invalid record (&&&&)
- All the above features, except Login_device, can be converted to numeric after cleaning.
- **Duplicated Categories to be Combined**

  The following columns contain duplicated records that can be combined to ensure consistency:

  - **Gender**:

  Combine similar values like "Male" and "M" into a single category (e.g., "Male")..

  - **account_segment**:

  Combine similar values like "Super Plus" and "Super +" into a single category (e.g., "Super Plus").

Handling Columns with Incorrect/Invalid Entries

```
Converted all the columns to Numeric and corrected the invalid entries
```

Handling Columns that require refining the unique values

```
Updated the columns that require refining the unique values
```

Handling Numerical Columns that needs to be treated as Categorical

```
Converted certain columns to Categorical
```

Removing the columns that are not necessary

As the AccountID is just a unique identifier of the account, it does not add any significance in identifying patterns.

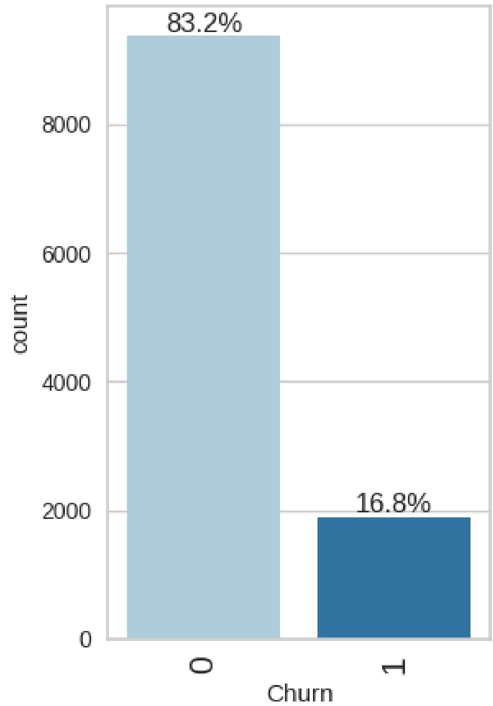
So we can remove the AccountID from the data set.

```
Numerical Columns : 10
['Churn', 'Tenure', 'CC_Contacted_LY', 'Account_user_count',
'rev_per_month', 'Complain_ly', 'rev_growth_yoy',
'coupon_used_for_payment', 'Day_Since_CC_connect', 'cashback']

Categorical Columns : 8
['City_Tier', 'Payment', 'Gender', 'Service_Score', 'account_segment',
'CC_Agent_Score', 'Marital_Status', 'Login_device']
```

Exploratory Data Analysis

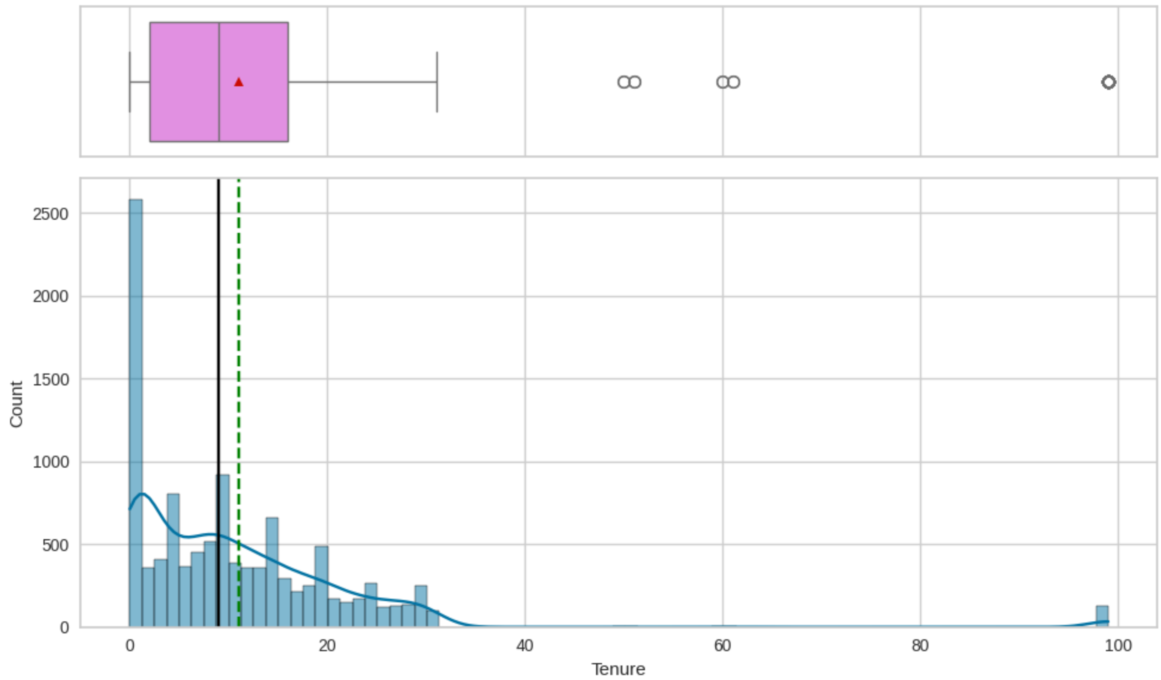
Univariate Analysis

```
Analysis for column: 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
Missing values: 0
Unique values: 2
```

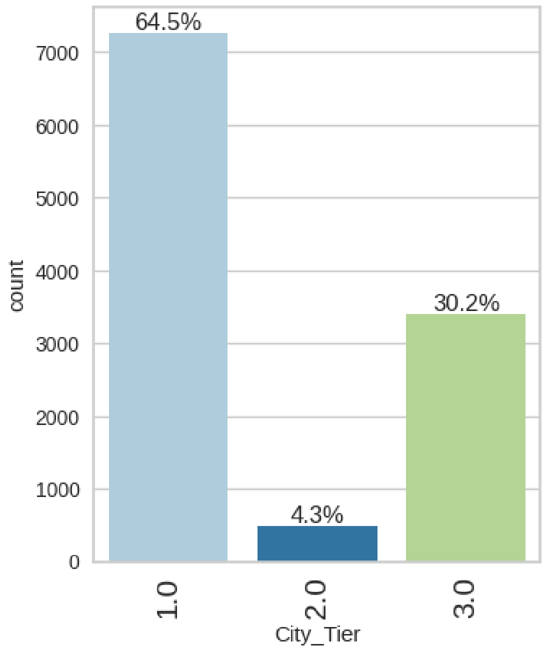



```
Analysis for column: Tenure
----------------------------------------
count    11042.000000
mean        11.025086
std         12.879782
min          0.000000
```

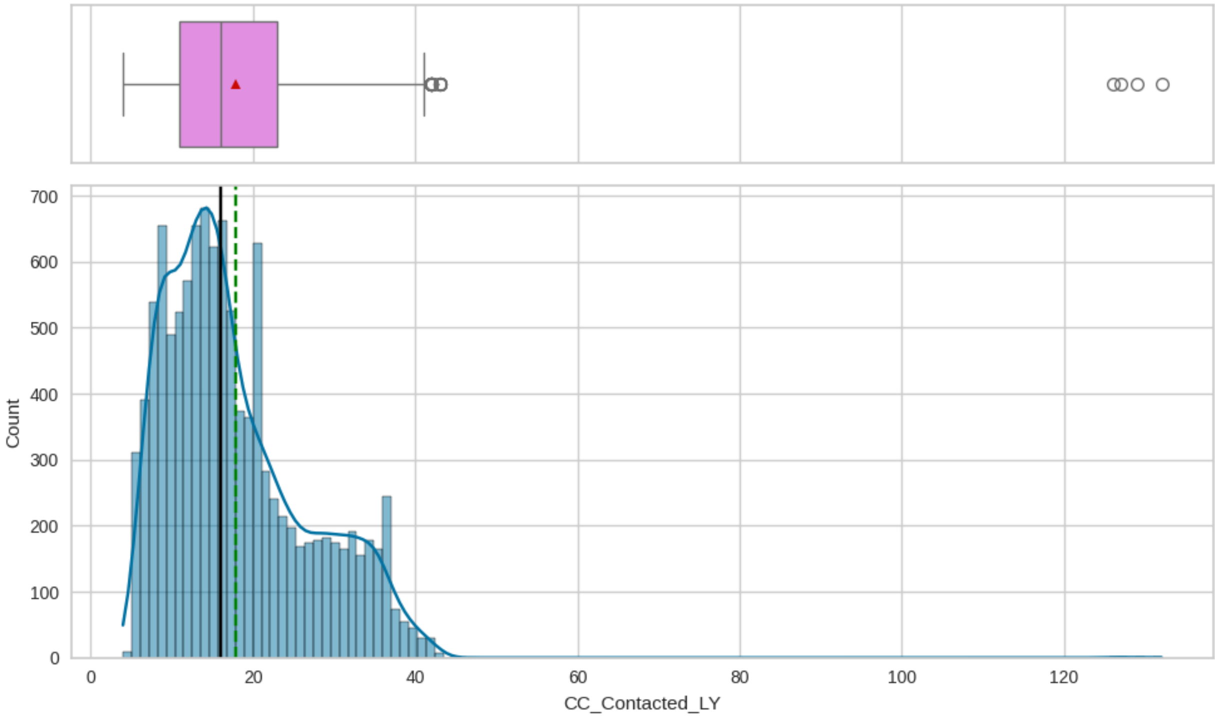
```
25%            2.000000
50%            9.000000
75%           16.000000
max           99.000000

Name: Tenure,
dtype: float64
Missing values: 218
Unique values: 37
```



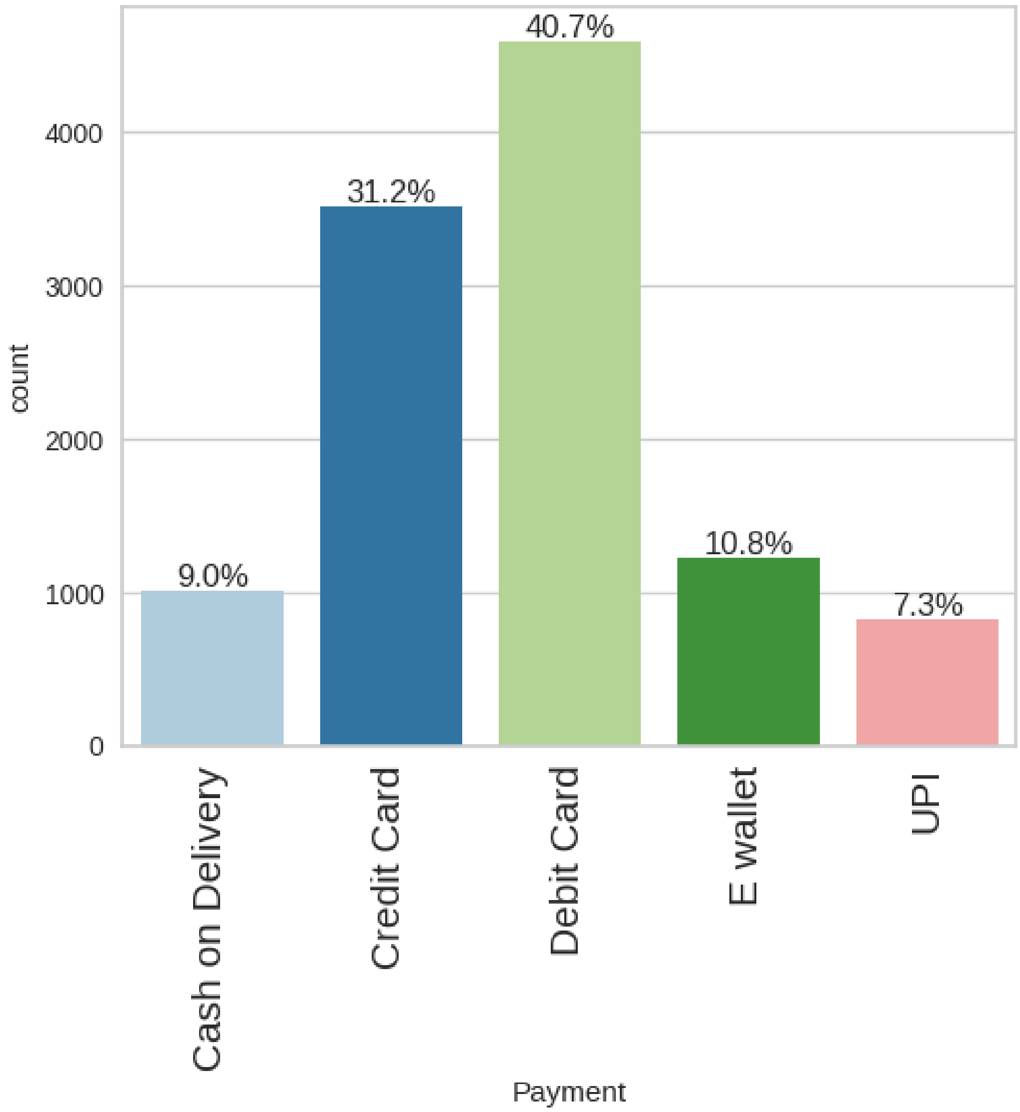

```
Analysis for column: City_Tier
----------------------------------------
count      11148.0
unique         3.0
top            1.0
freq        7263.0

Name: City_Tier
dtype: float64
Missing values: 112
Unique values: 3
```



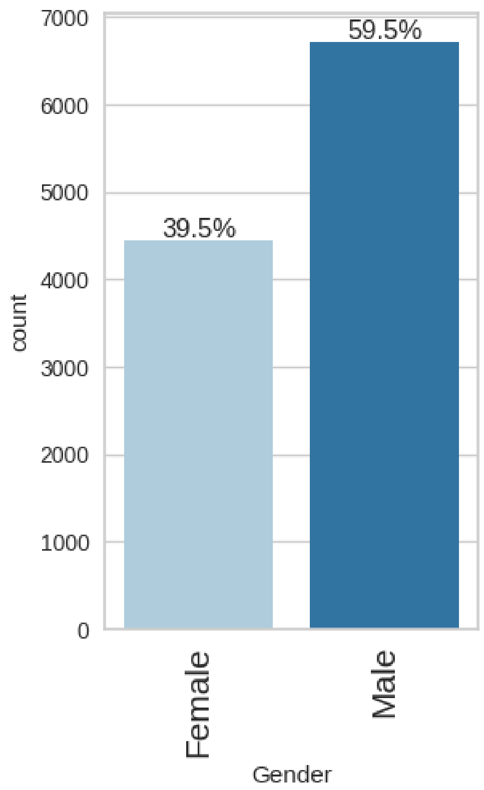

```
Analysis for column: CC_Contacted_LY
----------------------------------------
count    11158.000000
mean        17.867091
std          8.853269
min          4.000000
25%         11.000000
50%         16.000000
75%         23.000000
max        132.000000

Name: CC_Contacted_LY
dtype: float64
Missing values: 102
Unique values: 44
```



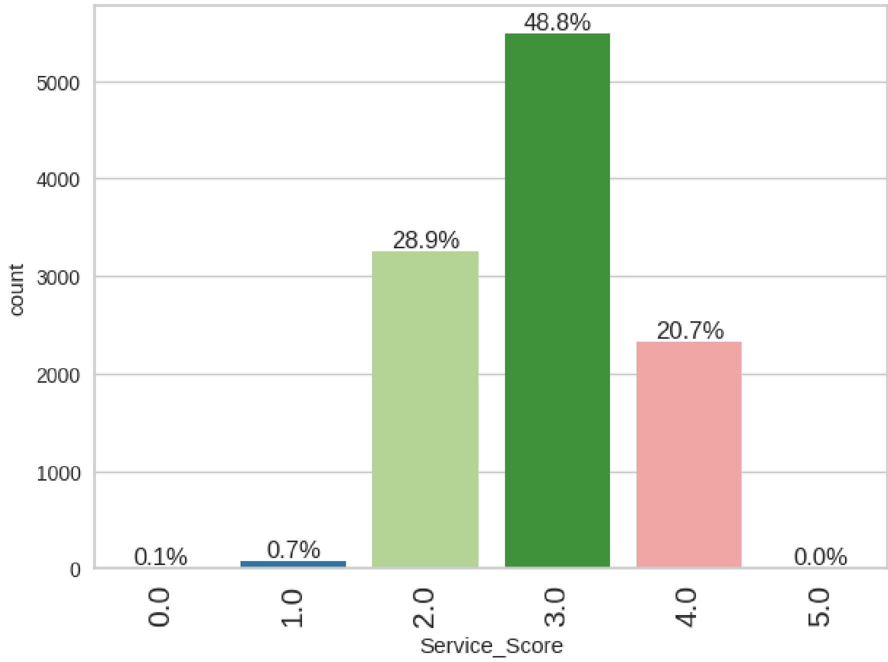

```
Analysis for column: Payment
----------------------------------------
count           11151
unique              5
top        Debit Card
freq             4587

Name: Payment
dtype: object
Missing values: 109
Unique values: 5
```



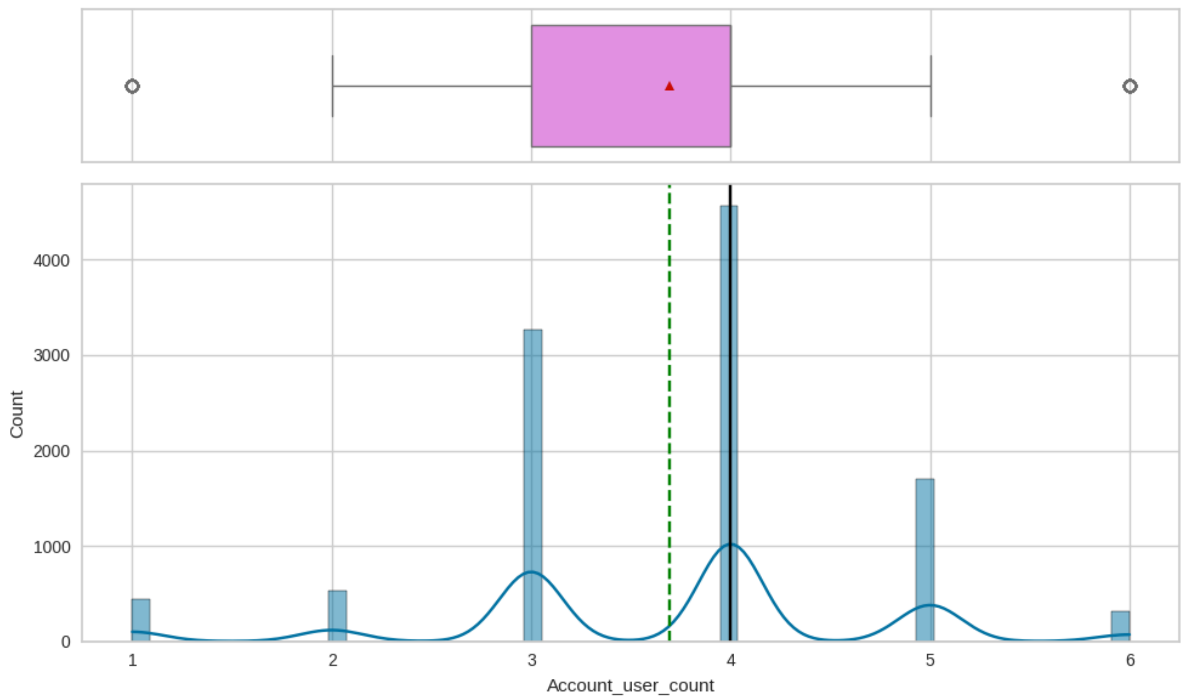

```
Analysis for column: Gender
----------------------------------------
count     11152
unique        2
top        Male
freq       6704

Name: Gender
dtype: object
Missing values: 108
Unique values: 2
```



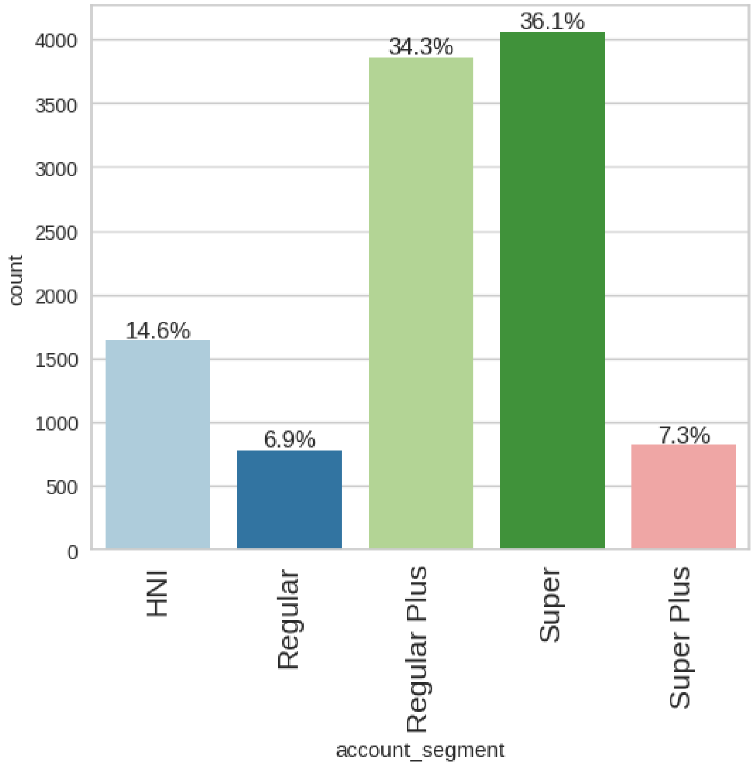

```
Analysis for column: Service_Score
----------------------------------------
count     11162.0
unique        6.0
top           3.0
freq       5490.0

Name: Service_Score
dtype: float64
Missing values: 98
Unique values: 6
```



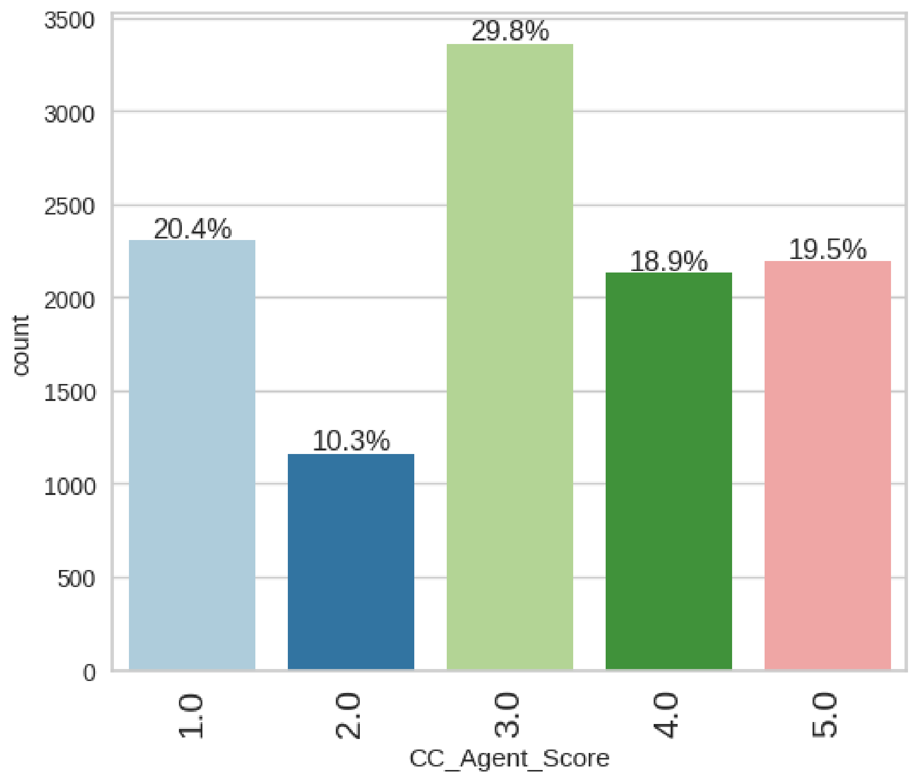

```
Analysis for column: Account_user_count
----------------------------------------
count    10816.000000
mean         3.692862
std          1.022976
min          1.000000
25%          3.000000
50%          4.000000
75%          4.000000
max          6.000000

Name: Account_user_count
dtype: float64
Missing values: 444
Unique values: 6
```



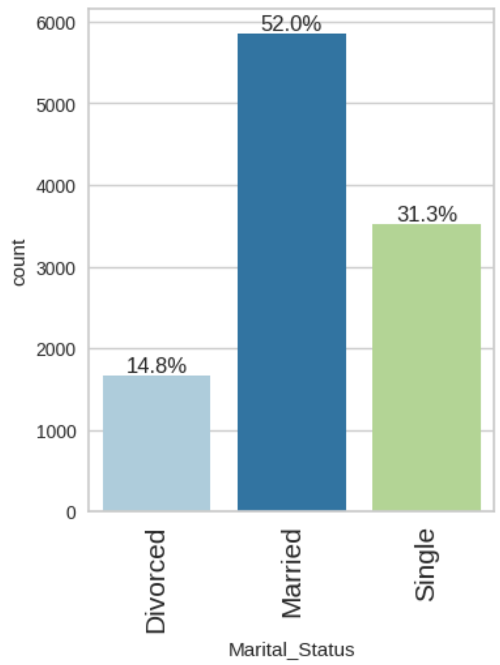

```
Analysis for column: account_segment
----------------------------------------
count      11163
unique         5
top        Super
freq        4062

Name: account_segment
dtype: object
Missing values: 97
Unique values: 5
```



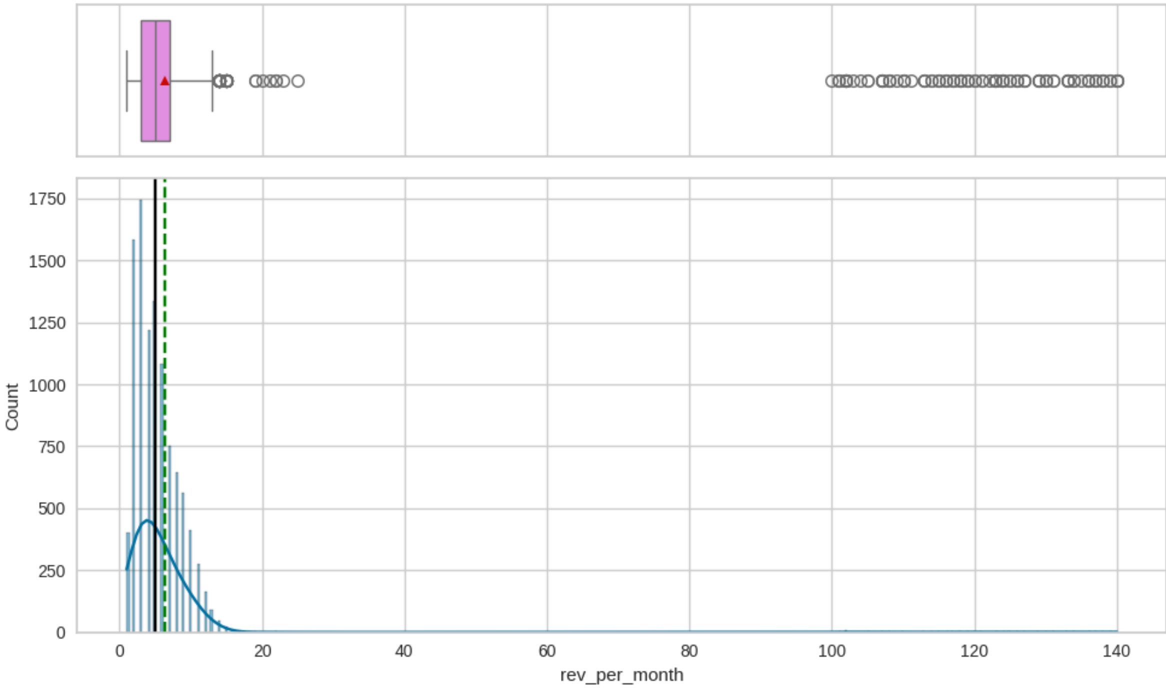

```
Analysis for column: CC_Agent_Score
----------------------------------------
count     11144.0
unique        5.0
top           3.0
freq       3360.0

Name: CC_Agent_Score
dtype: float64
Missing values: 116
Unique values: 5
```



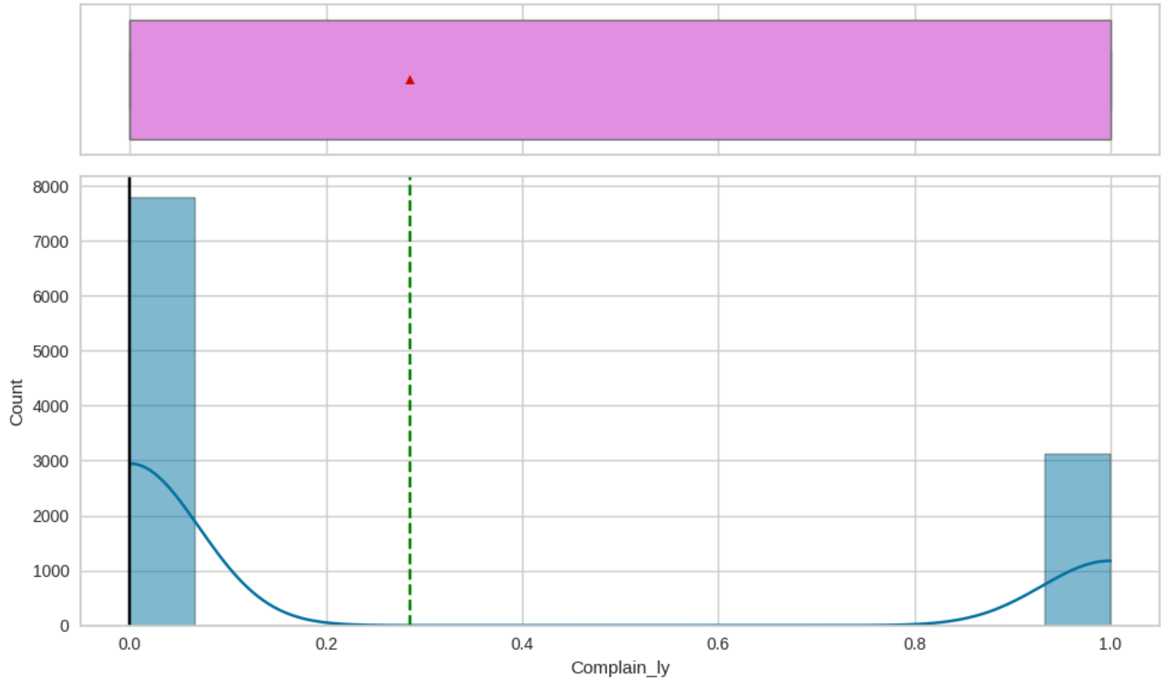

```
Analysis for column: Marital_Status
----------------------------------------
count        11048
unique           3
top        Married
freq          5860

Name: Marital_Status
dtype: object
Missing values: 212
Unique values: 3
```



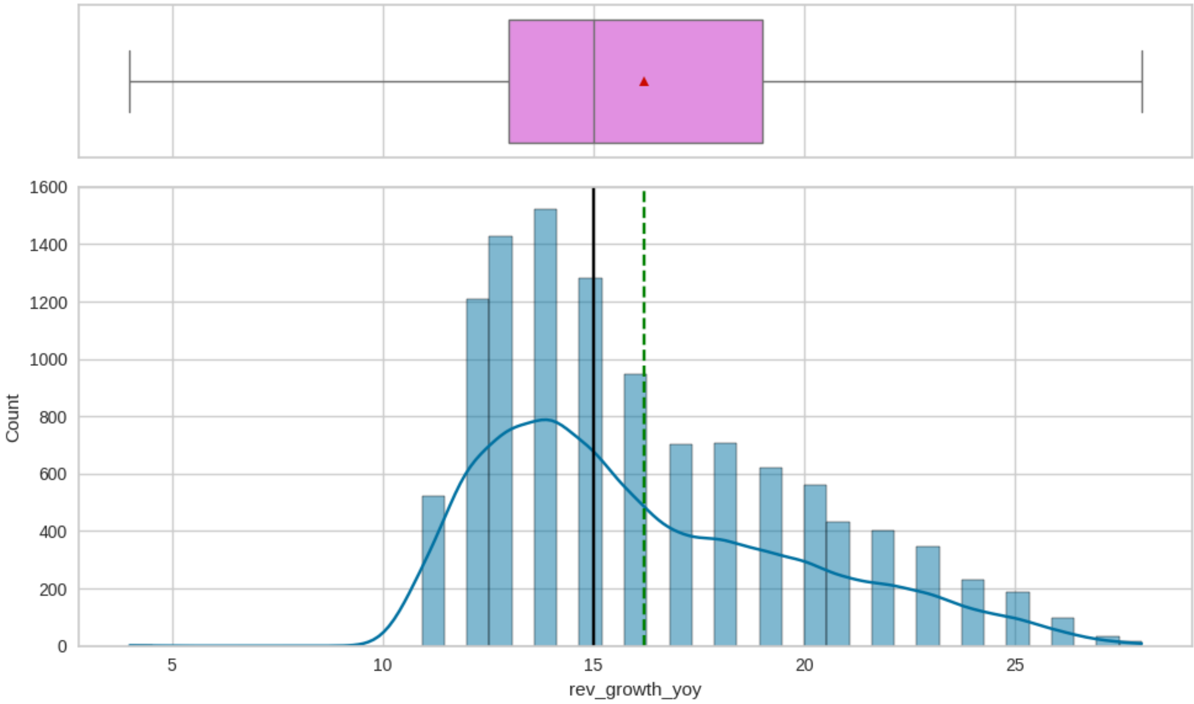

```
Analysis for column: rev_per_month
----------------------------------------
count    10469.000000
mean         6.362594
std         11.909686
min          1.000000
25%          3.000000
50%          5.000000
75%          7.000000
max        140.000000

Name: rev_per_month
dtype: float64
Missing values: 791
Unique values: 58
```



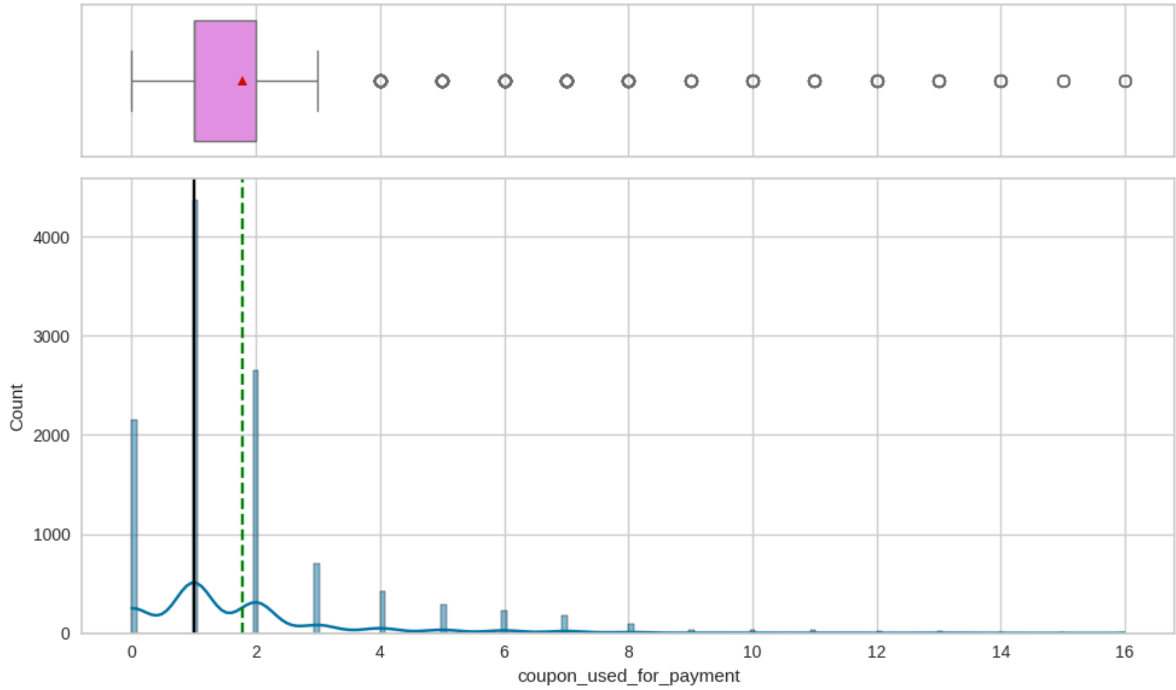

```
Analysis for column: Complain_ly
----------------------------------------
count    10903.000000
mean         0.285334
std          0.451594
min          0.000000
25%          0.000000
50%          0.000000
75%          1.000000
max          1.000000

Name: Complain_ly
dtype: float64
Missing values: 357
Unique values: 2
```



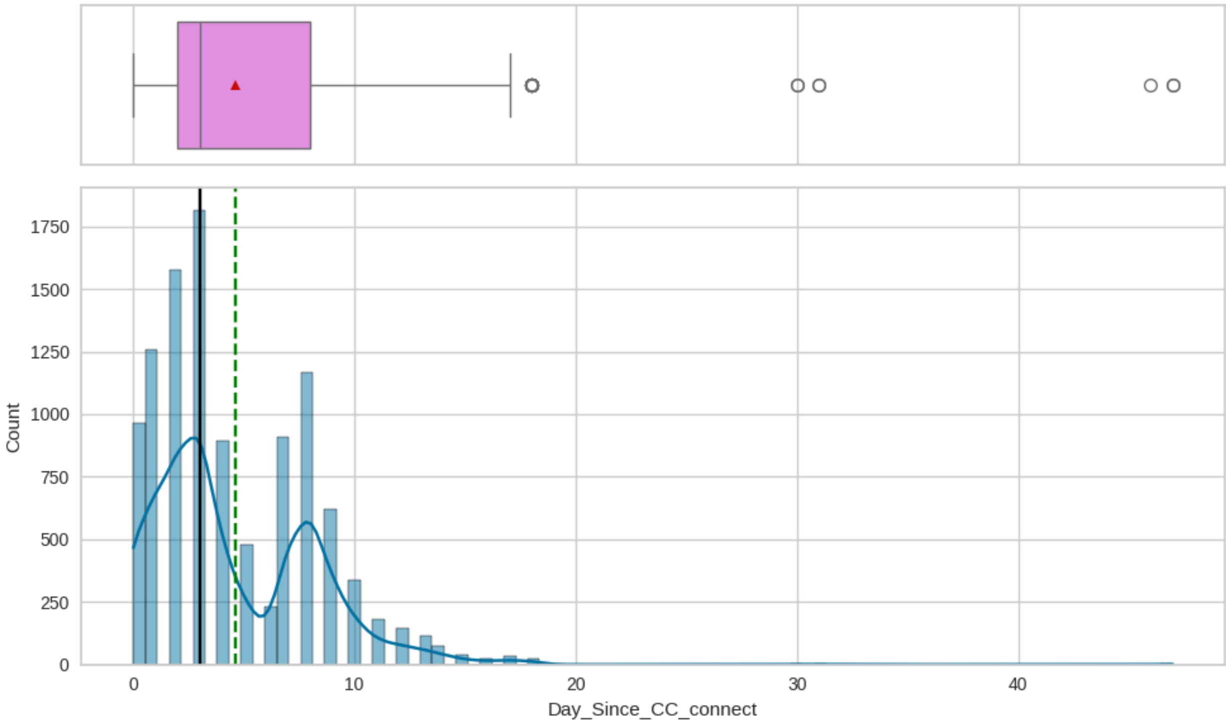

```
Analysis for column: rev_growth_yoy
----------------------------------------
count    11257.000000
mean        16.193391
std          3.757721
min          4.000000
25%         13.000000
50%         15.000000
75%         19.000000
max         28.000000

Name: rev_growth_yoy
dtype: float64
Missing values: 3
Unique values: 19
```



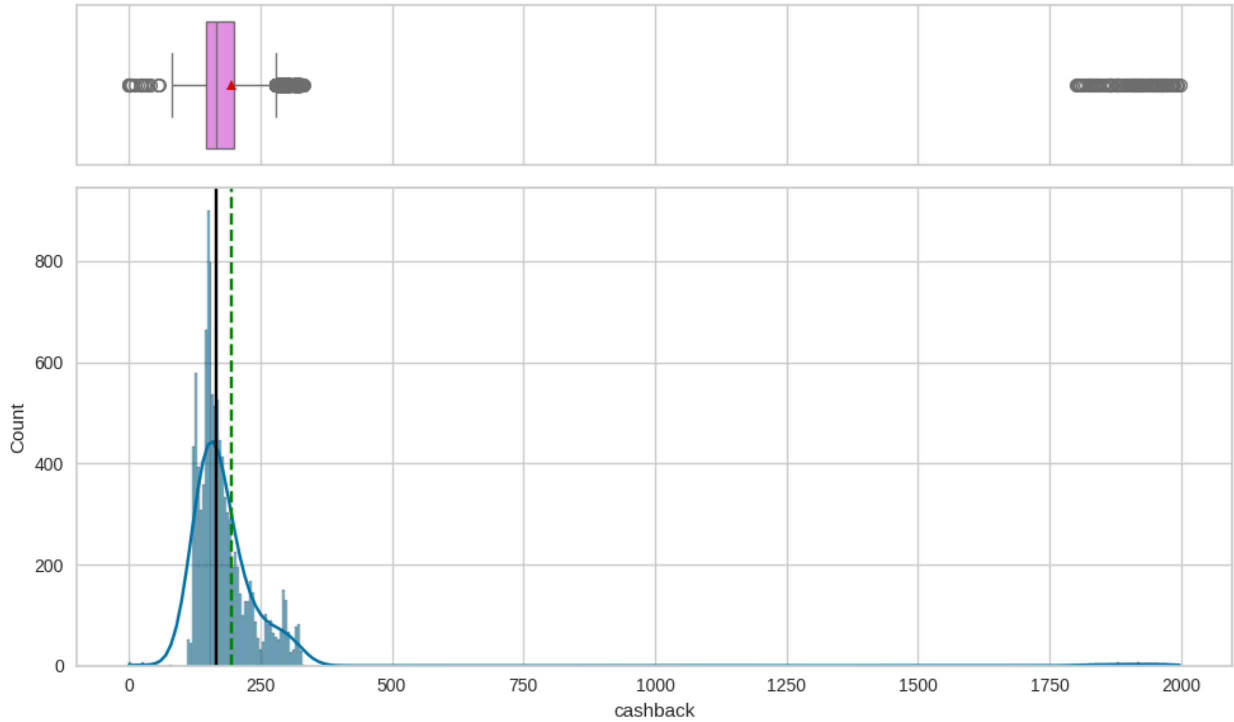

```
Analysis for column: coupon_used_for_payment
----------------------------------------
count     11257.000000
mean          1.790619
std           1.969551
min           0.000000
25%           1.000000
50%           1.000000
75%           2.000000
max          16.000000

Name: coupon_used_for_payment
dtype: float64
Missing values: 3
Unique values: 17
```



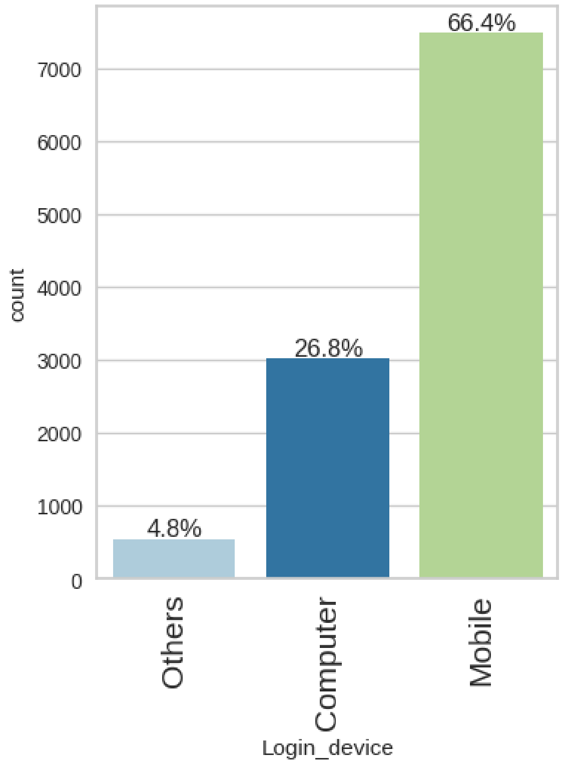

```
Analysis for column: Day_Since_CC_connect
----------------------------------------
count    10902.000000
mean         4.633187
std          3.697637
min          0.000000
25%          2.000000
50%          3.000000
75%          8.000000
max         47.000000

Name: Day_Since_CC_connect
dtype: float64
Missing values: 358
Unique values: 23
```



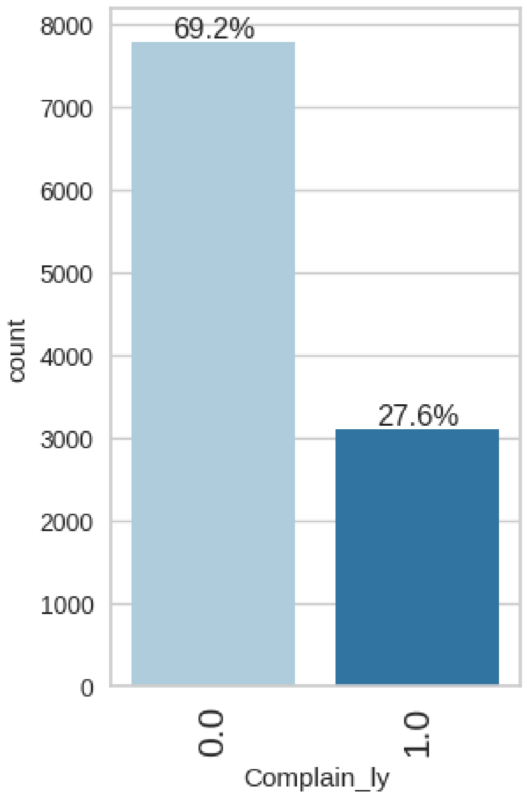

```
Analysis for column: cashback
----------------------------------------
count    10787.000000
mean       196.236370
std        178.660514
min          0.000000
25%        147.210000
50%        165.250000
75%        200.010000
max       1997.000000

Name: cashback
dtype: float64
Missing values: 473
Unique values: 5692
```




```
Analysis for column: Login_device
----------------------------------------
count      11039
unique         3
top       Mobile
freq        7482

Name: Login_device
dtype: object
Missing values: 221
Unique values: 3
```




```
Analysis for column: Complain_ly
----------------------------------------
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: Complain_ly
dtype: float64
Missing values: 357
Unique values: 2
```



Observations:

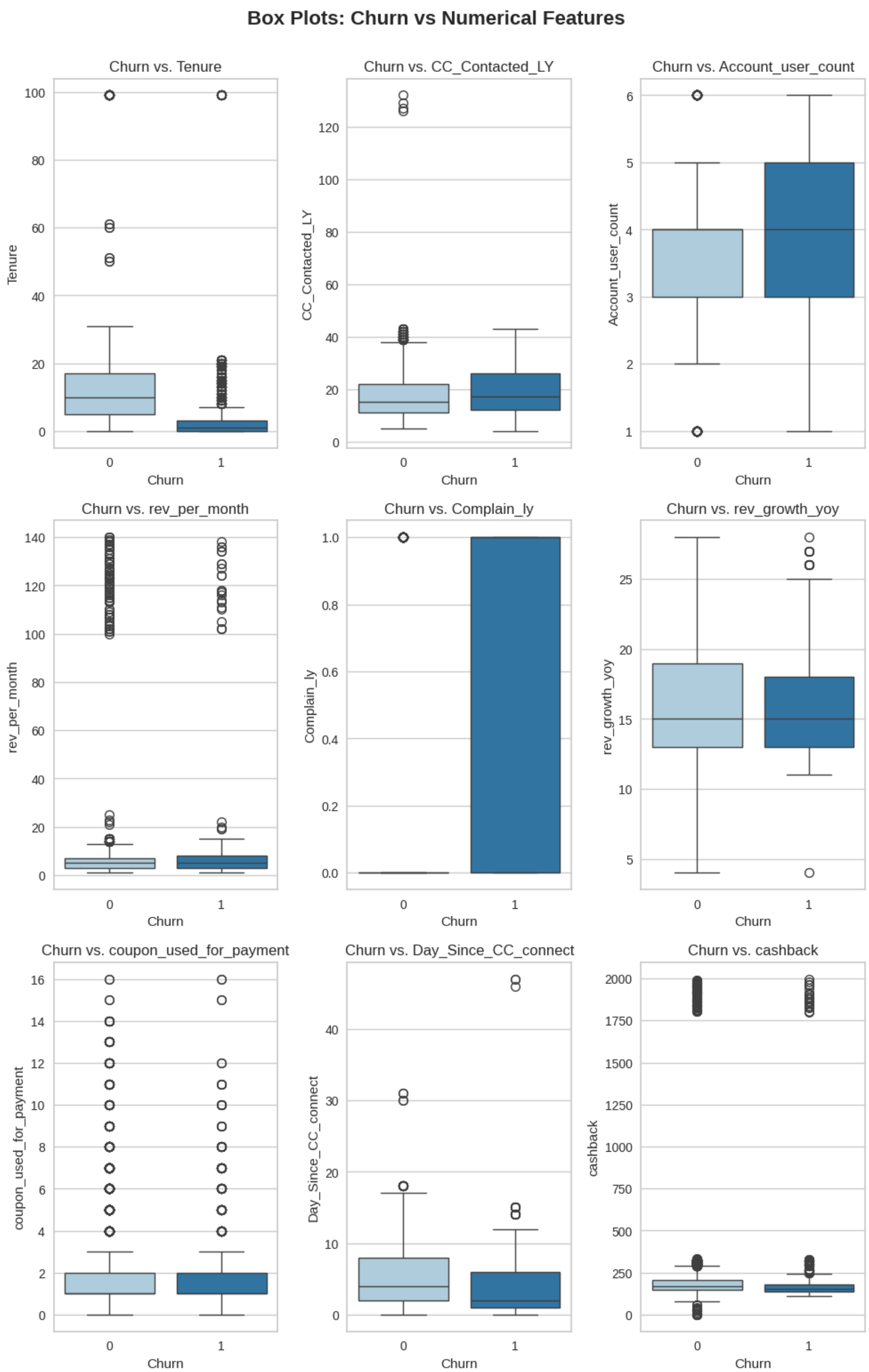
- **churn:**
    - This will be considered as the output variable
    - The Churn column is binary, with 0 (no churn) and 1 (churn).
    - No missing values are present in the column.
    - The dataset is highly imbalanced, with 83.16% of customers not churning and 16.84% churning.
    - We need to consider oversampling techniques to address the class imbalance.
- **Tenure:**
    - The Tenure column has a right-skewed distribution, with most customers having low tenure (0–20 months).
    - The average tenure is approximately 11 months.
    - There are 218 missing values and outliers (e.g., 99 months) that need to be addressed.
- **City_Tier**
    - Majority of customers are from Tier 1 cities with the frequency (around 7000).
    - City Tier 2.0 and City Tier 3.0 have significantly lower frequencies.
    - There are 112 missing values that need to be addressed.
- **CC_Contacted_LY:**
    - The CC_Contacted_LY column has a right-skewed distribution, with most customers being contacted 10–30 times.
    - There are 102 missing values and outliers (e.g., 132 contacts) that need to be addressed.

- **Payment:**
    - Majority of customers prefer using Debit Card for payments.
    - Credit Card is the second most frequent payment method.
    - E-wallet, Cash on Delivery, and UPI have significantly lower frequencies.
    - There are 109 missing values that need to be addressed
- **Gender:**
    - Gender distribution is heavily skewed toward Male customers, with fewer Female customers.
    - There are 109 missing values that need to be addressed
- **Service_Score:**
    - Distribution is heavily skewed with most customers rating the service as 3.0., and fewer customers giving other scores.
    - There are 98 missing values that need to be addressed..
- **Account_user_count:**
    - The distribution is concentrated around 4 users per account, with most accounts having 3–5 users.
    - There are 445 missing values that need to be addressed.
    - There are no significant outliers, as the data is tightly clustered around the median.
- **Account_segment:**
    - Majority of customers belong to the Super account segment.
    - Regular Plus is the second most frequent account segment.
    - HNI, Super Plus, and Regular have significantly lower frequencies.
    - There are 97 missing values that need to be addressed.
- **CC_Agent_Score:**
    - Majority of customers rated the agent as 3.0.
    - Other agent scores (e.g., 1.0, 2.0, 4.0, 5.0) have much lower frequencies.
    - There are 116 missing values that need to be addressed.
- **Marital_Status:**
    - Data is heavily skewed towards Married customers,with fewer Single and least no. of Divorced customers.
    - There are 212 missing values that need to be addressed.
- **rev_per_month:**
- The rev_per_month column has a right-skewed distribution, with most customers generating low revenue (1–20).
- There are 791 missing values and outliers (e.g., 140) that need to be addressed.
- **Complain_ly:**
    - The data is heavily skewed with most customers not filing a complaint in the last year.
    - There are 357 missing values that need to be addressed.
- **rev_growth_yoy:**
    - The majority of customers have a revenue growth between 10% and 20%, with a peak around 15%.
    - The average revenue growth year-over-year is approximately 16.19%.
    - There are 3 missing values and a few outliers (e.g., 28%) that need to be addressed.
- **coupon_used_for_payment:**
    - The coupon_used_for_payment column has a right-skewed distribution, with most customers using 1–2 coupons.

  - There are 3 missing values and outliers (e.g., 16 coupons) that need to be addressed.
- **Day_Since_CC_connect:**
  - The Day_Since_CC_connect column has a right-skewed distribution, with most customers connecting recently (within the last 10 days).
  - The average number of days since the last CC connect is approximately 4.63.
  - There are 358 missing values and outliers (e.g., 47 days) that need to be addressed.
- **cashback:**
  - The cashback column has a right-skewed distribution, with majority of customers received cashback amounts between 0 and 500, with a peak around 150–200.
  - The average cashback amount is approximately 196.26.
  - There are 473 missing values and outliers (e.g., 1,997) that need to be addressed.
- **Login_device:**
  - The distribution is heavily skewed toward Mobile login devices, with fewer customers using Computer or Others.
  - There are 221 missing values that need to be addressed.

Bivariate Analysis

Churn Vs Numerical Columns



Box Plots: Churn vs Numerical Features
Churn vs. Tenure
Churn vs. CC_Contacted_LY
Churn vs. Account_user_count
Churn vs. rev_per_month
Churn vs. Complain_ly
Churn vs. rev_growth_yoy
Churn vs. coupon_used_for_payment
Churn vs. Day_Since_CC_connect
Churn vs. cashback
Tenure
CC_Contacted_LY
Account_user_count
rev_per_month
Complain_ly
rev_growth_yoy
coupon_used_for_payment
Day_Since_CC_connect
cashback
Churn
0
1

Observations:

- **Churn vs Tenure:**
  - The median tenure for churned customers is likely lower than for non-churned customers. This suggests that customers with shorter tenures are more likely to churn.
  - Outliers indicate that some customers with unusually long tenures still churned. These cases should be investigated further.
- **CC_Contacted_LY:**
  - The median number of contacts for churned customers is slightly higher than for non-churned customers. This suggests that customers who were contacted more frequently are more likely to churn.
- **Account_user_count:**
  - The median account user count for churned customers is higher. This suggests that accounts with more than 3 users are more likely to churn.
- **rev_per_month:**
  - The median revenue per month is similar for both churned and non-churned customers. This suggests that revenue per month alone may not be a strong predictor of churn.
- **rev_growth_yoy:**
  - The median revenue growth is slightly lower for churned customers compared to non-churned customers. This suggests that accounts with lower revenue growth may be more likely to churn.
  - Outliers are present in both groups, but churned customers have more extreme outliers in the higher range of revenue growth.
  - Most customers, whether they churn or not, have low revenue per month.
- **coupon_user_for_payment:**
  - The median coupon usage is almost the same for both churned and non-churned customers. This suggests that coupon usage alone does not have a strong impact on churn.
  - A majority of customers use very few coupons, indicating that heavy coupon usage is not common.
- **Day_Since_CC_connect:**
  - The IQR for both groups is fairly narrow, meaning that most customers contact customer care frequently.
  - Churned customers tend to have slightly fewer recent interactions, meaning they might be disengaged or have unresolved issues.
  - There are a few extreme cases where some customers have not contacted customer care for over 30+ days.
- **cashback:**
  - The median cashback for both churned and non-churned customers is quite similar.
  - Both groups have a narrow IQR, meaning most cashback values fall within a similar range.
  - The presence of high cashback values does not prevent churn, meaning that some users still leave despite receiving large rewards.

Outliers:

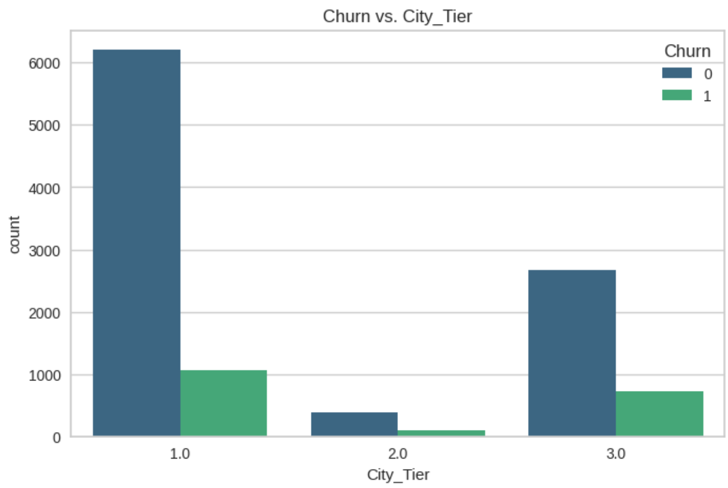
Clearly, there are outliers in the following columns:

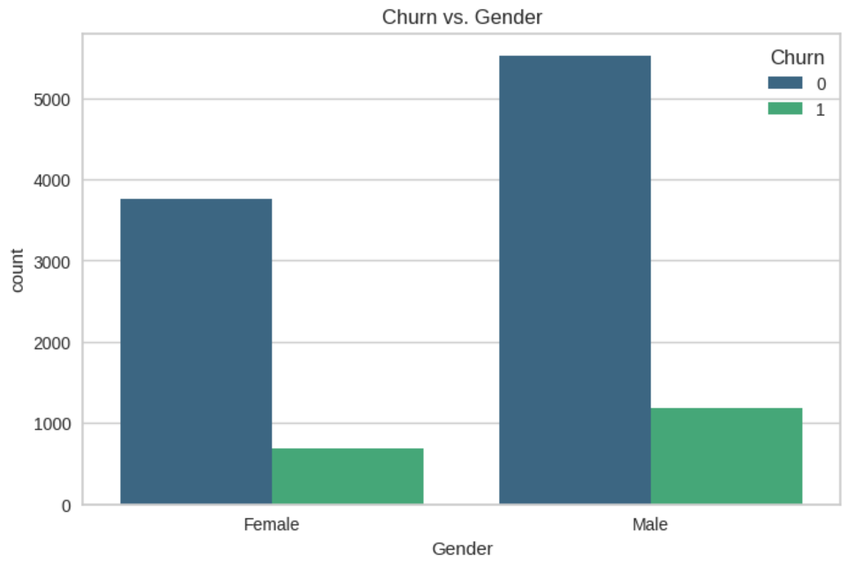
- Tenure
- CC_Contacted_LY
- Account_user_count
- rev_per_month
- rev_growth_yoy
- coupon_user_for_payment
- Day_Since_CC_connect
- cashback

However, these outliers are reasonable or valid. Therefore, we will **not** treat outliers in these columns. Instead, we will standardize the columns to adjust for deviations.

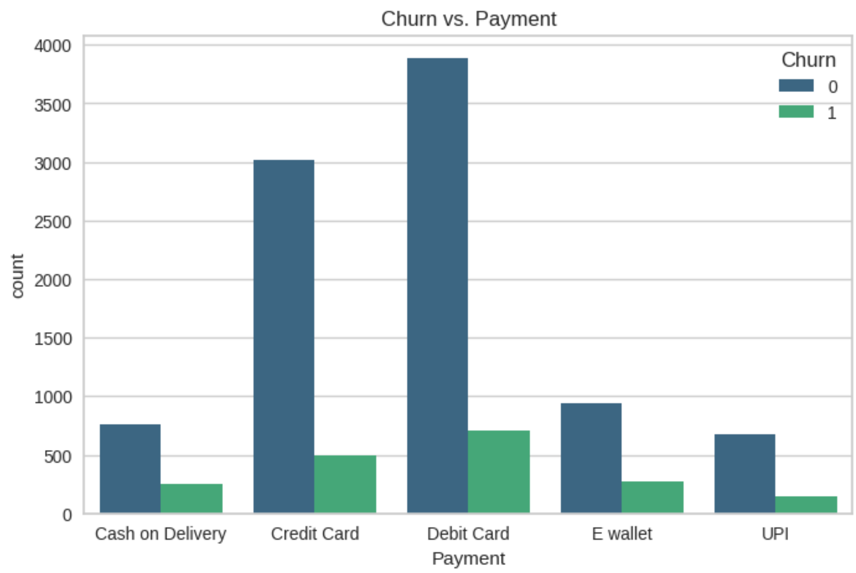
For **Account_user_count** and **Complain_ly**, these are discrete values with low levels. We will **not** treat outliers for these columns and will handle them as they are.

Churn Vs Categorical Columns

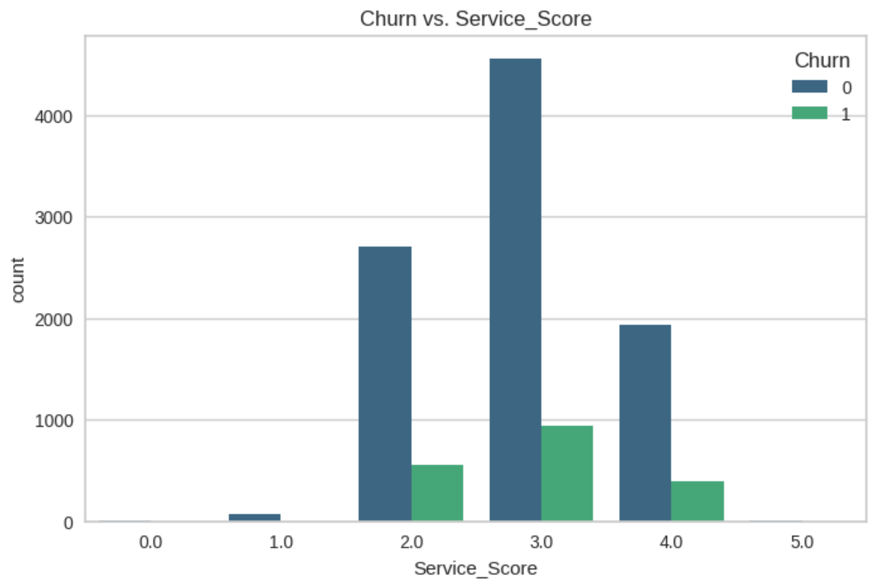




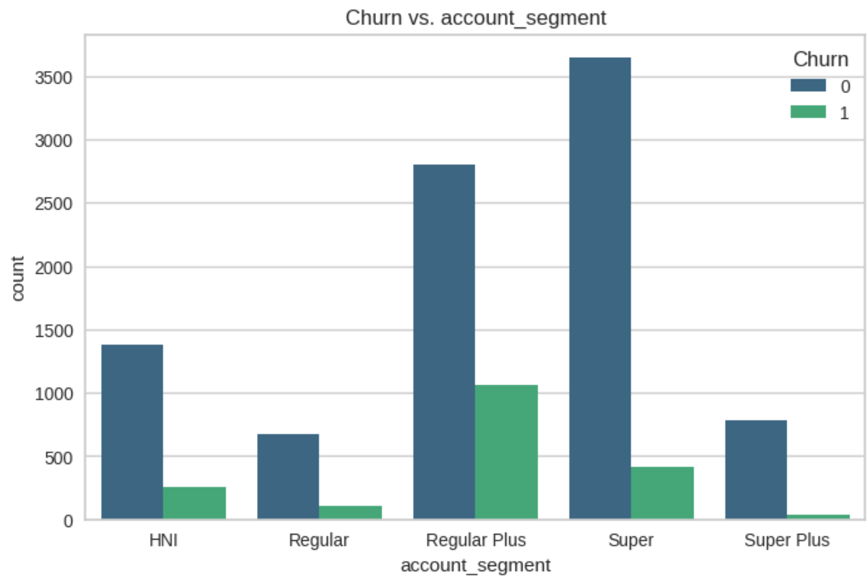
Churn vs. Gender
Churn
0
1
count
5000
4000
3000
2000
1000
0
Female
Male
Gender

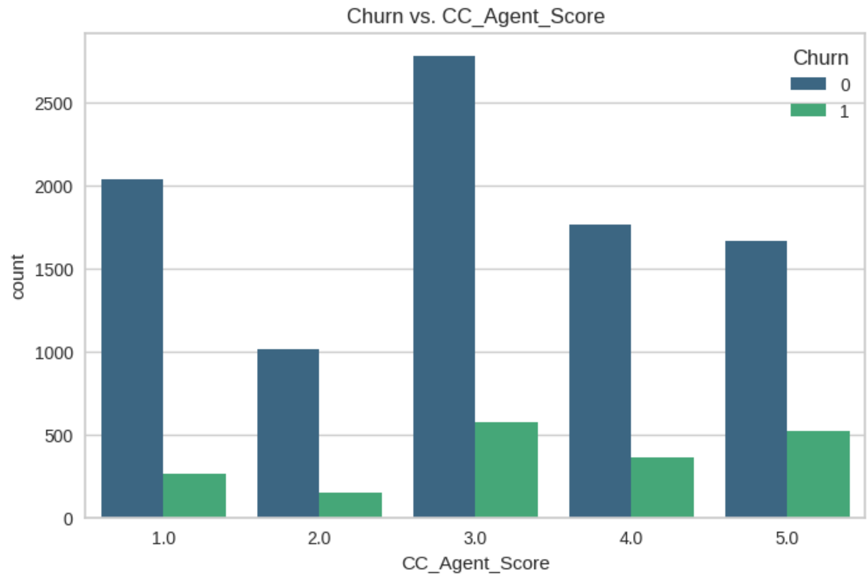


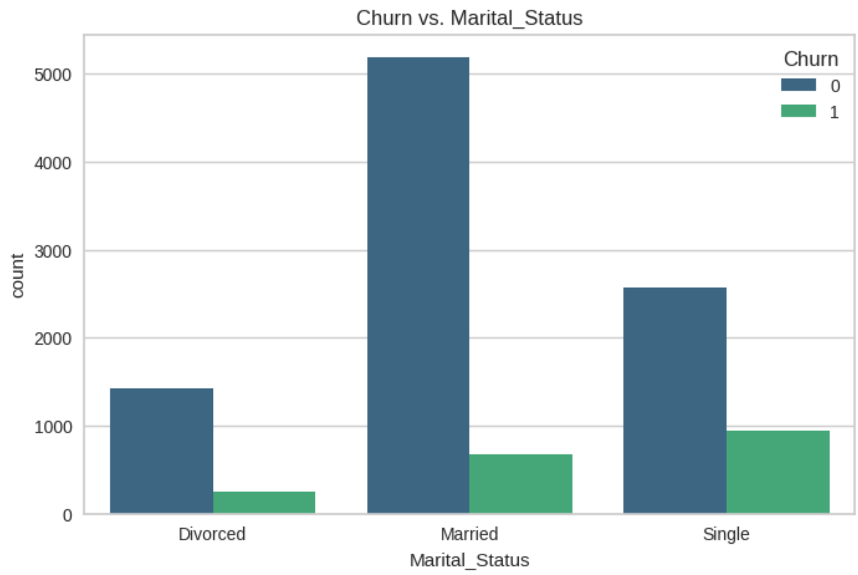
Churn vs. Payment
Churn
0
1
count
4000
3500
3000
2500
2000
1500
1000
500
0
Cash on Delivery
Credit Card
Debit Card
E wallet
UPI
Payment

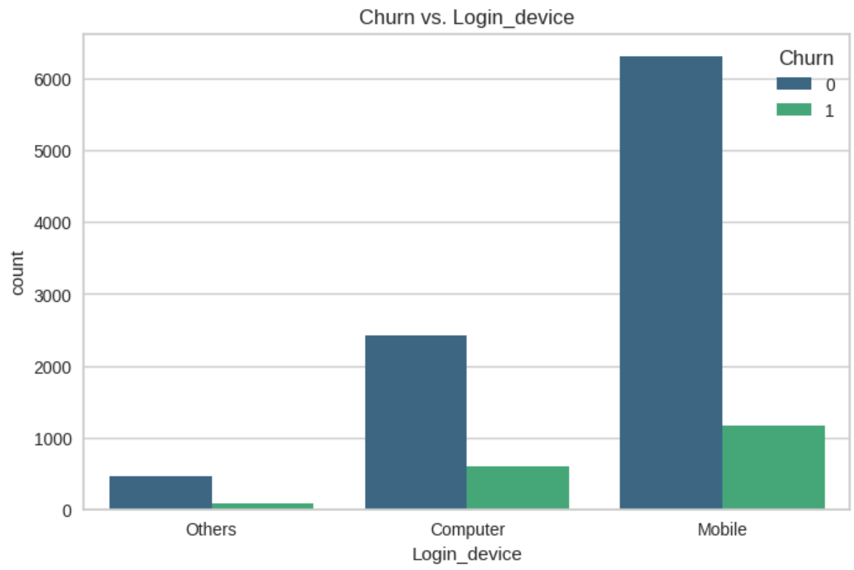


Churn vs. Service_Score
Churn
0
1
count
4000
3000
2000
1000
0
0.0
1.0
2.0
3.0
4.0
5.0
Service_Score


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


Churn vs. CC_Agent_Score
Churn
0
1
count
2500
2000
1500
1000
500
0
1.0
2.0
3.0
4.0
5.0
CC_Agent_Score


Churn vs. Marital_Status
Churn
0
1
count
5000
4000
3000
2000
1000
0
Divorced
Married
Single
Marital_Status

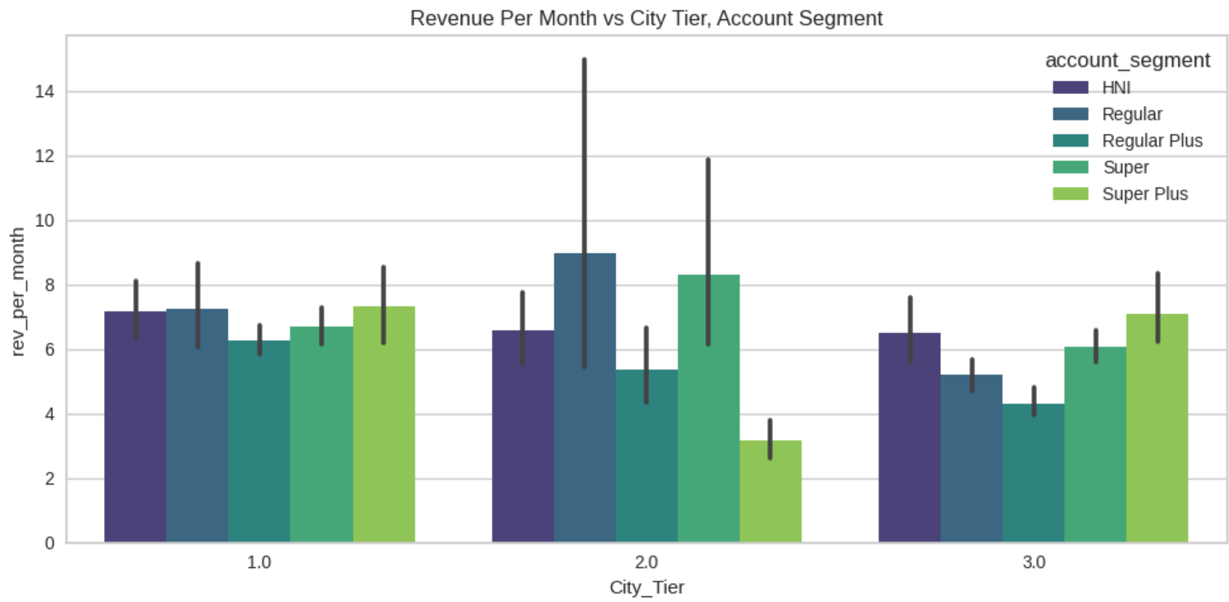


Observations:

- **Churn vs City_Tier:**
  - The majority of customers are from **City Tier 1**, followed by **City Tier 3**, and the least from **City Tier 2**.
  - **City Tier 1** has the highest churn numbers, but the churned proportion is relatively low compared to its total population.
  - The churn proportion appears relatively higher in **City Tier 3** compared to **City Tier 1**.
  - **City Tier 2** has the least churn cases and the smallest customer base overall.
- **Churn vs Payment:**
  - The highest number of customers use **Debit Cards**, followed by **Credit Cards**.
  - The least-used payment method appears to be **UPI**.
  - **Debit Card** users have the highest number of churners, likely because it's the most used payment method.
  - Customers who use **Cash on Delivery** and **E-Wallets** also show notable churn.
  - **UPI** shows the lowest absolute churn numbers, indicating it may be used by more loyal customers.
- **Churn vs Gender:**
  - More **male** customers churned than **female** customers.
  - The overall customer base is larger for males.
- **Churn vs Account_user_count:**
  - Accounts with **3 or 4 users** are the most common.
  - Churn is highest for accounts with **4 users**, indicating that mid-sized accounts might have a higher churn risk.
  - Very small (**1-2 users**) and very large (**6 users**) accounts show lower churn.
- **Churn vs Account_segment:**
  - **Super** and **Regular Plus** segments have the highest number of customers.

- Churn is highest in the **Regular Plus** segment, suggesting that mid-tier customers may be more likely to leave.
    - **Super Plus** customers have the lowest churn.
    - **Regular** and **Super** segments have moderate churn ratios.
- **Churn vs Marital_status:**
    - **Married** customers form the largest group, followed by **single** and then **divorced** individuals.
    - Churn is highest among **single** customers, suggesting they may be more likely to leave than married or divorced individuals.
    - **Married** customers have the lowest churn rate relative to their total population.
    - **Divorced** customers have the smallest representation in the dataset.
- **Churn vs Login_Device:**
    - Most customers use **Mobile** for login, followed by **Computer** and then **Others**.
    - Churn is highest among **Mobile** users.
    - **Computer** users have a moderate churn rate, but their overall count is lower than mobile users.
    - Customers using **Other** devices have the lowest churn.

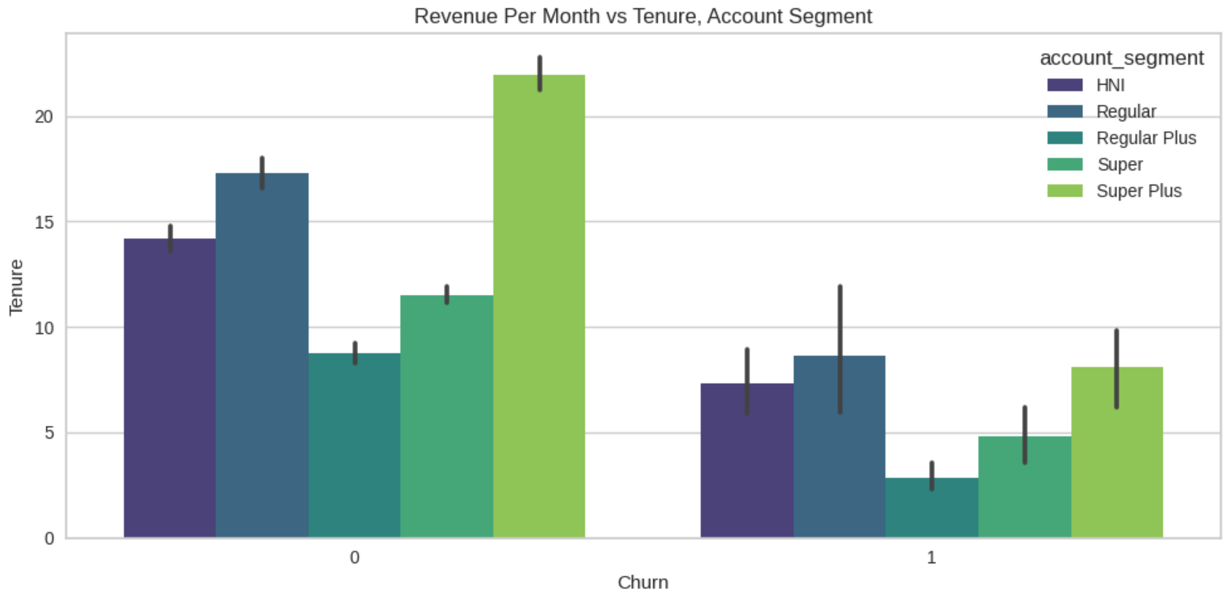
Multivariate Analysis



Observations:

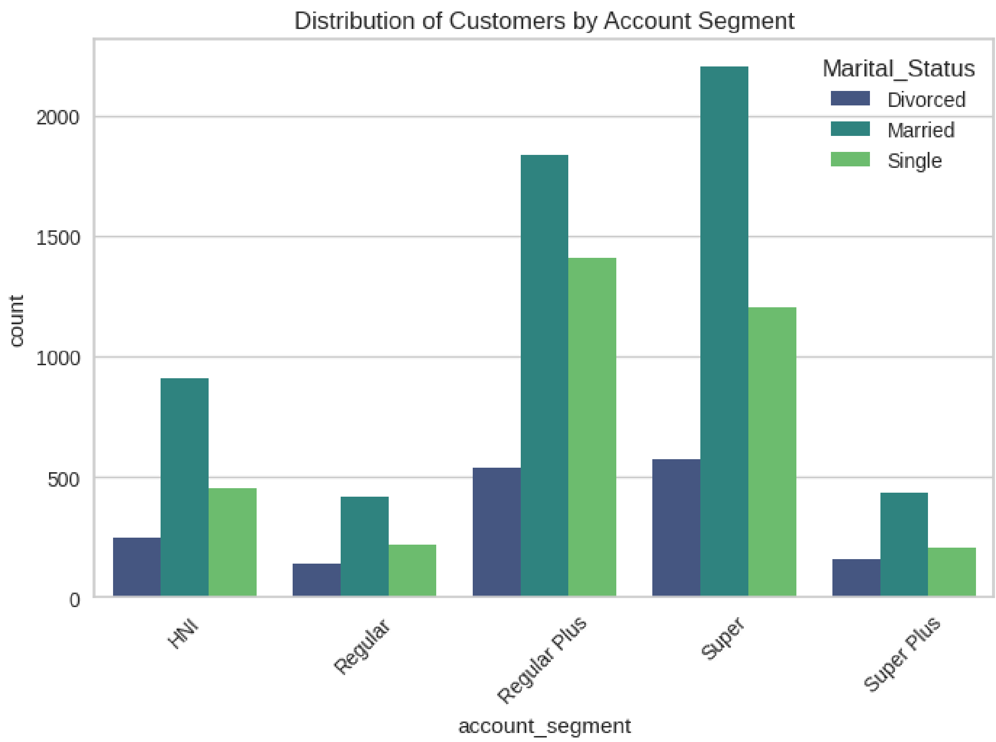
- City Tier 1 has a balanced revenue distribution across all segments.
- City Tier 2 shows extreme variations—some segments earn much higher revenue, while others earn significantly less.
- City Tier 3 has the lowest revenue overall, suggesting fewer high-revenue customers.
- City Tier 1 maintains a more stable revenue distribution.
- Regular and Super account segments tend to have higher revenue in City Tier 2.

- Super Plus segment shows a dip in City Tier 2, indicating lower revenue compared to other segments.
- HNI (High Net Worth Individuals) maintain steady revenue across tiers.



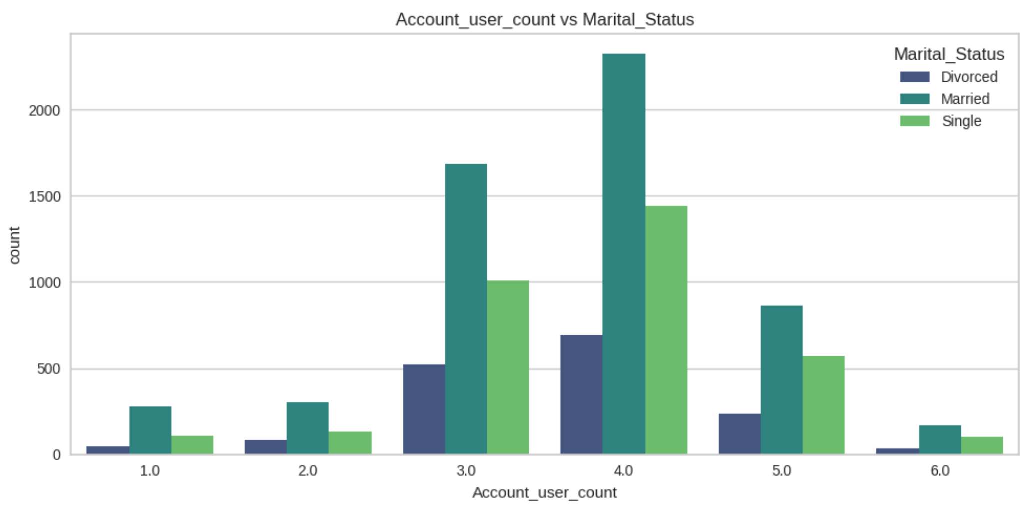
Observations:

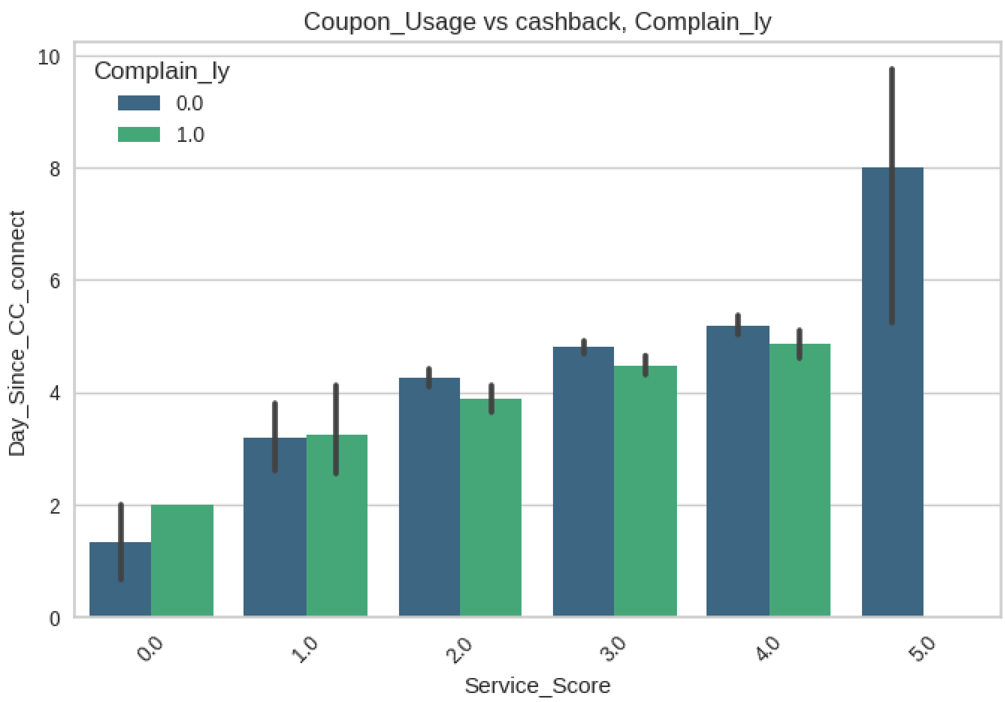
- Customers with No Churn (Churn = 0) Have Higher Tenure:
    - Across all account segments, customers who have not churned tend to have significantly higher tenure compared to those who have churned.
    - The Super Plus segment stands out with the highest tenure for non-churned customers.
- Churned Customers (Churn = 1) Have Lower Tenure:
    - For all account segments, the tenure of churned customers is noticeably lower.
    - The Regular Plus segment has the lowest tenure among churned customers.
- Super Plus and Regular Segments Have the Highest Tenure:
    - Super Plus customers have the longest tenure among non-churned customers.
    - Regular customers also show a relatively high tenure.
- Higher Churn in Segments with Lower Tenure:
    - The Regular Plus and Super segments have the lowest tenure among churned customers, indicating they may be more prone to customer churn.
    - Super Plus customers seem more loyal, given their high tenure even among non-churned customers.



Observations:

- The Super segment has the largest number of customers, followed by the Regular Plus segment.
- Across all account segments, the Married category has the highest number of customers.
- The Divorced category consistently has the lowest count across all account segments.
- The HNI (High Net-worth Individual) and Super Plus segments have comparatively fewer customers than other segments.
- The Regular and HNI segments have more balanced distributions among different marital statuses.



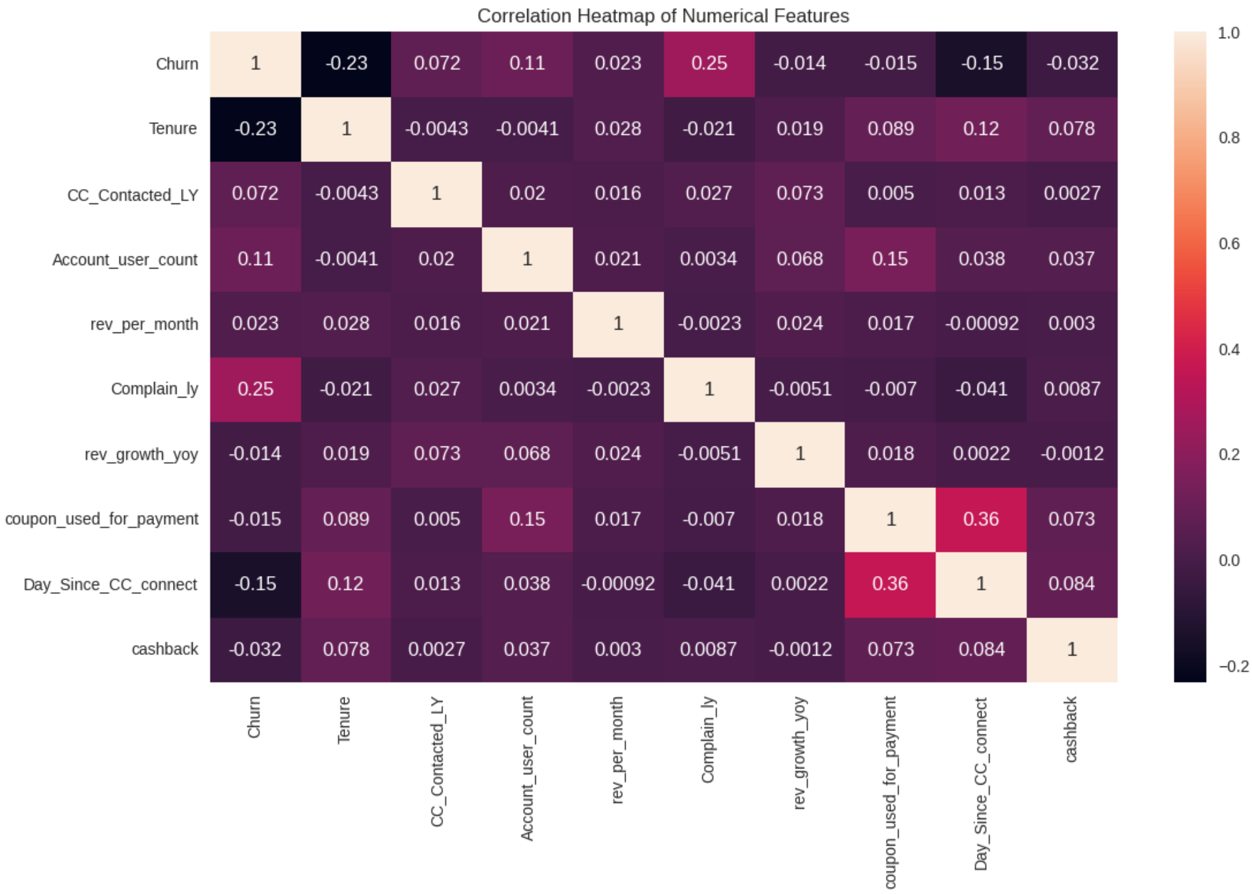


Pair Plot for Numerical Columns:

Observations:

There seems to be some clustering, indicating that customers with specific revenue levels tend to exhibit similar growth patterns between below features:

- CC_Contacted_LY vs cashback
- CC_Contacted_LY vs coupon_used_for_payment
- coupon_used_for_payment vs cashback
- rev_per_month vs rev_growth_yoy



Observations:

- **Churn Correlations:**
  - Tenure: There is a moderate negative correlation (-0.23) between tenure and churn, indicating that customers with longer tenure are less likely to churn.
  - Complain_ly: There is a positive correlation (0.25) between recent complaints and churn, suggesting that customers who have complained recently are more likely to churn.
  - Day_Since_CC_connect: This has a negative correlation (-0.15) with churn, indicating that customers who have not connected their credit card recently are more likely to churn

- **Service_Score:**
  - Account_user_count: There is a moderate positive correlation (0.32) between service score and the number of account users, indicating that accounts with more users tend to have higher service scores.
  - Coupon_used_for_payment: There is a positive correlation (0.18) between service score and the use of coupons for payment, suggesting that customers with higher service scores are more likely to use coupons.
- **coupon_used_for_payment:**
  - Account_user_count: A moderate positive correlation (0.15) suggests that accounts with more no. of users customers use coupons for payments frequently

    - Day_Since_CC_connect: There is a positive correlation (0.36) between coupon_used_for_payment and the days since the last connect with CC, suggesting that customers who use coupons frequently are less likely to have a CC connect in recent days.

- **Weak Correlations:**
    - Most numerical variables have very weak correlations with each other, indicating limited linear relationships.
    - CC Agent Score, Revenue per Month, and Growth YoY have negligible correlations with other features, meaning they don't strongly influence churn or other metrics.

Data Preprocessing (Continued)

Handling Columns with Missing Values

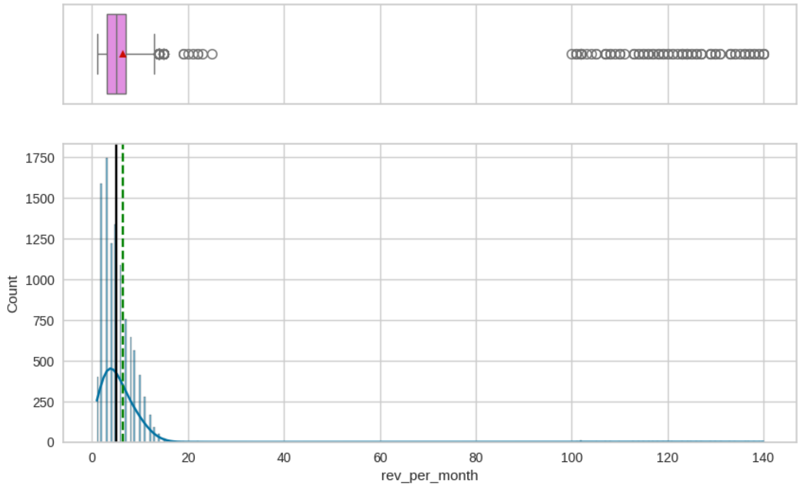
`check_missing_values in Each Columns`

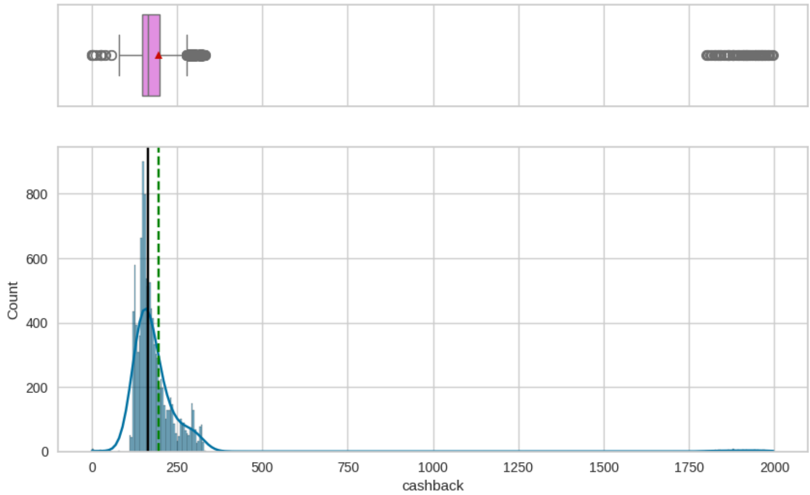
	Missing_Values	%_Missing_Values
rev_per_month	791	7.02
cashback	473	4.20
Account_user_count	444	3.94
Day_Since_CC_connect	358	3.18
Complain_ly	357	3.17
Login_device	221	1.96
Tenure	218	1.94
Marital_Status	212	1.88
CC_Agent_Score	116	1.03
City_Tier	112	0.99
Payment	109	0.97
Gender	108	0.96
CC_Contacted_LY	102	0.91
Service_Score	98	0.87
account_segment	97	0.86
rev_growth_yoy	3	0.03
coupon_used_for_payment	3	0.03
Churn	0	0.00

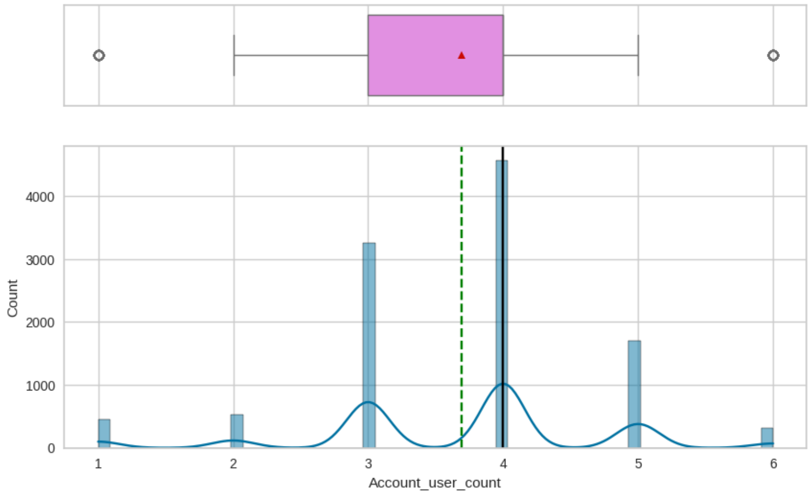
Observations:

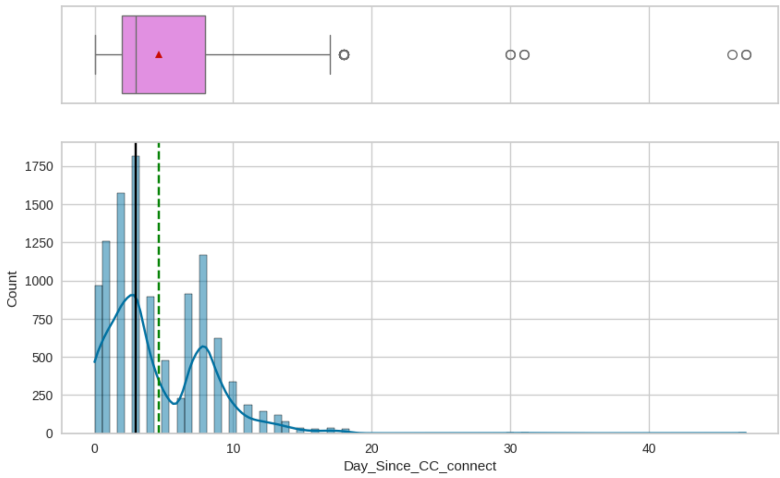
- There are several columns with missing values.
- We will treat missing values after checking their distribution

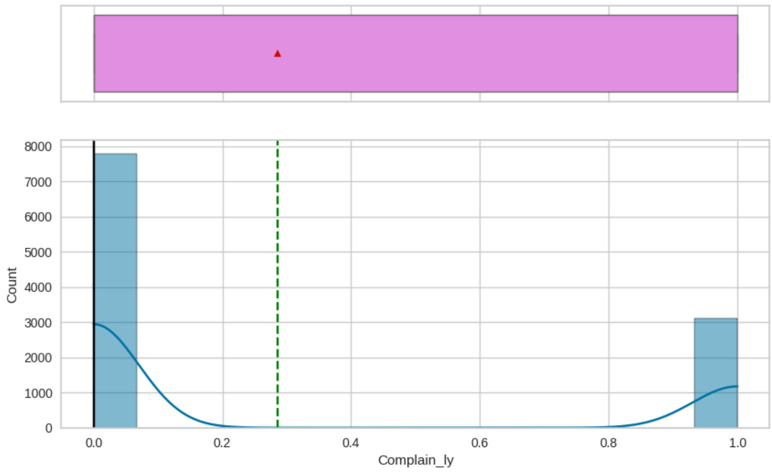
Checking the Distributions for all the Numerical Columns with missing Values

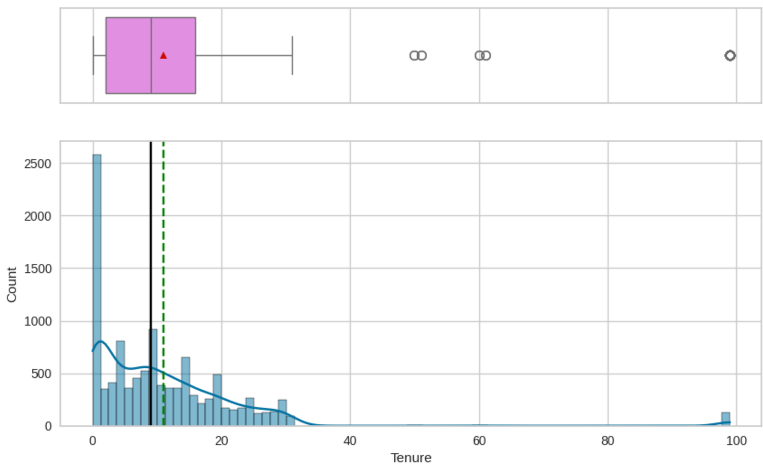


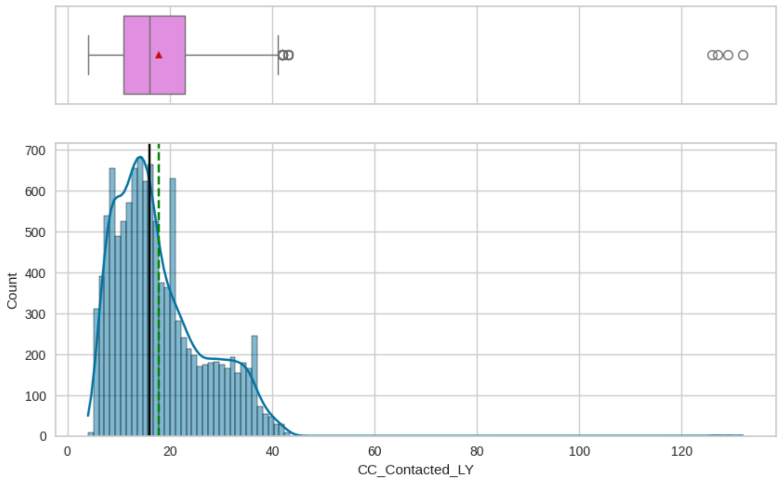


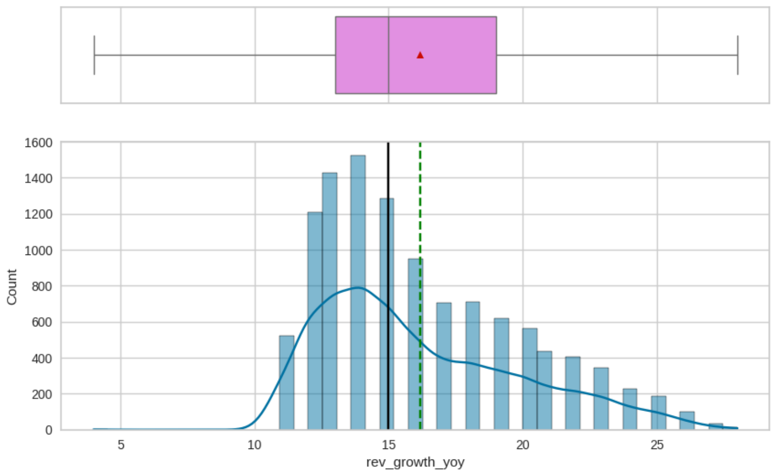


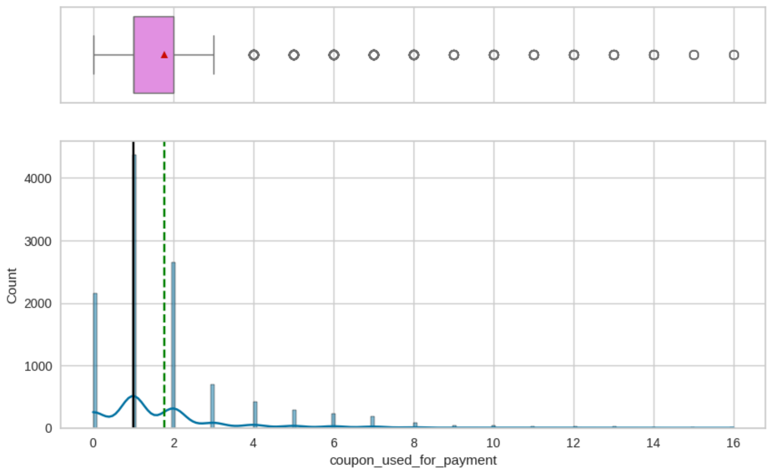

Count
Day_Since_CC_connect
0
10
20
30
40
0
250
500
750
1000
1250
1500
1750


Count
Complain_ly
0.0
0.2
0.4
0.6
0.8
1.0
0
1000
2000
3000
4000
5000
6000
7000
8000


Count
Tenure
0
20
40
60
80
100
0
500
1000
1500
2000
2500







Observations:

- **rev_per_month, cashback, Day_Since_CC_connect, Tenure, CC_Contacted_LY:** distributions are skewed and having outliers. We will impute the missing values in these numerical columns with median value.
- **Service_Score, CC_Agent_Score, City_Tier, Complain_ly, Marital_Status, Payment, Login_device, Gender, account_segment:** are all ategorical/ordinal in nature, missing values can be replaced with the most common values. We will impute them with mode value.

- **coupon_used_for_payment:** Since the data is heavily skewed, using the median (instead of mean) will prevent bias due to outliers.
- **rev_growth_yoy:** Mean Imputation is Suitable in this column because the distribution is fairly normal without extreme outliers
- **Account_user_count:** Given the multimodal distribution, Mode imputation will be suitable for this categorical count-like variables.

Values post Missing Value Treatment:

	Missing_Values	%_Missing_Values
Churn	0	0.0
Tenure	0	0.0
cashback	0	0.0
Day_Since_CC_connect	0	0.0
coupon_used_for_payment	0	0.0
rev_growth_yoy	0	0.0
Complain_ly	0	0.0
rev_per_month	0	0.0
Marital_Status	0	0.0
CC_Agent_Score	0	0.0
account_segment	0	0.0
Account_user_count	0	0.0
Service_Score	0	0.0
Gender	0	0.0
Payment	0	0.0
CC_Contacted_LY	0	0.0
City_Tier	0	0.0
Login_device	0	0.0

Train-Test Split

We will be doing the Train-Test split before going to Outlier treatment.

If you clean and preprocess the entire dataset before splitting:

- You risk data leakage because the test set information is used during training.
- Your model's performance metrics may be overly optimistic and not reflective of real-world performance.

Train Set Sample :

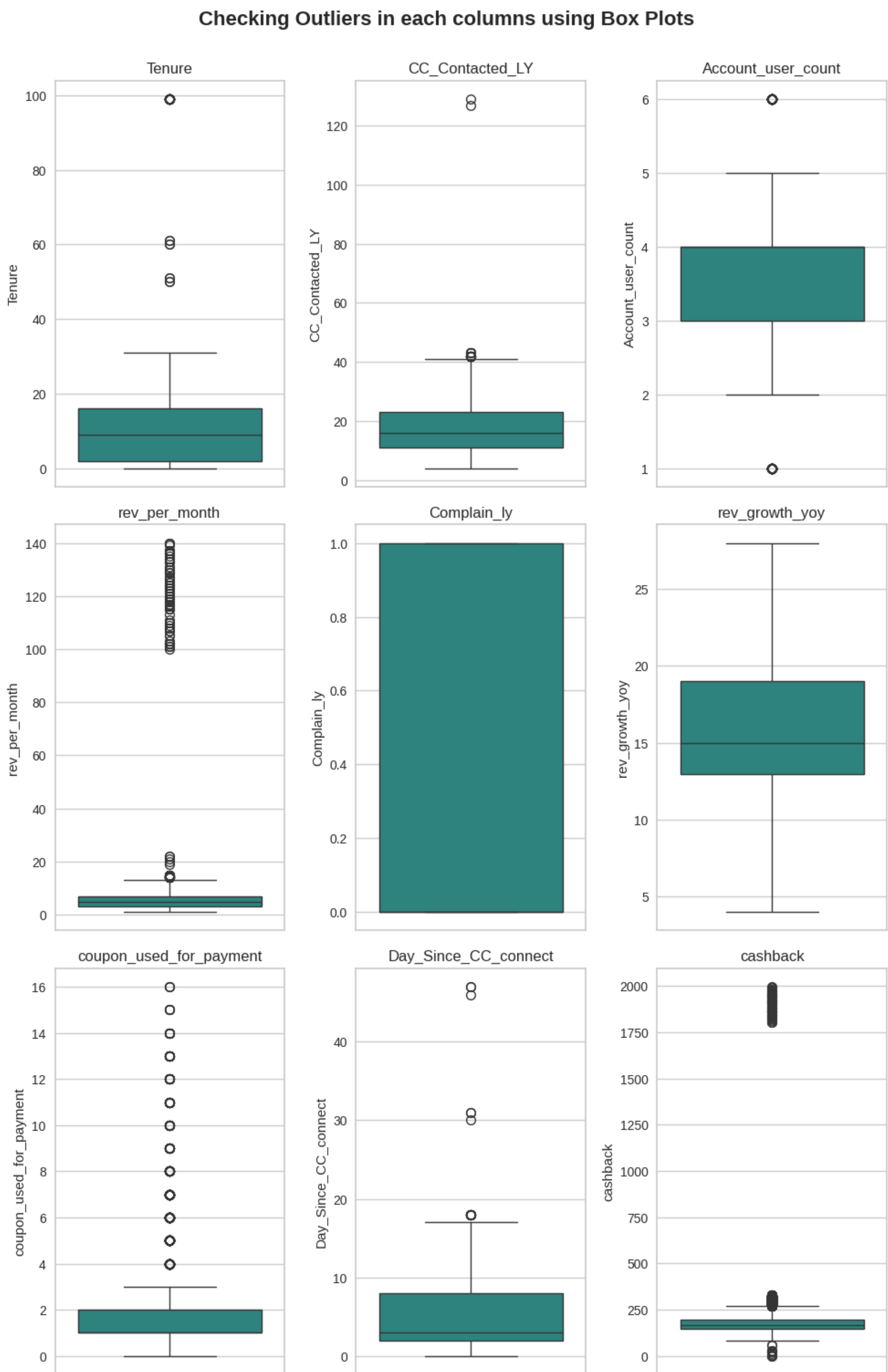
	Tenure	City_Tier	CC_Contacted_LY	Payment	Gender	Service_Score	Account_user_count	account_segment	CC_Agent_Score	Marital_Status	rev_per_month	Complain_ly
6135	0.0	1.0	8.0	Debit Card	Male	2.0	4.0	Super	2.0	Divorced	10.0	0.0
8088	5.0	1.0	9.0	Cash on Delivery	Male	3.0	3.0	Super	5.0	Married	9.0	0.0
1313	0.0	3.0	15.0	E wallet	Male	2.0	3.0	Regular Plus	5.0	Married	2.0	0.0
10426	1.0	1.0	38.0	Debit Card	Female	4.0	4.0	Regular Plus	4.0	Married	3.0	1.0
10924	9.0	1.0	16.0	Debit Card	Male	4.0	5.0	Regular Plus	1.0	Married	6.0	0.0

Train Set Sample :

	Tenure	City_Tier	CC_Contacted_LY	Payment	Gender	Service_Score	Account_user_count	account_segment	CC_Agent_Score	Marital_Status	rev_per_month	Co
6926	20.0	1.0	12.0	Credit Card	Female	3.0	3.0	HNI	4.0	Married	3.0	
1669	3.0	1.0	34.0	Credit Card	Female	3.0	3.0	Super	1.0	Married	2.0	
9498	1.0	3.0	12.0	Credit Card	Female	3.0	4.0	Regular Plus	3.0	Single	5.0	
3287	16.0	1.0	7.0	Debit Card	Female	3.0	5.0	Super	2.0	Divorced	11.0	
2973	29.0	1.0	27.0	Cash on Delivery	Male	3.0	4.0	Super	2.0	Divorced	5.0	

Handling Outliers

We will be performing Outlier Treatment only on Training set


Checking Outliers in each columns using Box Plots
Tenure
CC_Contacted_LY
Account_user_count
rev_per_month
Complain_ly
rev_growth_yoy
coupon_used_for_payment
Day_Since_CC_connect
cashback

```
Determining outlier values for: Tenure

Q1 = 2.0, Q3 = 16.0, 4*IQR = 56.0

Outlier values:
 [99.]


Determining outlier values for: CC_Contacted_LY

Q1 = 11.0, Q3 = 23.0, 4*IQR = 48.0

Outlier values:
 [127. 129.]


Determining outlier values for: Account_user_count

Q1 = 3.0, Q3 = 4.0, 4*IQR = 4.0

Outlier values:
 []


Determining outlier values for: rev_per_month

Q1 = 3.0, Q3 = 7.0, 4*IQR = 16.0

Outlier values:
 [ 22. 100. 101. 102. 103. 105. 107. 108. 109. 110. 111. 113. 115. 116.
 117. 118. 119. 120. 121. 122. 123. 124. 125. 126. 127. 129. 130. 131.
 133. 134. 135. 136. 137. 139. 140.]


Determining outlier values for: Complain_ly

Q1 = 0.0, Q3 = 1.0, 4*IQR = 4.0

Outlier values:
 []


Determining outlier values for: rev_growth_yoy

Q1 = 13.0, Q3 = 19.0, 4*IQR = 24.0

Outlier values:
 []
```

```
Determining outlier values for: coupon_used_for_payment

Q1 = 1.0, Q3 = 2.0, 4*IQR = 4.0

Outlier values:
 [ 6.  7.  8.  9. 10. 11. 12. 13. 14. 15. 16.]


Determining outlier values for: Day_Since_CC_connect

Q1 = 2.0, Q3 = 8.0, 4*IQR = 24.0

Outlier values:
 [30. 31. 46. 47.]


Determining outlier values for: cashback

Q1 = 147.93, Q3 = 197.25, 4*IQR = 197.27999999999997

Outlier values:
 [1804. 1807. 1813. 1817. 1824. 1826. 1827. 1833. 1835. 1839. 1843. 1844.
 1850. 1853. 1858. 1862. 1865. 1866. 1869. 1877. 1878. 1879. 1880. 1888.
 1890. 1894. 1896. 1902. 1903. 1908. 1911. 1912. 1913. 1914. 1916. 1917.
 1919. 1921. 1923. 1925. 1928. 1929. 1931. 1937. 1941. 1943. 1944. 1945.
 1946. 1951. 1953. 1954. 1957. 1958. 1961. 1965. 1967. 1971. 1972. 1978.
 1982. 1985. 1991. 1992. 1997.]
```

Observations:

- **Tenure**
- **Treatment:** Capping at 99th percentile
- **Reason:** Extremely high tenure values (>60 months) are rare and might distort analysis.

rev_growth_yoy

- **Treatment:** Capping at 99th percentile
- **Reason:** Extreme revenue growth values (>25%) may not be realistic and should be limited.

  Day_Since_CC_Connect

- **Treatment:** Capping at 99th percentile
- **Reason:** Values above 30 days are rare cases and could skew results.

  coupon_used_for_payment

- **Treatment:** Capping at 95th percentile
- **Reason:** Excessive coupon usage (>8) is rare and should be capped.

cashback

- **Treatment:** Log transformation
- **Reason:** Cashback values above 500 are extreme, and log transformation normalizes distribution.

CC_Contacted_LY

- **Treatment:** Capping at 99th percentile, Log transformation
- **Reason:** Values >50 indicate excessive complaints, which are unlikely and may be data errors.

rev_per_month

- **Treatment:** Log transformation
- **Reason:** High revenue values (>100) might distort analysis, requiring transformation.

Service_Score

- **Treatment:** No action
- **Reason:** Scores of 0 to 5 are valid.

Account_user_count

- **Treatment:** No action
- **Reason:** As the values are valid, no modification is necessary.

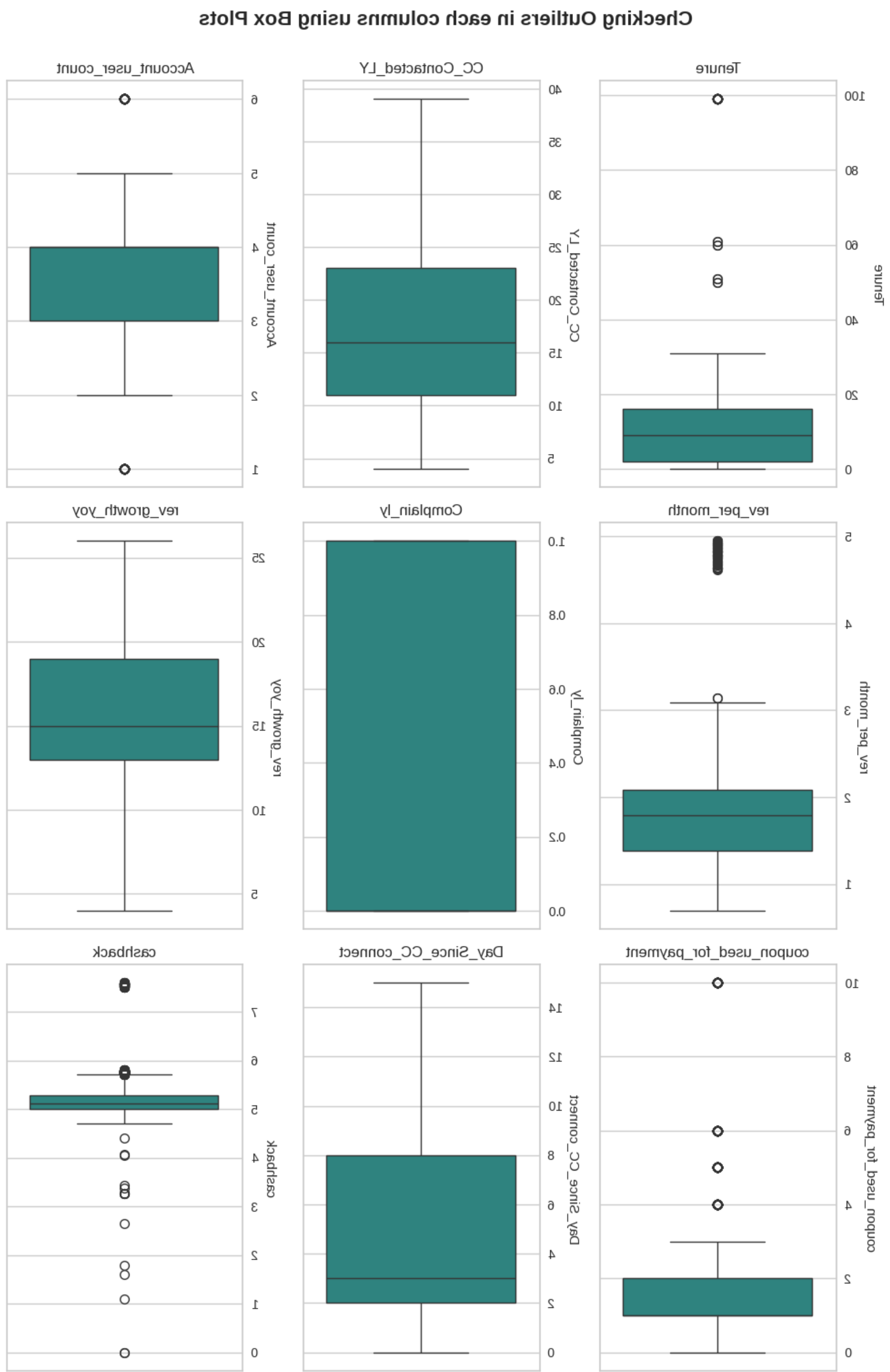
CC_Agent_Score

- **Treatment:** No action
- **Reason:** Outliers are minimal and fall within the expected range.

Complain_ly

- **Treatment:** No action
- **Reason:** This is a binary variable, so no outlier treatment is needed.

```
Treated all the outliers in the Training Set
```


Checking Outliers in each columns using Box Plots
Tenure
CC_Contacted_LY
Account_user_count
rev_per_month
Complain_ly
rev_growth_yoy
coupon_used_for_payment
Day_Since_CC_connect
cashback

Feature Engineering

Creating New Features:

Tenure_Level
New (0-3 Months)

Early (4-6 Months)

Growing (7-12 Months)

Established (1-2 Years)

Loyal (2+ Years)]

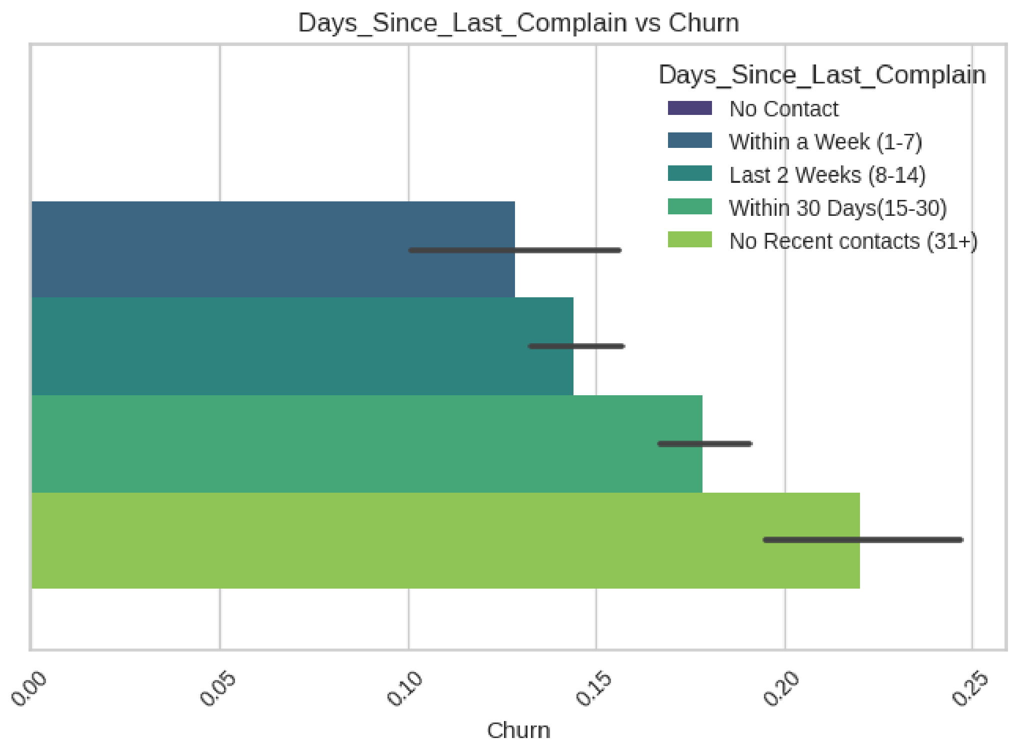
Days_Since_Last_Complain
No Contact

Within a Week (1-7)

Last 2 Weeks (8-14)

Within 30 Days(15-30)

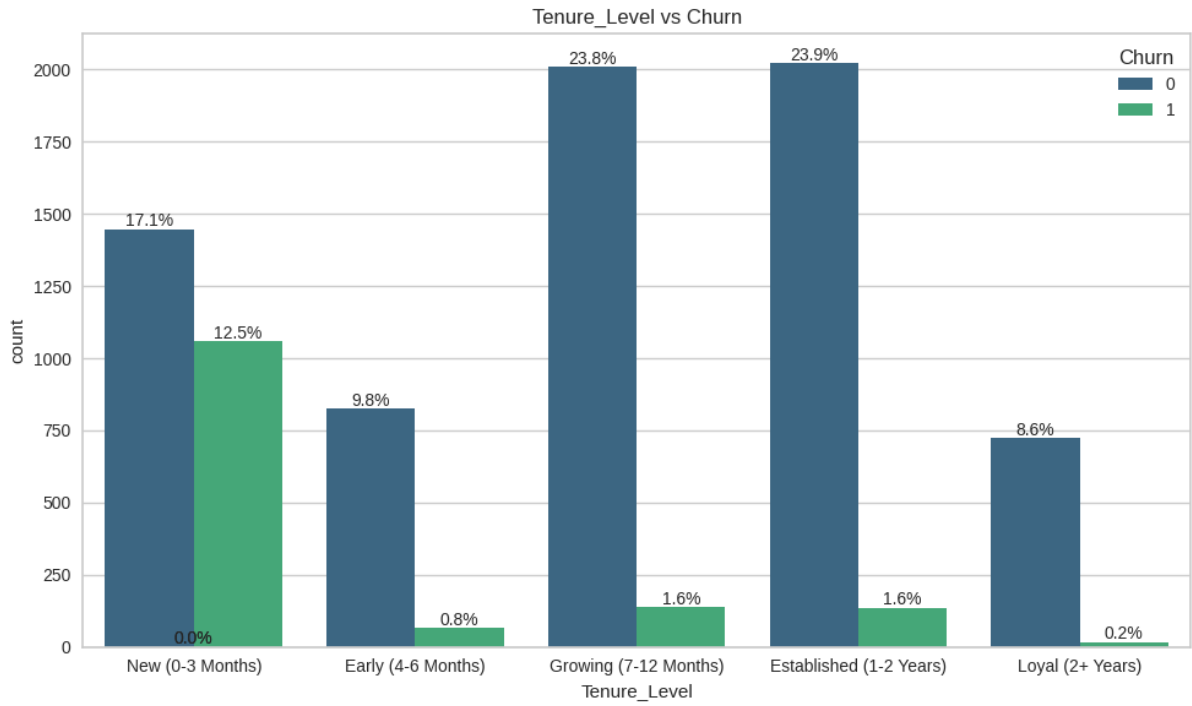
No Recent contacts (31+)

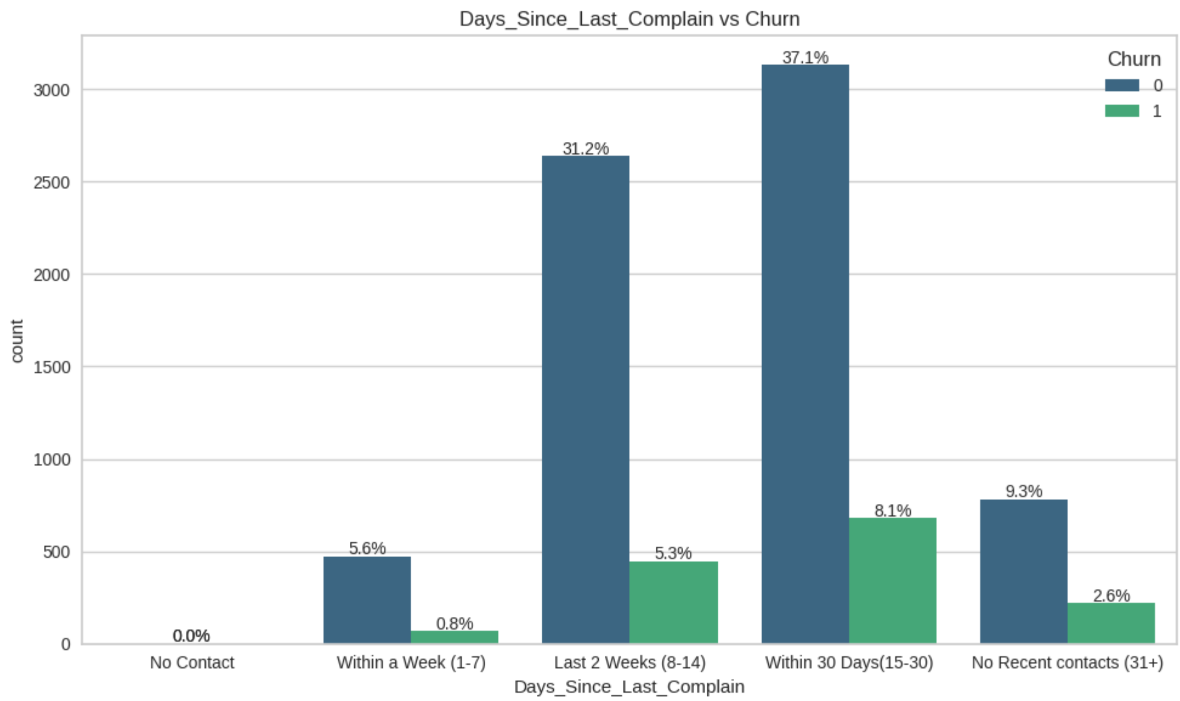


Loyalty_Score

Loyalty_Score = (Tenure * Service_Score* CC_Agent_Score)/ CC _Contacted_LY

Churn Vs New Features





Data Scaling

Scaling ensures that numerical features have similar ranges, improving model performance. We will be using below techniques:

- Standardization (Z-score Normalization)
- Min-Max Scaling

```
Scaled all the numerical columns using StandardScaler
```

Data Encoding

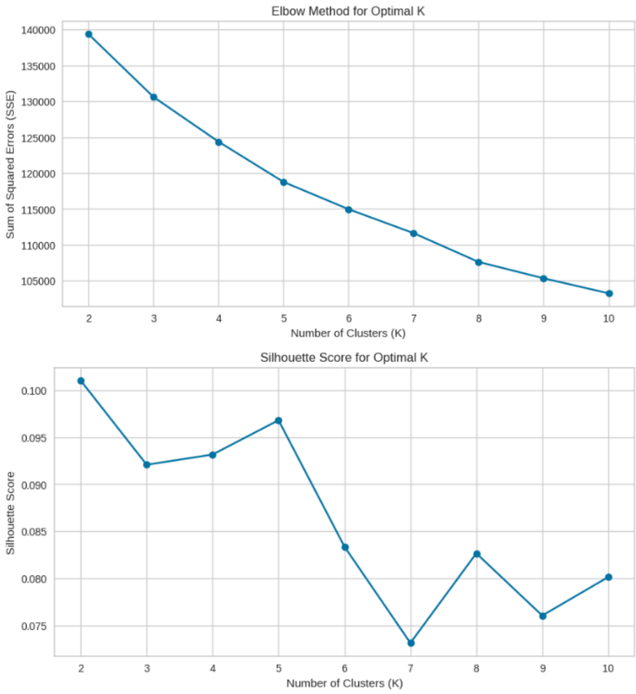
All th Categorical columns in the data needs to be converted into numerical format. We will be using the below Common techniques for the same

- One-Hot Encoding (OHE): Used for nominal categories (no order).
- Label Encoding: Used for ordinal categories (where order matters).

```
Ordinal Columns are :
['City_Tier', 'Service_Score', 'account_segment', 'CC_Agent_Score',
'Tenure_Level', 'Days_Since_Last_Complain']
Nominal Columns are :
['Payment', 'Gender', 'Marital_Status', 'Login_device']

Encoded all the Ordinal features using Label Encoder
Encoded all the Nominal features using One Hot Encoder
```

Clustering

Checking Elbow Plot




```
Optimal number of clusters (K) based on Silhouette Score: 2
```

Obeservations:

- From the Elbow Method plot, we observe that the rate of decrease in SSE slows down significantly around K = 5, indicating the optimal number of clusters.
- From the Silhouette Score plot, the highest score is observed at K = 2, but a reasonable choice considering the balance between compactness and separation is K = 5.

Based on the combination of both metrics, K = 5 appears to be the best choice for clustering the treated X_train dataset.

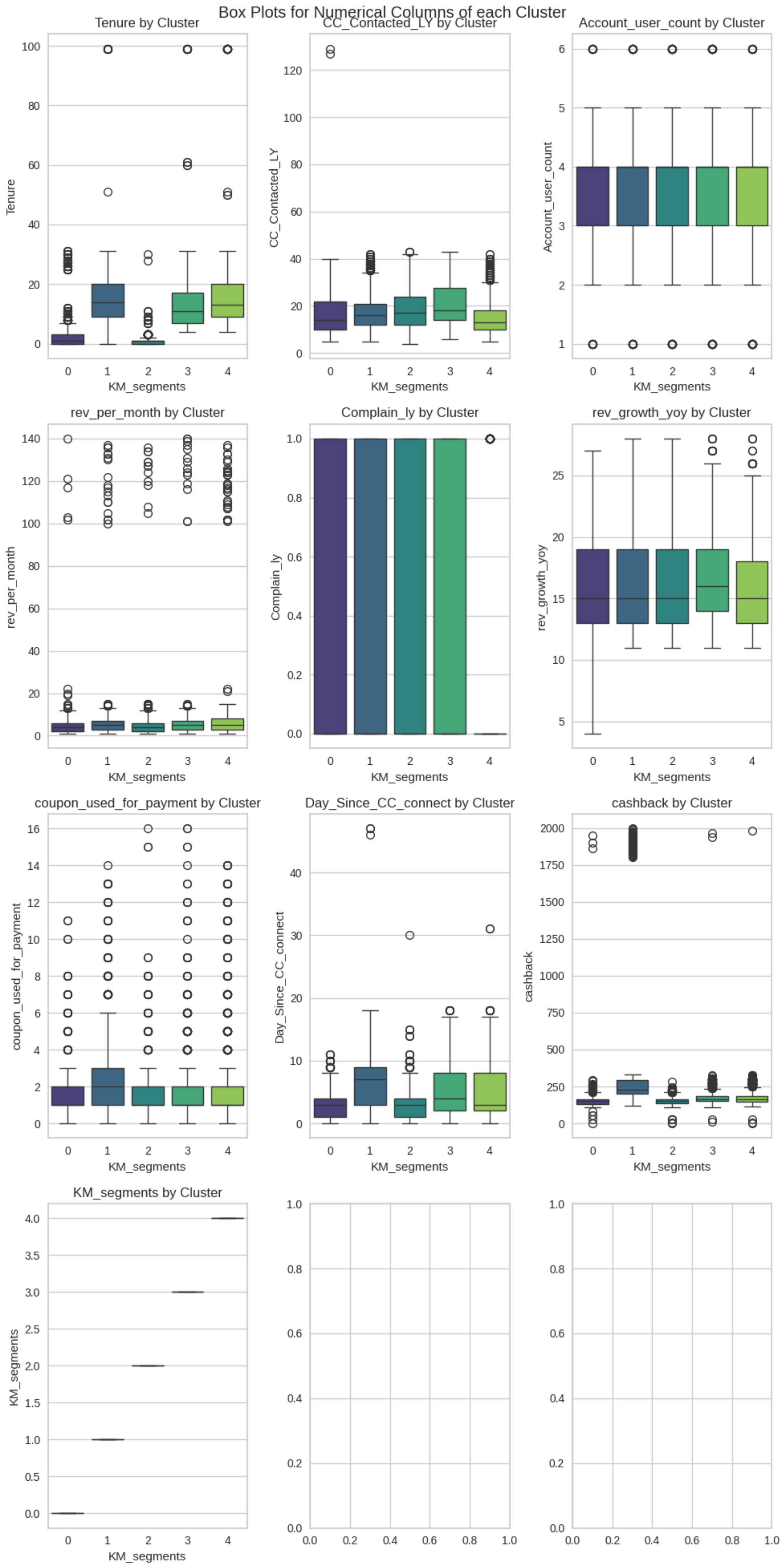
Cluster profiling

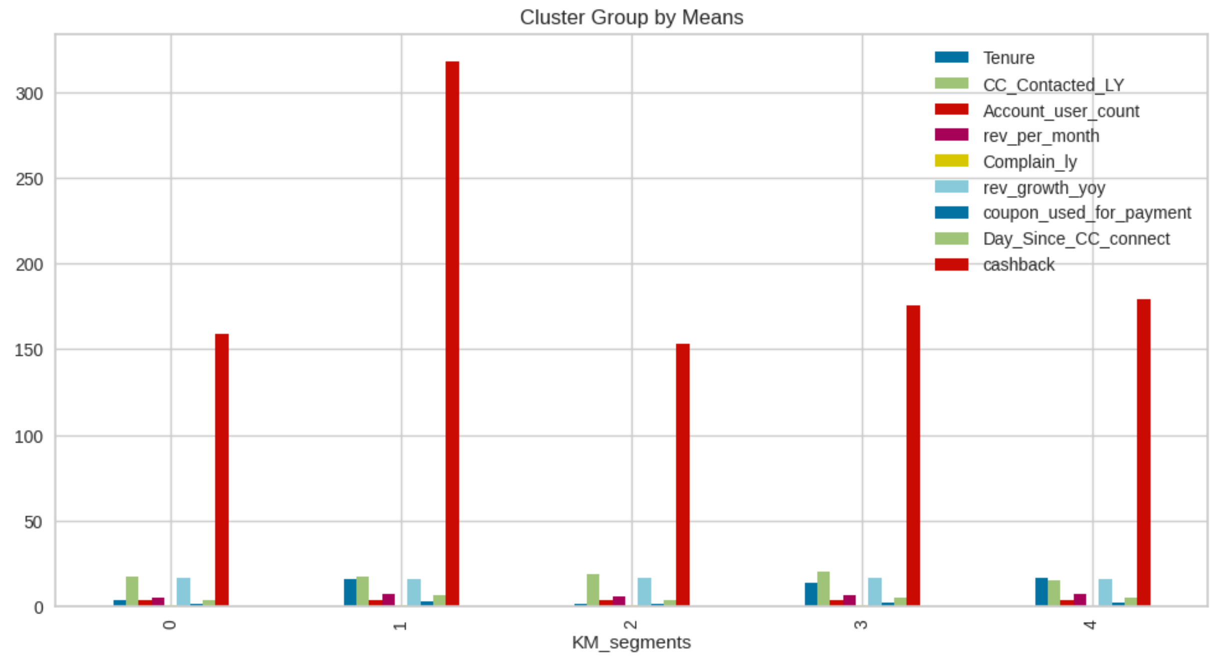
Cluster Profiling for Numerical Columns:

KM_segments	Tenure	CC_Contacted_LY	Account_user_count	rev_per_month	Complain_ly	rev_growth_yoy	coupon_used_for_payment	Day_Since_CC_connect	cashback	KM_segments
0	3.7445	17.3119	3.7399	5.0575	0.3337	16.4154	1.4246	3.1565	158.9619	0
1	15.6007	17.423	3.8236	6.8473	0.2757	16.0939	2.5372	6.5345	318.1128	1
2	1.0749	18.7058	3.75	5.3428	0.307	16.2086	1.2673	3.2299	153.4199	2
3	13.4088	20.4064	3.6836	6.0526	0.2686	16.6045	1.7215	4.8764	175.8307	3
4	16.7792	15.1479	3.5858	7.2868	0.2226	15.854	1.9832	4.777	179.1075	4

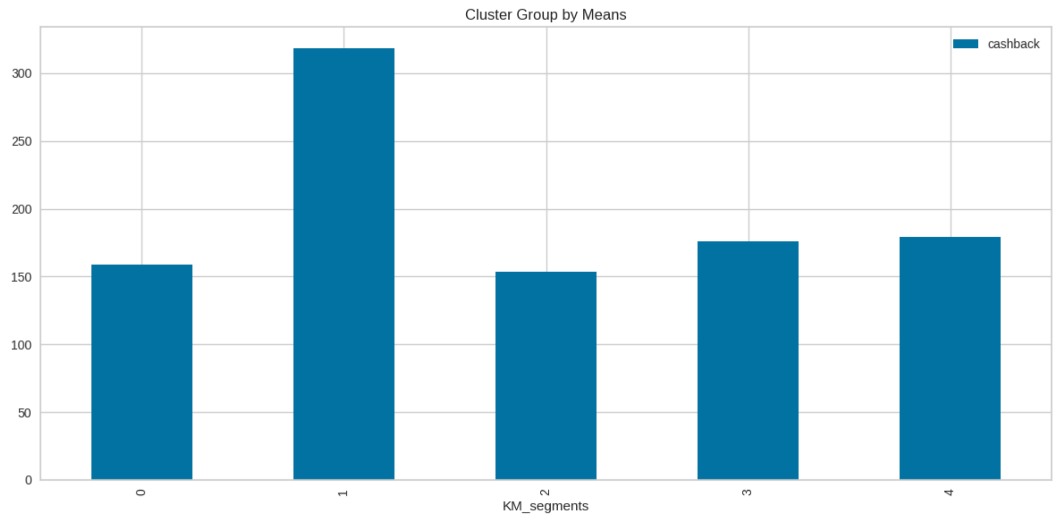
Cluster Profiling for Categorical Columns:

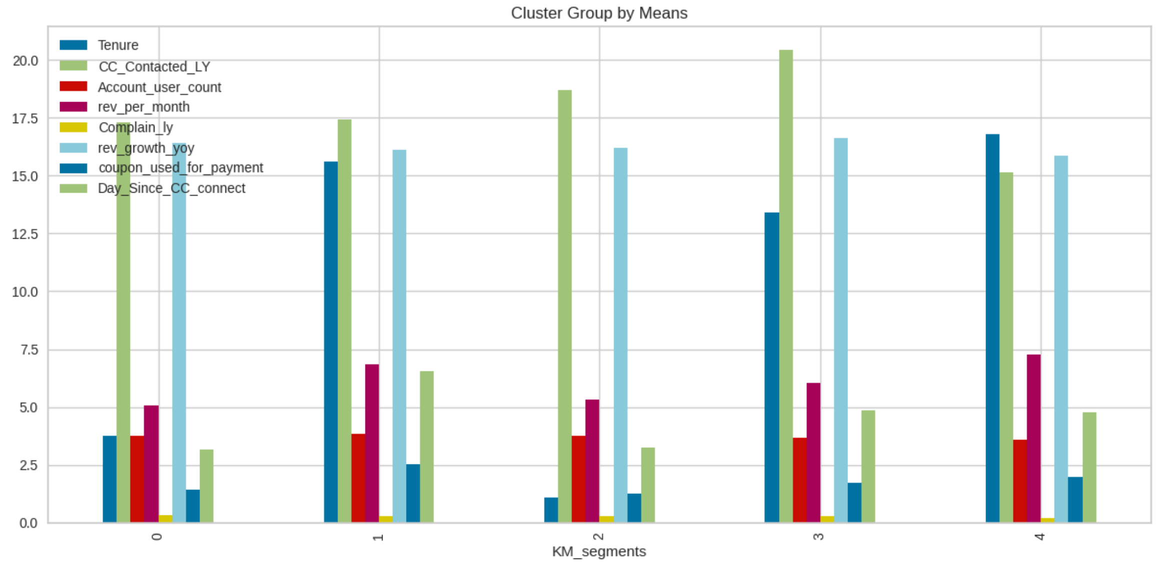
	City_Tier			Payment					Gender		Service_Score						account_segment					CC_Agent_Score					Marital_Status			Login_device		
KM_segments	1	2	3	Cash on Delivery	Credit Card	Debit Card	E wallet	UPI	Female	Male	0	1	2	3	4	5	HNI	Regular	Regular Plus	Super	Super Plus	1	2	3	4	5	Divorced	Married	Single	Computer	Mobile	Others
0	588	40	241	74	263	393	73	66	267	602	0	6	305	415	143	0	59	48	534	211	17	585	280	4	0	0	114	406	349	281	553	35
1	950	56	474	142	461	586	212	79	569	911	2	11	413	718	335	1	1041	374	29	31	5	282	146	517	308	227	239	854	387	343	1064	73
2	###	84	572	192	608	651	170	167	686	1102	0	7	533	904	342	2	124	69	1034	557	4	0	0	742	455	591	243	842	703	421	1283	84
3	###	57	659	167	605	1002	217	120	858	1253	2	15	585	###	445	2	2	28	630	1195	256	849	426	807	29	0	311	1208	592	667	1347	97
4	###	115	601	184	730	891	233	159	950	1247	2	22	611	###	488	0	6	69	663	1126	333	4	17	568	804	804	353	1259	585	589	1505	103


Box Plots for Numerical Columns of each Cluster
Tenure by Cluster
Tenure
KM_segments
CC_Contacted_LY by Cluster
CC_Contacted_LY
KM_segments
Account_user_count by Cluster
Account_user_count
KM_segments
rev_per_month by Cluster
rev_per_month
KM_segments
Complain_ly by Cluster
Complain_ly
KM_segments
rev_growth_yoy by Cluster
rev_growth_yoy
KM_segments
coupon_used_for_payment by Cluster
coupon_used_for_payment
KM_segments
Day_Since_CC_connect by Cluster
Day_Since_CC_connect
KM_segments
cashback by Cluster
cashback
KM_segments
KM_segments by Cluster
KM_segments
KM_segments



The values in cashback are significantly larger compared to those in other columns, so we're splitting the bar plot for better feature analysis.





Cluster Summary

Based on the provided cluster profiling for both numerical and categorical variables, the following insights can be drawn:

Cluster 0

- **Characteristics:**
  - Lowest **tenure (3.7 months)** and relatively high **CC contacted (17.3 times in LY)**.
  - Moderate **revenue per month (5.05)** and **revenue growth (16.4%)**.
  - High **complaint rate (33%)**.
  - Low engagement with **coupon payments (1.42)** and **cashback (158.9)**.
- **Customer Type:**
  - **New customers with high complaints and low loyalty**.
  - More likely to use **cash on delivery and debit cards**.
  - **Lower account segment representation (HNI & Super Plus)**.
  - Least representation in **higher CC agent scores**.

Cluster 1

- **Characteristics:**
  - High **tenure (15.6 months)** with moderate **CC contacts (17.4)**.
  - Moderate **revenue per month (6.84)** and **complaint rate (27.5%)**.
  - Higher **coupon usage (2.53)** and **cashback received (318.1)**.
- **Customer Type:**
  - **Loyal customers with stable revenue and engagement**.
  - More likely to use **credit cards and UPI payments**.
  - Good representation across **account segments**.
  - Higher CC agent score distribution.

Cluster 2

- **Characteristics:**
  - **Very low tenure (1.07 months)** and high **CC contacted (18.7 times in LY)**.
  - **Lower revenue per month (5.34)** and **moderate complaint rate (30.7%)**.
  - **Lowest coupon usage (1.26)** and **low cashback (153.4)**.
- **Customer Type:**
  - **New and dissatisfied customers with low spending and engagement**.
  - More likely to use **debit cards and cash on delivery**.
  - Less representation in premium account segments.
  - Low service score and CC agent score.

Cluster 3

- **Characteristics:**
  - **Moderate tenure (13.4 months)** and **highest CC contacted (20.4 times in LY)**.
  - **Moderate revenue per month (6.05)** and **complaint rate (26.8%)**.
  - **Moderate coupon usage (1.72)** and **higher cashback (175.8)**.
- **Customer Type:**
  - **Loyal but demanding customers with frequent CC interactions**.
  - More likely to use **E-wallets and credit cards**.
  - High representation in **Super Plus and Regular Plus** account segments.
  - Higher CC agent scores and service scores.

Cluster 4

- **Characteristics:**
  - **Highest tenure (16.7 months)** and **lowest CC contacted (15.1 times in LY)**.
  - **Highest revenue per month (7.28)** and **lowest complaint rate (22.2%)**.
  - **Lowest coupon usage (1.98)** but **highest cashback (179.1)**.
- **Customer Type:**
  - **Most loyal and high-value customers with low complaints**.
  - More likely to use **credit cards and mobile banking**.
  - Highest representation in **HNI and Super Plus** accounts.
  - Higher service scores and CC agent scores.

Overall Summary:

- **Cluster 2 and 0 represent new, dissatisfied, or low-value customers.**
- **Cluster 1 and 3 are engaged but demand more support (high CC contacts).**
- **Cluster 4 represents the best and most loyal customers.**

Business Recommendations and Actionable items:

1. Customer Segmentation and Personalization

- **Implement pre-defined customer segmentation based on needs, usage patterns, and spending behavior (e.g., deal seekers, tariff optimizers, etc.).**
- **Develop tailored acquisition strategies for each customer segment to maximize engagement and retention.**
- **Use customized email responses for priority customers to enhance interaction and satisfaction.**

2. Customer Acquisition Strategies

- **Launch referral programs to incentivize existing customers to bring in new ones.**
- **Collaborate with lifestyle vendors to offer vouchers and discounts to both new and loyal customers.**
- **Increase visibility and marketing efforts in Tier-2 cities to expand the customer base.**

3. Enhancing Customer Loyalty

- **Offer free cloud storage or other value-added services to loyal customers.**
- **Provide handwritten thank-you notes on invoices to create goodwill and strengthen customer relationships.**
- **Send small tokens of appreciation (e.g., gifts) on special occasions like birthdays or anniversaries.**

4. Reducing Customer Attrition

- **Analyze churn signals and triggers to proactively identify at-risk customers.**
- **Introduce subsidized offers for high-churn segments, such as single customers.**
- **Develop all-in-one family plans with extra services to make accessibility easier and more appealing.**

5. Improving Customer Experience

- **Create a specialized customer service team for top-tier customers to reduce waiting times and enhance their experience.**
- **Ensure timely resolution of all complaints and queries to maintain customer satisfaction.**
- **Conduct regular follow-ups and feedback sessions to address customer issues and improve service quality.**

6. Leveraging Payment Options

- **Promote the use of the company's e-wallet by offering discounts or cashback for transactions.**
- **Encourage hassle-free payment methods like standing instructions or UPI for convenience and safety.**

7. Customer Engagement and Feedback

- **Conduct satisfaction surveys to understand changing customer behavior and preferences.**
- **Regularly engage with customers through personalized campaigns and offers based on their profiles.**

8. Strategic Partnerships and Offers

- **Partner with other businesses to provide exclusive deals and vouchers to customers.**
- **Introduce joint loyalty programs with lifestyle brands to enhance customer retention.**

9. Focus on High-Churn Segments

- **Design targeted campaigns for high-churn groups, such as single customers or those in Tier-3 cities.**
- **Offer discounts or incentives to retain customers who show signs of disengagement.**

10. Operational Improvements

- **Ensure all customer-facing teams are trained to deliver consistent and high-quality service.**
- **Regularly update and optimize customer service processes to reduce friction and improve satisfaction.**