4/6/2025

Customer Churn Prediction

Business Report

Rejin CR
REJINCR001@GMAIL.COM

greatlearning
Power Ahead

Table of Contents

List of Figures

List of Tables

Customer Churn Prediction

Introduction

Defining the Problem Statement

The e-commerce industry is highly competitive, with businesses struggling to retain customers in a saturated market. Our client, a leading e-commerce company, faces an annual churn rate of 16.84% - meaning nearly 1 in 5 customers disengages each year, resulting in an estimated revenue loss of $1.463 billion. Unlike individual customer churn, this company deals with account-level churn, where a single account (potentially representing multiple users) stops transactions, amplifying the financial impact.

The primary challenge is proactively identifying at-risk accounts before they churn, allowing the company to deploy retention strategies efficiently. Without predictive insights, the business relies on reactive measures, leading to higher customer acquisition costs and declining profitability.

Need for the Study

Customer retention is 5 to 6 times cheaper than acquiring new customers
Additionally:

- Success rates for selling to existing customers are 60-70%, compared to just 5-20% for new prospects.
- A 5% increase in retention can boost profits by 25-95%.
- U.S. companies lose $136.8 billion annually due to preventable churn.

This project goes beyond predicting churn—it quantifies the financial impact and provides actionable strategies to:

1. **Reduce customer attrition** by targeting high-risk accounts.
2. **Optimize retention budgets** by focusing on cost-effective interventions.
3. **Enhance customer lifetime value (CLV)** by improving satisfaction and loyalty.

Business and Social Impact

Churn doesn’t just hurt revenue - it erodes brand reputation and increases operational costs. By leveraging machine learning, we help the company:

- **Prevent revenue leakage** by retaining high-value accounts.
- **Improve customer experience** through personalized engagement.
- **Strengthen competitive positioning** in a market where customer loyalty is scarce.

This project aligns with broader industry trends, where data-driven retention strategies are becoming a key differentiator. By reducing churn, the company can reinvest savings into growth initiatives, creating a sustainable business model.

Data Report

Data Collection Methodology

The dataset was provided by GreatLearning on behalf of a leading e-commerce company, covering customer account data for the past financial year.

- **Time Period:** 12 months (last financial year)
- **Frequency:** Static snapshot (one-time extract)
- **Collection Method:** Extracted from the company's CRM and transactional databases. This includes account-level metrics (e.g., tenure, revenue, customer care interactions).

Data Dictionary is available under *Appendix : Data Dictionary*

Data Quality

- **Duplicate Records:** Comprehensive check revealed no duplicate records in the dataset. Verified uniqueness of AccountID (primary key) - all 11,260 IDs were distinct.
- **Data Integrity Issues:** Found invalid entries containing #, $, +, @ in multiple columns in the dataset which might be due to data entry errors:
    - **Tenure:** '#' (invalid numeric value)
    - **Account_user_count:** '@' (invalid count)
    - **rev_per_month:** '+' (invalid revenue value)
    - **rev_growth_yoy:** '$' (invalid percentage)
    - **coupon_used_for_payment:** '#', '$', '*' (invalid counts)
- **Data Inconsistencies:** Found inconsistencies in the Categorical Values:
    - **Gender:** Contained both "Male"/"M" and "Female"/"F"
    - **account_segment:** Had variations like "Super Plus" vs "Super+" and "Regular+" vs "Regular"
- **Missing Values:** There were several missing values in the features:
    - **rev_per_month:** 7.02% missing values
    - **Complain_ly:** 3.17% missing values
    - **Tenure:** 1.94 % missing values
    - **Marital_Status:** 1.88% missing values
    - **City_Tier:** 0.99% missing values

Detailed chart is available under *Appendix : Missing Value Check*

- **Outliers:** Extreme values in several numerical columns:
    - **CC_Contacted_LY**: max = 132 contacts/year
    - **rev_per_month**
    - **cashback**

Handling Data Quality Issues

- **Special Character Treatment:**
    - All special character entries were replaced with NaN
    - Subsequent imputation performed based on column characteristics
- **Categorical Value Standardization:**
    - Consolidated gender values: "M" → "Male", "F" → "Female"
    - Unified segment names: "Super+" → "Super Plus", "Regular+" → "Regular Plus"
- **Data Type Corrections:**
    - Converted numeric fields with special characters to proper data types
    - rev_per_month and rev_growth_yoy converted from object to float
- **Missing Values and Outliers Treatments:** Handling missing values and extreme values continues post EDA...]

Data Structure & Initial Inspection

- The Dataset contains 11,260 rows (accounts) × 19 columns (features).
- These features can be classified into 3 groups:

Type	Features
Continuous Numerical	Tenure, CC_Contacted_LY, Account_user_count, rev_per_month, rev_growth_yoy, coupon_used_for_payment, Day_Since_CC_connect, cashback
Discrete Numerical	Churn, City_Tier, Service_Score, CC_Agent_Score, Complain_ly
Categorical	Payment, Gender, account_segment, Marital_Status, Login_device

Table 1 - Type of Features

- Key Features:
    - **Target:** Churn (binary: 0 = retained, 1 = churned). It feature does not have missing values
    - **Numerical:** Tenure, rev_per_month, CC_Contacted_LY, Complaint_LY
    - **Categorical:** Payment, City_Tier, Gender, Marital_Status, account_segment.

Exploratory Data Analysis (EDA) and Business Implications

Key Findings from EDA

Univariate Analysis

Churn

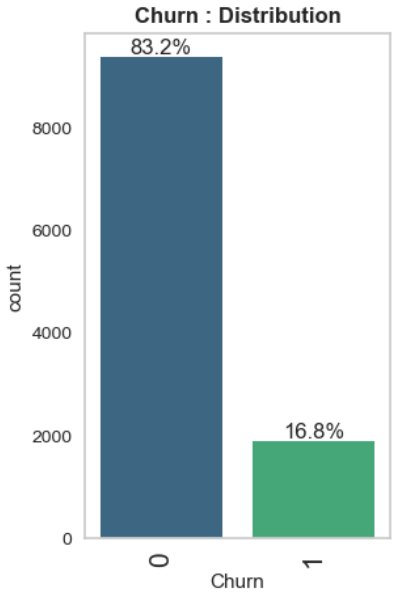


Figure 1 - Univariate Analysis - Churn

- **Churn Rate:** The dataset has a 16.84% churn rate, indicating 1 in 5 customers discontinue services annually.
- It is having binary values: 0 = retained, 1 = churned
- This is the target variable for our predictions and it does not have any missing values

City_Tier

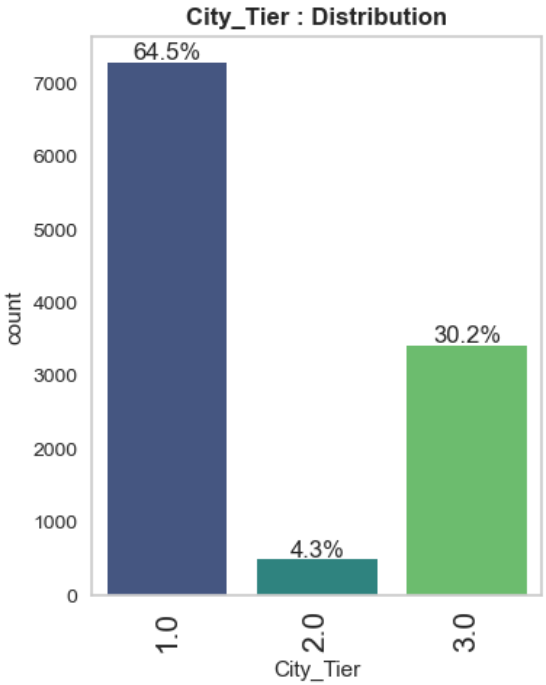


Figure 2 - Univariate Analysis - City_Tier

- The dataset is heavily skewed toward City Tier 1, which accounts for 64.5% of customers.
- City Tier 3 has the smallest representation (4.3%), indicating that very few customers reside in Tier 3 cities.
- There are 112 missing values in the City_Tier column, which need to be handled

Tenure

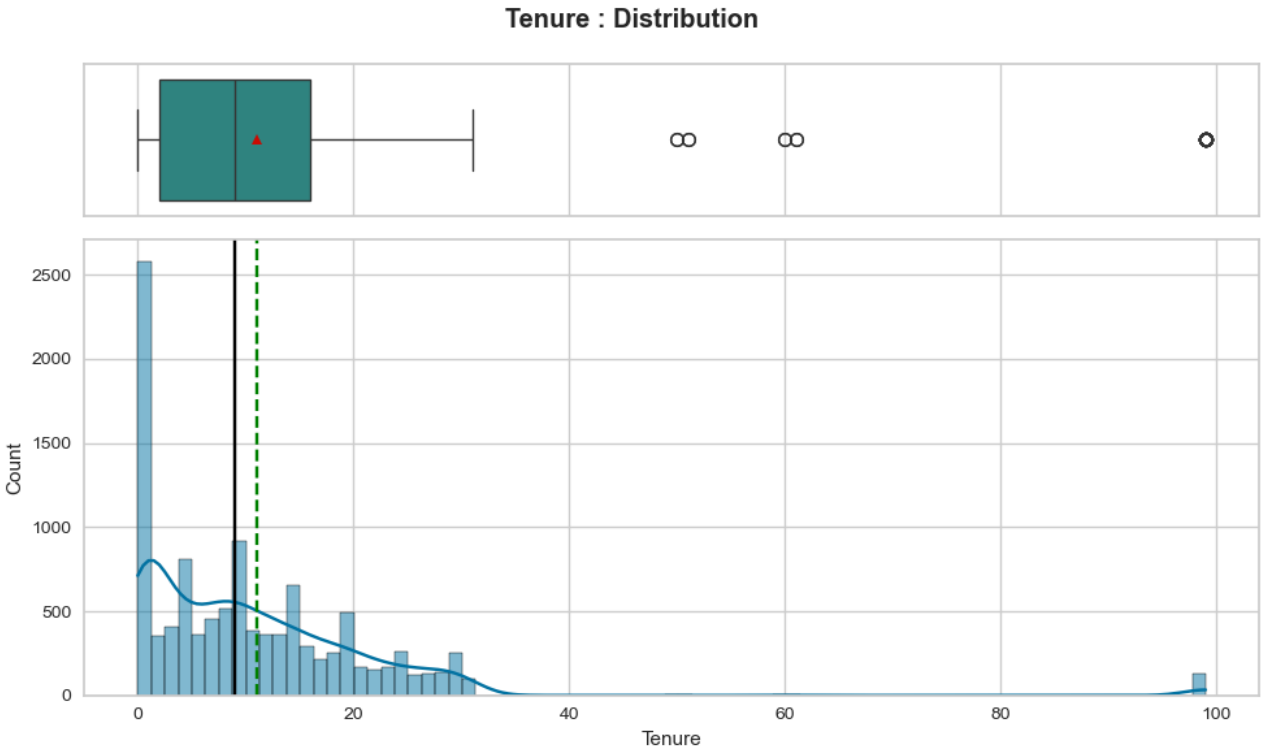


Figure 3 - Univariate Analysis - Tenure

Statistics	
count	11042
mean	11.02509
std	12.87978
min	0
0.25	2
0.5	9
0.75	16
max	99

Table 2 - Tenure - Statistics

- The distribution of tenure is right-skewed, with most customers having lower tenure values (0–20 months) and fewer customers having higher tenure values (up to 99 months).
- The peak at lower tenure values suggests that many customers are relatively new.
- There are 218 missing values in the Tenure column, which need to be handled
- The maximum tenure value (99 months) is significantly higher than the 75th percentile (16 months), suggesting potential outliers or long-term customers.

Payment

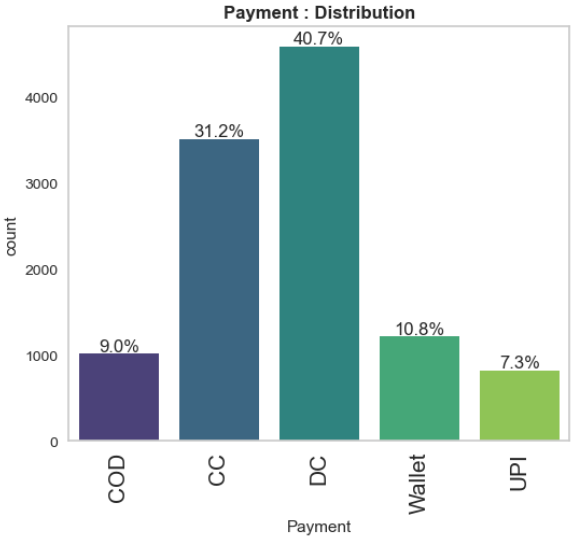


Figure 4 - Univariate Analysis - Payment

- The dataset is heavily skewed toward Credit Card (CC), which accounts for 31.2% of customers.
- Debit Card (DC) and Wallet are less frequently used, with 9.0% and 10.8% respectively.
- A small percentage (7.3%) of customers use UPI methods.
- There are 109 missing values in the Payment column, which need to be handled

CC_Contacted_LY

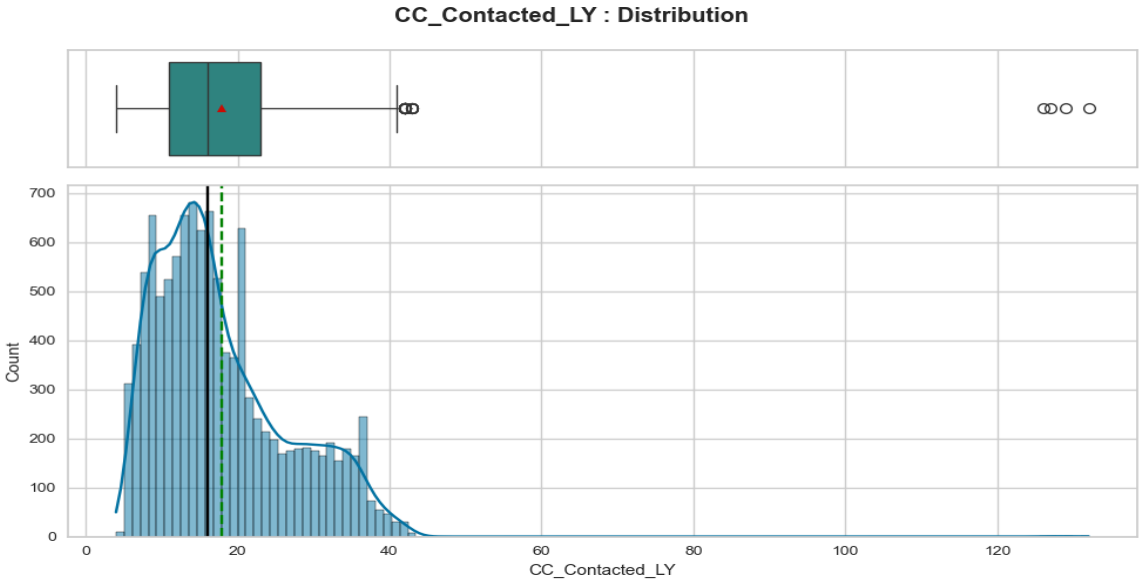


Figure 5 - Univariate Analysis - CC_Contacted_LY

Statistics	
count	11158
mean	17.86709
std	8.853269
min	4
25%	11
50%	16
75%	23
max	132

Table 3 - CC_Contacted_LY Statistics

- The distribution is right-skewed, with most customers being contacted between 10 and 20 times and a long tail extending up to 132 contacts.
- The maximum value (132) is significantly higher than the 75th percentile (23), suggesting potential outliers or customers who were contacted excessively.
- There are 102 missing values in the column, which need to be handled

Gender

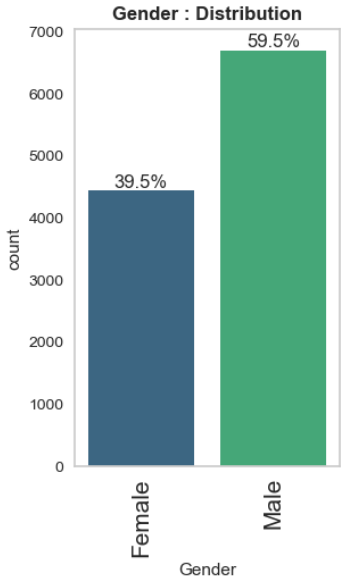


Figure 6 - Univariate Analysis - Gender

- Gender distribution is heavily skewed toward Male customers, with fewer Female customers.
- There are 109 missing values that need to be addressed

Account_segment

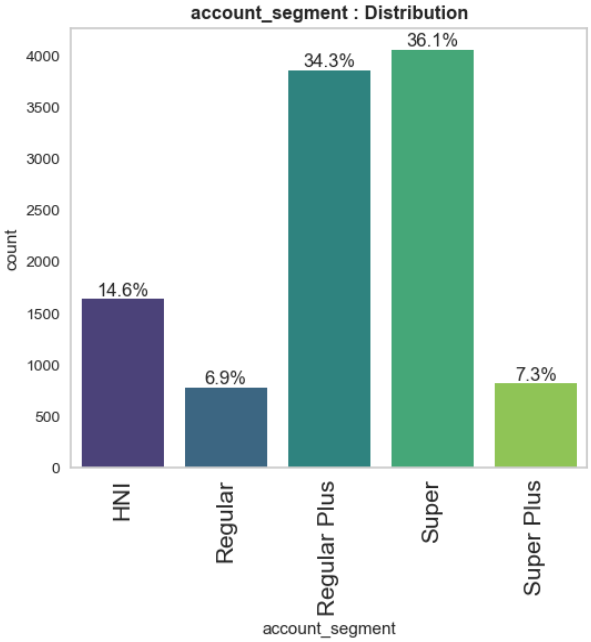


Figure 7 - Univariate Analysis - account_segment

- 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.

Marital_Status

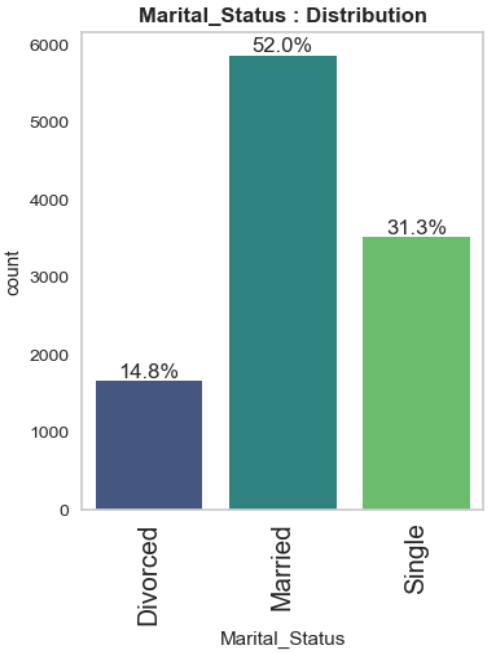


Figure 8 - Univariate Analysis - Marital_status

- 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.

Complain_LY

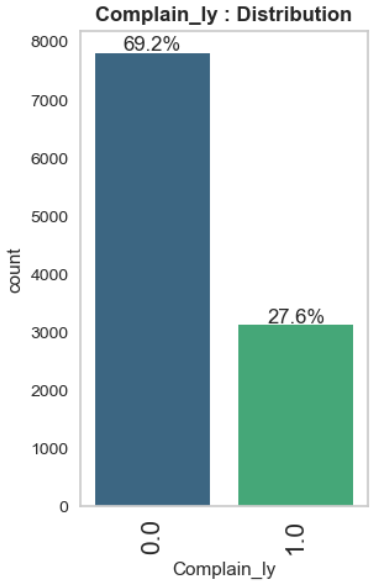


Figure 9 - Univariate Analysis - Complaint_LY

- 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_per_month

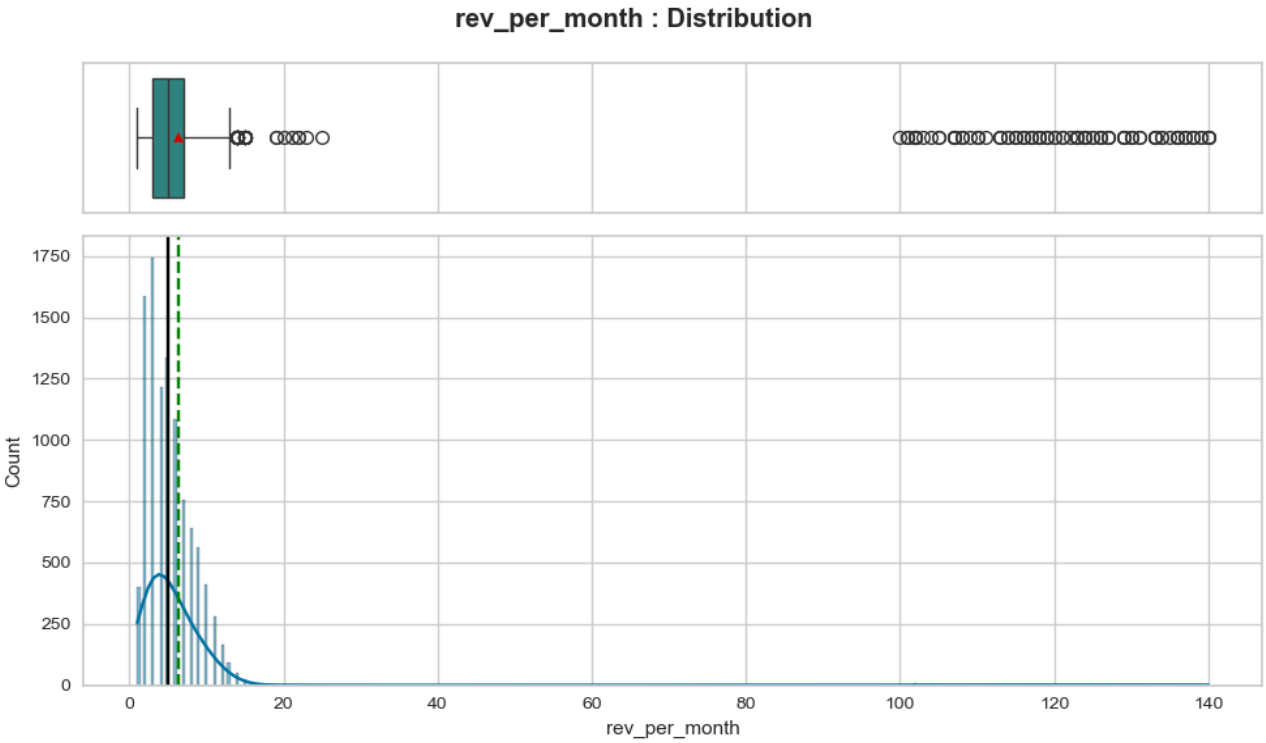


Figure 10 - Univariate Analysis - rev_per_month

Statistics	
count	11158
mean	17.86709
std	8.853269
min	4
25%	11
50%	16
75%	23
max	132

Table 4 - rev_per_month Statistics

- 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.

Day_since_CC_connect

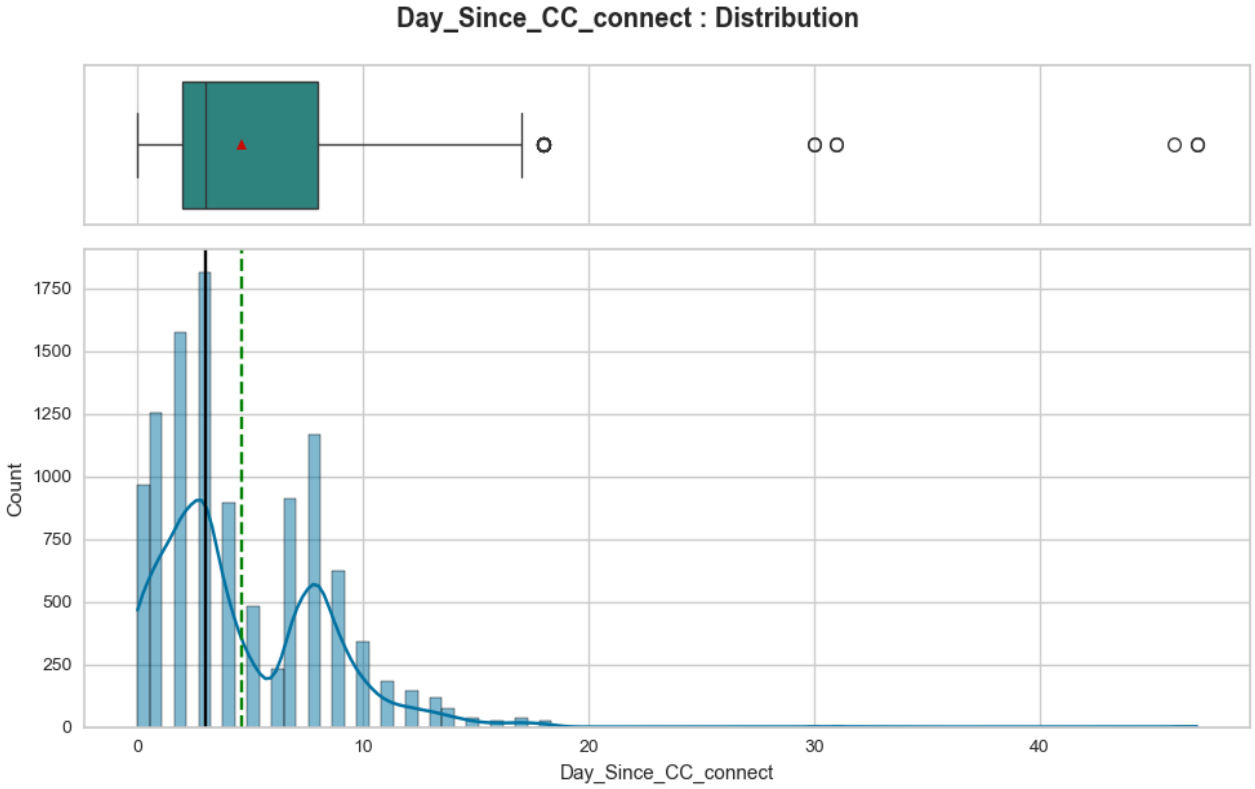


Figure 11 - Univariate Analysis - Day_since_CC_connect

Statistics	
count	10902
mean	4.633187
std	3.697637
min	0
25%	2
50%	3
75%	8
max	47

Table 5 - Day_since_CC_connect - Statistics

- 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.

Bivariate Analysis

Churn Vs Numerical Columns

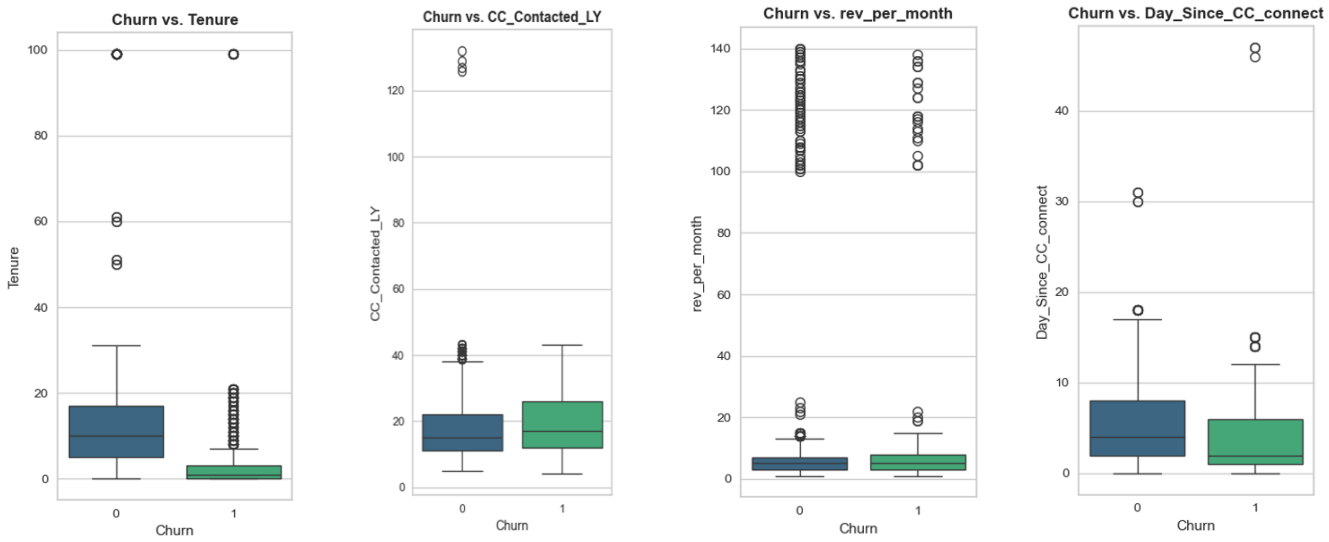


Figure 12 - Bivariate Analysis - Churn Vs Numerical Columns

- **Lower-tenure customers churn more frequently**, suggesting that newer customers are at higher risk.
- Long-term customers exhibit greater loyalty, while those with shorter tenures are more likely to leave.
- **Churned customers contact support more frequently**, with a higher median number of interactions.
- Excessive customer care contacts (outliers) correlate strongly with churn, possibly indicating unresolved issues or dissatisfaction.
- Some churned customers had **long gaps in support interactions**, which may suggest disengagement before leaving.
- **Lower-revenue customers are more prone to churn**, while high-revenue customers show greater retention.
- However, **even some high-revenue customers churn**, indicating that revenue alone does not guarantee loyalty.
- **Recent customer care interactions** often precede churn, signaling potential dissatisfaction or service issues.
- Extreme cases (e.g., very high/low interaction frequency) highlight **at-risk segments** that may need targeted retention strategies.

Churn Vs Categorical Columns

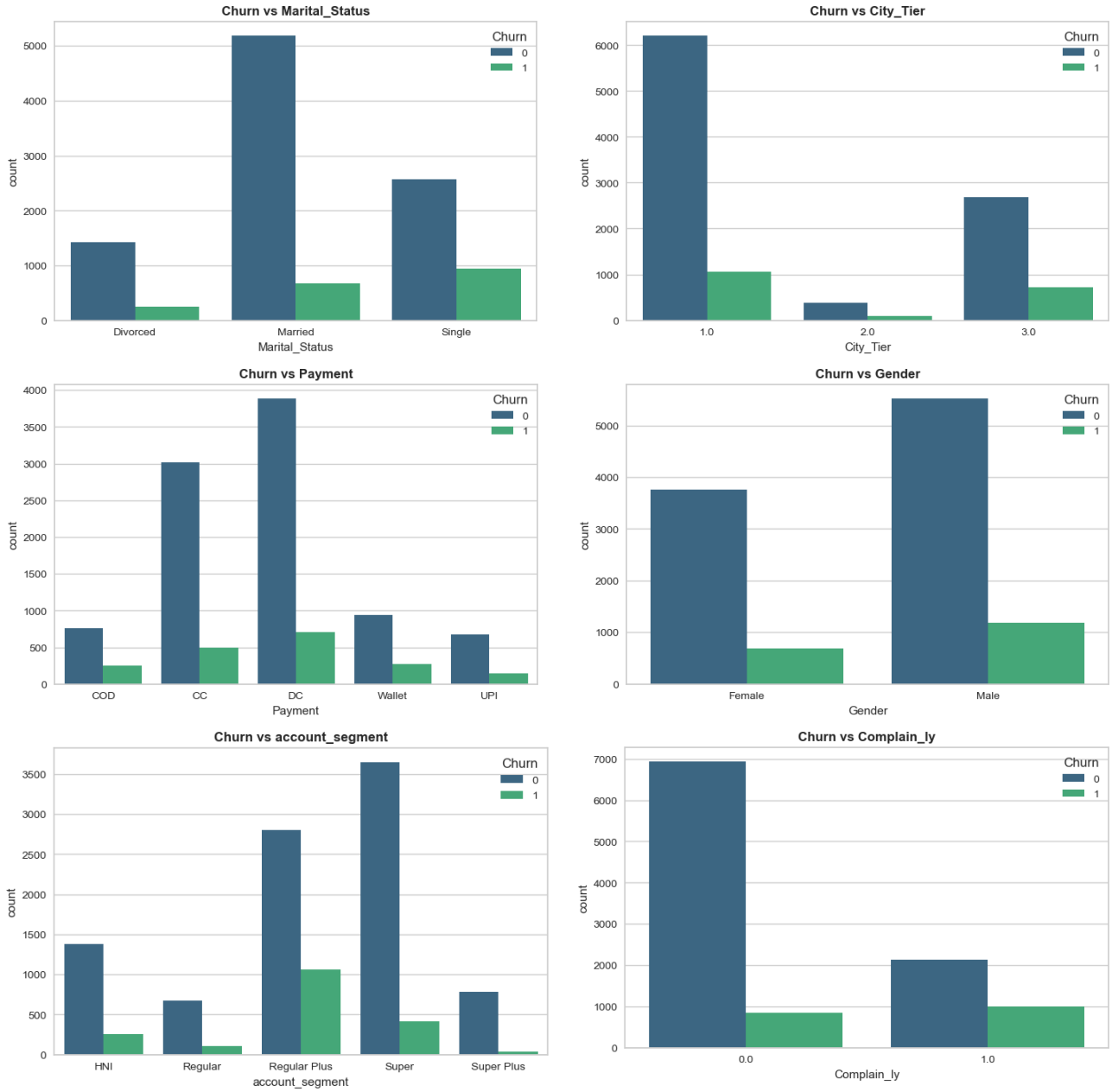


Figure 13 - Bivariate Analysis - Churn Vs Categorical Columns

Churn vs. City Tier

- **City Tier 1 has the highest customer base and churn rate**, making it a critical segment for retention efforts.
- **City Tier 2 shows the lowest engagement**, with fewer customers and minimal churn—suggesting limited market penetration.
- **City Tier 3 exhibits moderate churn,** indicating potential retention challenges despite a smaller customer base.

Actionable Insight:

- **Prioritize Tier 1** cities with loyalty programs and personalized offers to curb churn.
- Investigate **Tier 2's low engagement**—could be untapped potential or lack of awareness.
- Optimize customer experience in **Tier 3** to prevent further attrition.

Churn vs. Payment Method

- **Debit/Credit Card users dominate but churn more,** likely due to higher expectations or competition.
- **Wallet/UPI users show lower churn**, possibly due to convenience or loyalty incentives.
- **COD users are minimal and low-risk,** indicating infrequent purchases.

Actionable Insight:

- Target **card** users with exclusive perks or seamless payment experiences.
- Encourage **Wallet/UPI** adoption through rewards to retain stable customers.
- Re-engage **COD users** with subscription models or discounts for repeat purchases.

Churn vs. Gender

- **Male customers churn more despite being the majority**, signalling potential dissatisfaction.
- **Female customers exhibit higher loyalty,** possibly due to tailored engagement or product affinity.

Actionable Insight:

- Analyze male churn drivers (eg. service gaps, pricing) and address them proactively.
- Replicate successful female-centric strategies (eg. personalized communication) for males.

Churn vs. Account Segment

- "**Regular Plus**" segment has high churn, suggesting unmet needs or poor perceived value.
- "**Super Plus**" and "**Regular**" segments retain better, likely due to premium benefits or satisfaction.

Actionable Insight:

- Revamp "**Regular Plus**" offerings (e.g., enhanced support, value-added services).
- Strengthen **"Super" segment loyalty** with exclusive perks to prevent attrition.

Churn vs. Marital Status

- **Single customers churn more,** possibly due to less commitment or varied preferences.
- **Married** customers **are more stable**, while divorced users show the lowest churn.

Actionable Insight:

- Design targeted campaigns for Singles (e.g., flexible plans, social engagement).
- Leverage family-oriented benefits for Married customers to reinforce loyalty.

Churn vs. Complaints

- **Complainers churn significantly more**, highlighting unresolved issues as a key driver.
- **Non-complainers are passive but stay longer**, possibly due to satisfaction or inertia.

Actionable Insight:

- **Improve complaint resolution** speed and transparency to retain dissatisfied customers.
- **Proactively engage** silent users to uncover hidden pain points before they churn.

Multivariate Analysis

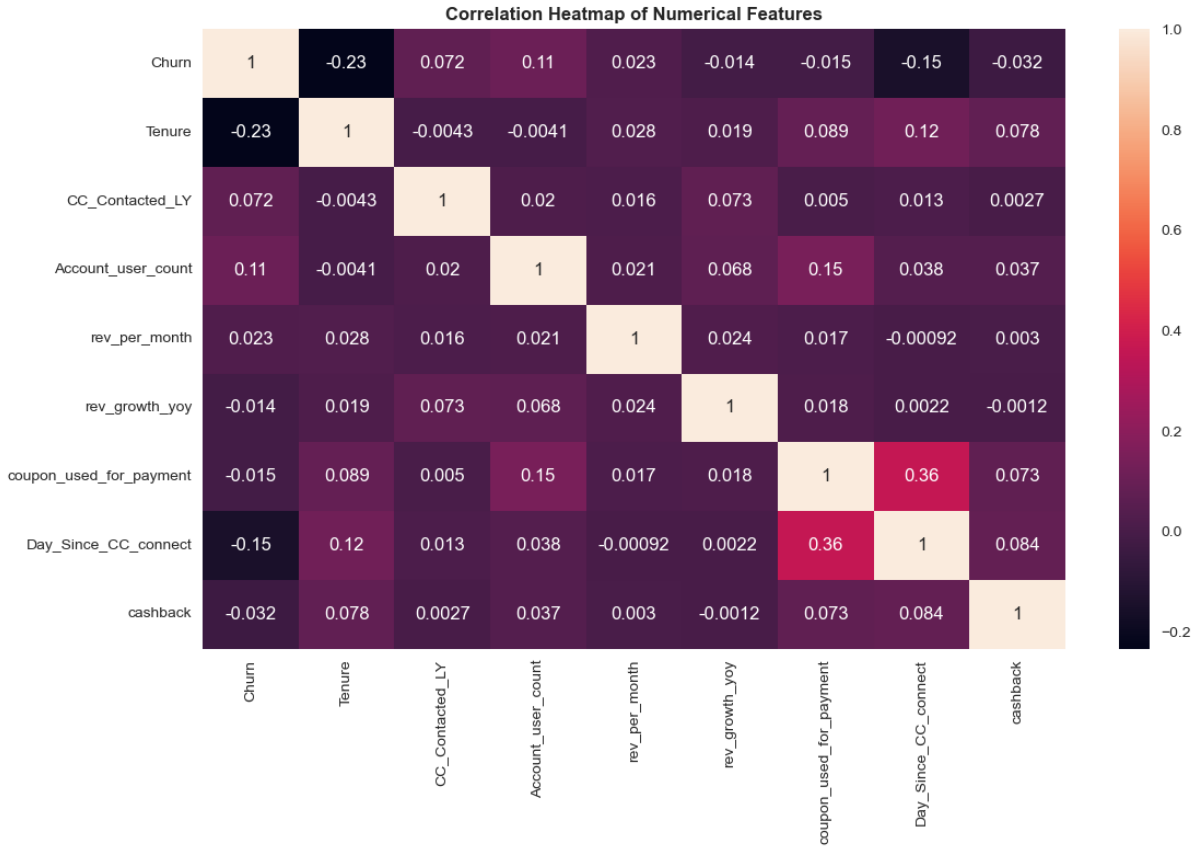


Figure 14 - Multivariate Analysis - Heatmap of Numerical Features

Churn Correlations

- **Churn is negatively correlated with Tenure (-0.23)**, suggesting that customers with longer tenure are less likely to churn.
- **Day_Since_CC_connect has a weak negative correlation with Churn (-0.15)**, implying that recent interactions with customer care might reduce churn.
- Similarly, **Account_user_count** has a weak positive correlation with Churn (0.11)

Tenure Correlations

- **Tenure has a weak positive correlation with Day_Since_CC_connect (0.12)**, indicating that long-tenured customers may have slightly more time since their last customer care interaction.

Customer Engagement & Revenue Relationships

- **Account_user_count and coupon_used_for_payment (0.15) indicate that accounts with more users are slightly more likely to use coupons for payment.**
- **Day_Since_CC_connect and coupon_used_for_payment (0.36) have the highest correlation in the heatmap,** suggesting that customers who interact with customer care might be more likely to use coupons.

Revenue & Growth Trends

- **Revenue per month (rev_per_month) and revenue growth year-over-year (rev_growth_yoy) show minimal correlation with other factors**, indicating revenue trends are not strongly linked to churn, tenure, or account interactions.

Business Implications

Below are the key business implications from Exploratory Data Analysis:

1. Prioritize Retention Efforts for High-Churn Segments

a. Focus on Newer Customers (Low Tenure)

- **Observation:** Lower-tenure customers (0–20 months) churn more frequently.
- **Implication:** New customers are at higher risk due to unmet expectations or poor onboarding experiences.
- **Action:**
  - Strengthen onboarding programs (e.g., personalized welcome offers, proactive support).
  - Implement early engagement strategies (e.g., feedback surveys, loyalty incentives).

b. Address High Churn in City Tier 1

- **Observation:** Tier 1 cities contribute the most customers (64.5%) but also the highest churn rate.
- **Implication:** Competitive markets (Tier 1) may lead to easy switching behavior.
- **Action:**
  - Offer exclusive discounts or rewards for long-term customers.
  - Enhance service quality (e.g., faster delivery, better customer support).

c. Target "Regular Plus" Account Segment

- **Observation**: "Regular Plus" customers churn more despite being a significant segment.
- **Implication:** They may feel undervalued compared to "Super" or "Super Plus" tiers.
- **Action:**
  - Introduce mid-tier loyalty benefits (e.g., cashback, priority support).
  - Upsell premium features to enhance perceived value.

2. Optimize Payment & Customer Support Strategies

a. Reduce Churn Among Card (CC/DC) Users

- **Observation:** Credit/Debit Card users churn more than Wallet/UPI users.
- **Implication:** Card users may face payment failures or expect better service.
- **Action:**
  - Improve payment success rates (e.g., retry mechanisms, saved card benefits).
  - Offer incentives for switching to Wallet/UPI (e.g., cashback on first use).

b. Improve Complaint Resolution Efficiency

- **Observation:** Customers who complained had significantly higher churn.
- **Implication:** Unresolved issues drive dissatisfaction.
- **Action:**
  - Implement faster complaint resolution (e.g., AI chatbots, dedicated support teams).
  - Proactively follow up with dissatisfied customers (e.g., apology discounts).

c. Balance Customer Care Interactions

- **Observation:** Excessive CC contacts correlate with churn (some customers contacted 132 times!).
- **Implication:** Over-contacting can annoy customers, while under-contacting may lead to disengagement.
- **Action:**
  - Optimize contact frequency (e.g., predictive outreach based on customer needs).
  - Train agents to resolve issues in fewer interactions.

3. Strengthen Customer Engagement & Loyalty

a. Increase Engagement with Single Customers

- **Observation:** Single customers churn more than Married/Divorced ones.
- **Implication:** They may have less brand attachment or seek variety.
- **Action:**
  - Launch personalized offers (e.g., discounts on trending products).
  - Gamify engagement (e.g., reward points for referrals).

b. Leverage Revenue-Based Retention Strategies

- **Observation:** Low-revenue customers churn more, but even some high-revenue customers leave.
- **Implication:** Revenue alone doesn’t guarantee loyalty; experience matters.
- **Action:**
  - For low-revenue customers: Introduce budget-friendly loyalty programs.
  - For high-revenue customers: Offer VIP perks (e.g., concierge service, exclusive deals).

c. Reduce Gaps in Customer Care Interactions

- **Observation:** Long gaps since last CC interaction precede churn.
- **Implication:** Disengaged customers are at risk of leaving.
- **Action:**
  - Send re-engagement emails/SMS (e.g., "We miss you!" offers).
  - Use AI to predict disengagement and trigger retention campaigns.

Data Cleaning and Pre-processing

The foundation of a robust customer churn prediction model lies in meticulous data preprocessing. This phase ensures data quality, mitigates bias, and enhances model generalizability. A critical step in this process is the **train-test split**, where the dataset is divided into training, validation, and test sets **before** any preprocessing. This prevents **data leakage**, ensuring that the model's performance metrics reflect real-world applicability rather than inflated training accuracy.

Following the split, systematic approaches were applied to handle missing values, treat outliers, and transform variables. Key steps included:

- **Missing Value Imputation:** Median/mode imputation based on data distribution.
- **Outlier Treatment:** Capping/log transformations to minimize skewness.
- **Feature Engineering:** Creating derived variables (e.g., Loyalty_Score) to strengthen predictive signals.

This structured preprocessing pipeline ensures the dataset is optimized for accurate and interpretable churn prediction.

Handling Missing Values and Outliers

Approach for Missing Values

The dataset contained missing values across multiple columns, which were addressed based on data distribution and business context:

1. Numerical Columns (Median Imputation)

Columns Treated: Tenure, CC_Contacted_LY, rev_per_month, cashback, Day_Since_CC_connect.

Reason: These columns were right-skewed with outliers. Median imputation preserved central tendency without being influenced by extreme values.

2. Categorical Columns (Mode Imputation)

Columns Treated: City_Tier, Payment, Gender, Service_Score, account_segment, Marital_Status, Complain_ly.

Reason: Mode imputation maintained the most frequent category, ensuring minimal distortion of categorical distributions.

3. Special Cases

rev_growth_yoy: Mean imputation was used due to its near-normal distribution.

Account_user_count: Treated as a categorical-like numerical variable; mode imputation was applied.

Preventing Data Leakage

Imputers were fitted **only on the training set** and applied to validation/test sets to avoid bias.

Outlier Treatment

Outliers were capped at the 95th/99th percentiles or log-transformed to mitigate skewness:

Column	Treatment	Reason
Tenure	Capped at 99th percentile	Extreme values (e.g., 99 months) were rare and distorted analysis.
CC_Contacted_LY	Capped at 99th percentile	Excessive contacts (>50) likely indicated data errors or unresolved issues.
rev_per_month	Log transformation	High revenue values (>100) skewed the distribution.
cashback	Capped at 99th percentile	Extreme cashback amounts (>500) were unrealistic.
coupon_used_for_payment	Capped at 95th percentile	Excessive coupon usage (>8) was rare and unrepresentative.

Table 6 - Outlier Treatment

Variable Transformations

Binning Numerical Variables

Tenure → Tenure_Group:

Binned into categories (e.g., "0–3 months," "3–12 months") to capture non-linear relationships with churn.

Impact: Cramer's V increased from 0.23 (original) to 0.44, strengthening predictive power.

Feature Engineering

Loyalty_Score:

Derived from Tenure, coupon_used_for_payment, rev_per_month, and Complain_ly.

Formula:

Loyalty_Score = (Tenure / max(Tenure)) + (coupon_usage / max(coupon_usage)) + (revenue / max(revenue)) - (Complain_ly / max(Complain_ly))

Correlation with Churn: -0.135, indicating higher scores reduce churn likelihood.

Variable Addition and Removal

Variables Added

Tenure_Group: Replaced Tenure to improve interpretability and model performance.

Loyalty_Score: Aggregated multiple behavioral metrics into a single churn indicator.

Variables Removed

AccountID: Unique identifier with no predictive value.

Tenure: Redundant after binning.

For New Data: Use saved imputers, outlier thresholds, and encoders to ensure consistency.

Modeling Priority: Focus on high-impact features like Tenure_Group, Loyalty_Score, and customer interaction metrics.

This preprocessing pipeline ensures the dataset is optimized for accurate churn prediction while maintaining business interpretability.

Model Building

When evaluating a customer churn prediction model, it is crucial to understand the types of errors the model can make and their implications. The model can make wrong predictions in two primary ways:

1. False Positives (Type I Error):

- The model predicts that a customer will churn, but in reality, they do not.
- Implication: The business may spend resources (e.g., discounts, offers) on retaining customers who were not going to leave, leading to unnecessary costs.

2. False Negatives (Type II Error):

- The model predicts that a customer will not churn, but in reality, they do.
- Implication: The business loses customers without attempting to retain them, resulting in lost revenue and potential damage to customer relationships.

Which case is more important?

False Negatives are Often More Critical:

- Losing a customer (FN) is typically more costly than spending resources on retaining a customer who wasn't going to leave (FP).
- Retaining customers is usually cheaper than acquiring new ones, so missing a churning customer (FN) can have a higher long-term impact.

Which metric to optimize?

Here are some widely used metrics for evaluating churn prediction models:

- **Accuracy:** Measures the proportion of correctly classified customers (both churners and non-churners). However, accuracy can be misleading in imbalanced datasets where the number of non-churners far exceeds churners.

- **Precision:** Measures the proportion of predicted churners who are actually churners. High precision means fewer false positives, which is important if retention efforts are expensive.
- **Recall (Sensitivity):** Measures the proportion of actual churners correctly identified by the model. High recall means fewer false negatives, which is critical if missing a churner is costly.
- **F1-Score:** The harmonic mean of precision and recall. It is a good metric when you want to balance both false positives and false negatives.
- **ROC-AUC:** Measures the model's ability to distinguish between churners and non-churners across different probability thresholds. A high AUC indicates a better-performing model.

Considering the Dataset:

i. **Business Goal:** Retain high-value customers who are likely to churn.

ii. **Data Characteristics:** Imbalanced dataset **(83.2% non-churners, 16.8% churners)**.

iii. **Metric Choice:** Prioritize **Recall Score** to ensure that most high-value churners are identified, even if it means some false positives.

We will refine the model to improve precision by targeting retention efforts based on customer value.

Optimize Recall when minimizing FNs is critical (e.g., when losing customers is more costly than retaining non-churning customers).

We would want Recall-Score to be maximized, the greater the Recall-Score higher the chances of predicting FN classes correctly.

Recall (Sensitivity):

- **Formula:** Recall = TP / (TP + FN)
- **Implication:** Ensures that most churning customers are identified, even if it means some non-churning customers are flagged.

Model Selection

The primary objective of this project was to predict customer churn accurately, with a focus on minimizing **false negatives** (missed churners) due to their higher business impact.

The project tested multiple algorithms to balance **recall** (minimizing missed churners) and **precision** (avoiding unnecessary retention costs).

Models Evaluated:

Model	Key Strength	Why It Was Considered/Rejected
Logistic Regression	Interpretability	Rejected: Lower recall (66.1% validation).
Random Forest	Handles non-linearity	Rejected: Overfit (training recall: 100%, validation: 76.6%).
XGBoost	Boosting performance	Rejected: Overfit (training recall: 99.9%, validation: 84.5%).
SVM (RBF Kernel)	High-dimensional separation	Chosen Best recall (89.7%) + generalizability.

Table 7 - Model Evaluation

After evaluating multiple models, the **Support Vector Classifier (SVC)** was selected as the final model due to its superior performance in recall (89.71%) and balanced precision (91.15%).

Key Reasons for Choosing SVC:

- **High Recall:** Critical for identifying as many churners as possible.
- **Generalization:** Minimal overfitting compared to tree-based models (e.g., Random Forest, XGBoost).
- **Robustness:** Handles non-linear relationships well with the RBF kernel.

Other models considered:

- **Logistic Regression:** Simple but lower recall (80.86%).
- **Random Forest/XGBoost:** Overfit on training data (recall dropped significantly on validation).

Data Preprocessing for Model Improvement

Feature Encoding:

Categorical Variables (e.g., Payment_Method, Marital_Status) were encoded using:

- **One-Hot Encoding**: For nominal features (e.g., Payment_COD, Payment_UPI).
- **Ordinal Encoding**: For ordinal features (e.g., Tenure_Group: "Low"=0, "Medium"=1, "High"=2).

Feature Scaling:

Applied **StandardScaler** to normalize numerical features (e.g., rev_per_month, CC_Agent_Score) to mean=0 and variance=1.

This is critical for SVC (distance-based) and Logistic Regression (convergence).

Performance Optimization Steps

- **Class Imbalance Handling:** Applied **SMOTE** to oversample the minority class (churners),

  - Applied **SMOTE** to oversample churners (16.8% → 50% in training data).
  - Result: Recall improved from 66.1% to 89.7%.

- **Feature Selection:** Removed multicollinear features (VIF > 5) and insignificant predictors (p-value > 0.05).
  - Removed multicollinear features (e.g., Loyalty_Score with VIF > 17).
  - Dropped insignificant predictors (p-value > 0.05) like Service_Score.
- **Hyperparameter Tuning:** Further optimized SVC chosing best configuration with Hyperparameter turning :
  - **Best Configuration:** C=10, kernel=rbf, class_weight=balanced.
- **Threshold Adjustment:** Used ROC-AUC and precision-recall curves to set an optimal decision threshold (0.52), balancing recall and precision.
  - Used **precision-recall curves** to select a threshold of **0.52** (default: 0.5).
  - Trade-off: Recall = 89.7%, Precision = 91.2% (vs. 85%/88% at default threshold).

Model Validation

Validation Approach

The model was validated using:

1. **Data Splitting:**
   Stratified Train-Validation-Test Split (70-15-15). Ensured proportional representation of churners.
2. **Key Metrics:**

Metric	Purpose	SVC Performance
Recall	% of churners correctly identified	0.8971
Precision	% of predicted churners who actually left	0.9115
F1-Score	Balance of recall/precision	0.9043
ROC-AUC	Model's ranking capability (0.5=random)	0.99

Table 8 - SVC Key Metrics

  - **Recall (Sensitivity):** 89.71% (minimizes missed churners).
  - **Precision:** 91.15% (limits false positives).
  - **F1-Score:** 90.43% (harmonic mean of precision/recall).
  - **ROC-AUC:** 0.99 (excellent class separation).

3. ROC-AUC Curve

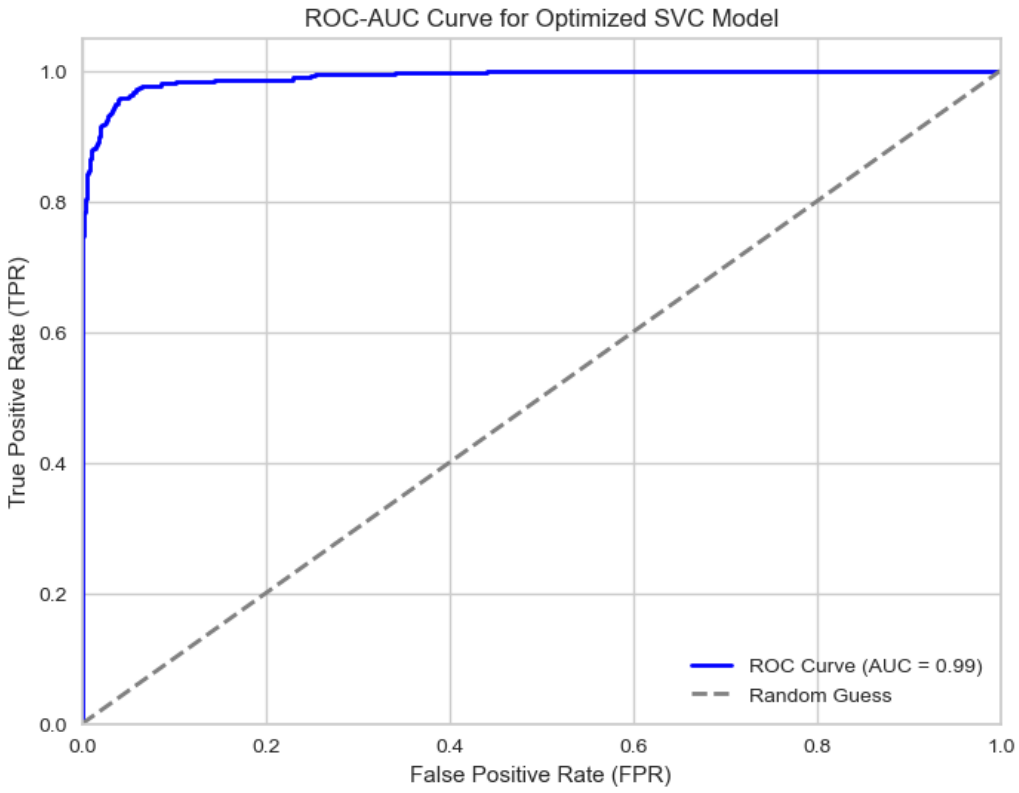


Figure 15 - ROC-AUC Curve for Optimized SVC Model

High AUC Score (0.99)

- The model achieves an AUC of 0.99, indicating exceptional classification performance.
- This suggests that the model can effectively distinguish between positive and negative classes.

Near-Perfect Performance

- The ROC curve is close to the top-left corner, demonstrating high sensitivity (True Positive Rate) with very low False Positives.
- This means the model correctly identifies most positive cases while minimizing incorrect predictions.

Minimal False Positives

- The False Positive Rate (FPR) remains close to zero for most threshold values.
- This indicates that the model makes very few incorrect positive predictions.

Strong Separation of Classes

- The ROC curve (blue line) is significantly above the random classifier line (grey dashed line).
- This confirms that the model performs far better than random guessing

4. **Confusion Matrix (On Test Set)**

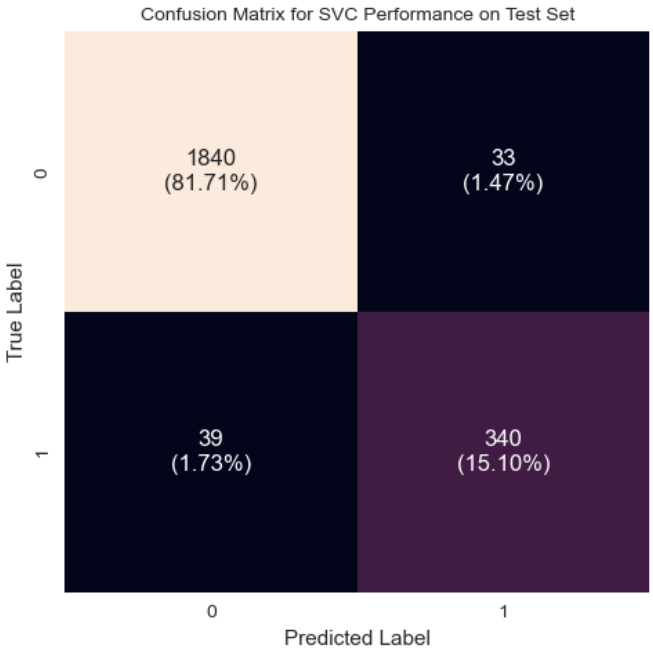


Figure 16 - Confusion Matrix of SVC on Test Set

- **High Recall (89.71%):** The model successfully captures most churn cases, which is crucial for customer retention strategies.
- **Low False Negatives:** Only 39 actual churn cases were missed, meaning the model effectively identifies customers at risk of leaving.
- **Low False Positives:** With just 33 non-churn customers misclassified as churners, unnecessary interventions are minimal.

Why Not Just Accuracy?

- **Accuracy:** 96.8% accuracy was misleading because the dataset had 83% non-churners. A "dumb model" predicting "no churn" for everyone would achieve 83% accuracy but 0% recall.

- **Solution:** Focused on **recall-driven metrics** to align with business goals.

Cross-Validation

- Used **5-fold cross-validation** during hyperparameter tuning to ensure stability.

- **SVC Consistency:** Recall ranged from 87–92% across folds (low variance).

Top 5 Features Influencing Churn

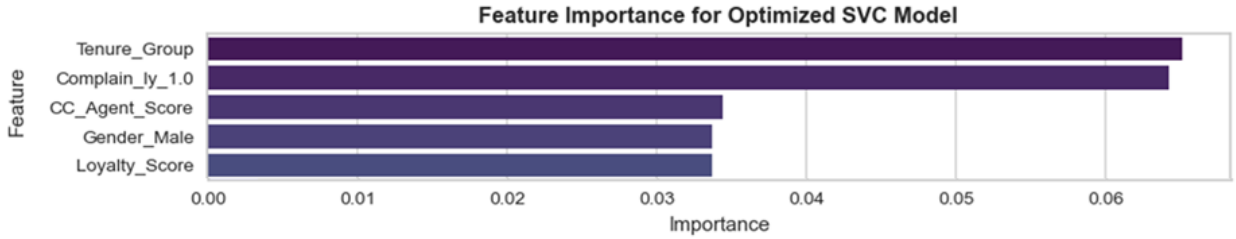


Figure 17 - Top 5 Features for Optimized SVC Model

Feature	Importance
Tenure_Group	0.065142
Complain_ly_1.0	0.064298
CC_Agent_Score	0.034503
Gender_Male	0.033792
Loyalty_Score	0.033748

Figure 18 - Feature Importance Table for Optimized SVC Model

- **Tenure_Group**: Customer tenure is the most influential factor in predicting churn. This suggests that how long a customer has been associated with the company significantly impacts churn probability.
- **Complain_ly_1.0**: Customers who complained in the last year are highly likely to churn, indicating dissatisfaction.
- **CC_Agent_Score**: The quality of customer care interactions (e.g., service ratings) plays a critical role in retention.
- **Gender_Male**: Male customers appear to have a higher correlation with churn compared to females.
- **Loyalty_Score**: Customer loyalty score significantly influences retention.

Final Interpretation & Recommendations

Key Insights

1. **Tenure Group & Complaints Are Key Indicators of Churn**
    - Customers with shorter tenure are more likely to churn. Retention strategies should focus on new customers.
    - Complaints in the last year strongly influence churn. Improving customer service and addressing complaints proactively can reduce churn.
2. **Customer Service Score Matters**
    - CC_Agent_Score and Loyalty_Score are among the top features. Customers who rate support poorly or have low loyalty scores are at high risk.

- Providing personalized offers and enhancing support quality can help retain customers.

3. **Revenue and Payment Methods Influence Churn**
    - Higher revenue per month correlates with retention. Understanding spending patterns and incentivizing consistent spending can enhance retention.
    - Payment methods like Wallet, UPI, and COD have lower importance but still contribute to churn. Offering more flexible and preferred payment options can improve customer experience.
4. **Demographics Play a Role**
    - Gender (Male) and Marital Status (Single) appear as significant predictors. Tailored marketing campaigns targeting these demographics could improve engagement.
    - City_Tier and Account Segment indicate that location and customer type influence churn behavior.

Actionable Recommendations

1. Enhance Customer Support & Retention Efforts

- **Proactive Complaint Resolution**
    - Resolve customer complaints **within 24 hours** to prevent dissatisfaction.
    - Implement a **dedicated escalation team** for high-risk accounts.
- **Agent Training & Performance**
    - Conduct **regular training** for support agents to improve **CC_Agent_Score**.
    - Introduce **customer satisfaction (CSAT) incentives** for agents.
- **Early Engagement for New Customers**
    - Strengthen onboarding processes with welcome offers and tutorials.
    - Schedule proactive check-ins within the first 30 days to address concerns.

2. Targeted Retention Campaigns

- **High-Risk Customer Segmentation**
    - Focus on **short-tenure customers, single males, and frequent complainers**.
    - Deploy **personalized offers** (e.g., discounts, free upgrades) to retain them.
- **Loyalty & Rewards Programs**
    - Introduce **tiered loyalty benefits** for long-term customers.
    - Offer **exclusive perks** (early access, cashback) to high-value users.

3. Optimize Payment & Revenue Strategies

- **Incentivize Consistent Spending**
  - Provide **discounts or bonus rewards** for customers with high **rev_per_month**.
  - Launch **subscription plans** with auto-renewal benefits.
- **Flexible Payment Options**
  - Promote **autopay incentives** (e.g., 5% cashback for UPI/Wallet users).
  - Simplify **payment processes** to reduce friction.

4. Leverage Predictive Analytics for Decision-Making

- **Real-Time Churn Alerts**
  - Integrate the **SVM model into CRM tools** to flag at-risk customers.
  - Trigger **automated retention offers** based on churn probability.
- **Continuous Model Improvement**
  - **Re-train the model quarterly** with updated customer data.
  - Monitor **feature importance shifts** (e.g., new churn drivers).

5. Refine Marketing & Customer Engagement

- **Personalized Communication**
  - Use **behavioral insights** to tailor email/SMS campaigns.
  - Highlight **exclusive benefits** for at-risk segments.
- **Feedback-Driven Improvements**
  - Conduct **exit surveys** for churned customers to identify trends.
  - Adjust strategies based on **real-time feedback**.

By implementing these strategies, the business can **minimize revenue loss** and **strengthen long-term customer relationships**.

Expected Outcomes:

- **Reduced churn rate** by **15-20%** within 6 months.
- **Higher customer lifetime value (LTV)** through targeted retention.
- **Improved customer satisfaction (CSAT & NPS)** via proactive support.

Appendix

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

Table 9 - Data Dictionary

Exploratory Data Analysis - Detailed

Univariate Analysis

Service_Score

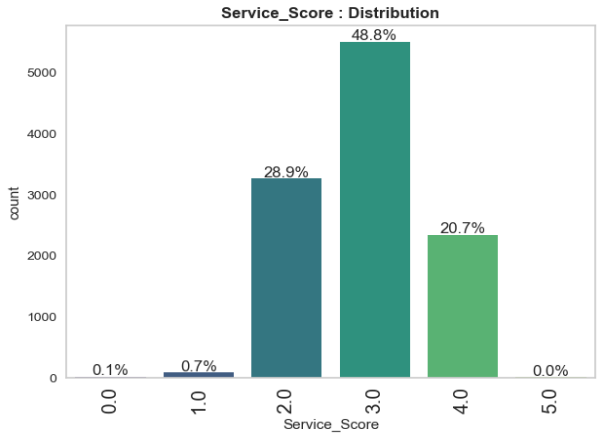


Figure 19 - Univariate Analysis - Service_Score

- The majority of customers (48.8%) gave a service score of 3.0.
- There are 98 missing values in the Service_Score column, which need to be handled
- Very few customers gave scores of 1.0 and 0.0.

Account_user_count

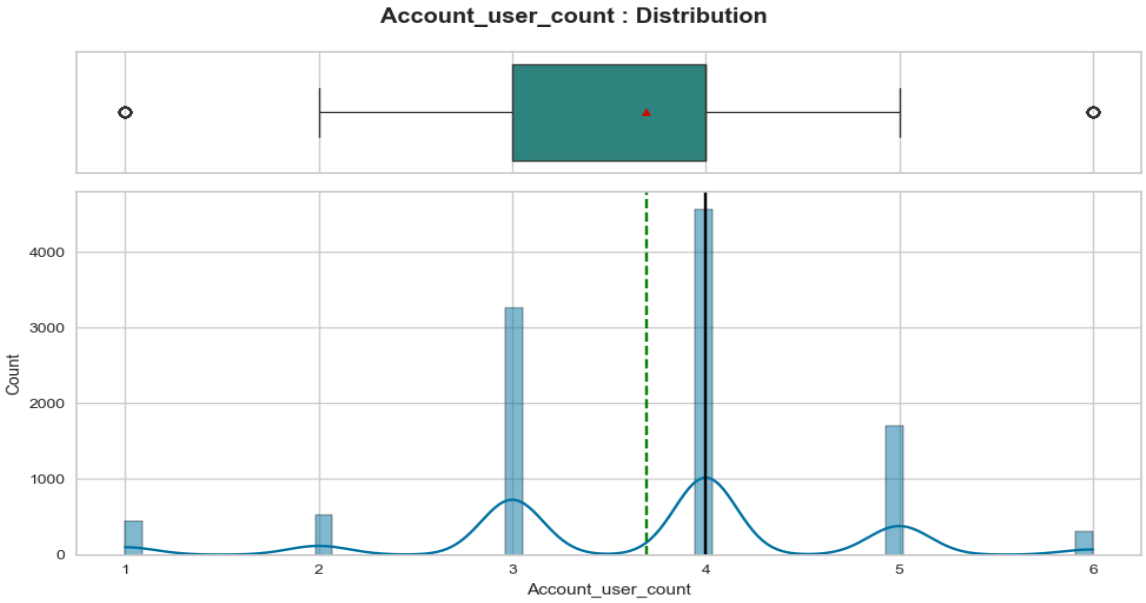


Figure 20- Univariate Analysis - Account_user_count

Statistics	
count	10816
mean	3.692862
std	1.022976
min	1
25%	3
50%	4
75%	4
max	6

Table 10 - Account_user_count Statistics

- The dataset is heavily skewed toward 4.0, which accounts for 40.6% of accounts.
- 3.0 and 2.0 are less frequent, with 29.0% and 15.1% respectively.
- Very few accounts have 1.0 user (2.8%).
- There are 444 missing values in the column, which need to be handled
- There are no significant outliers, as the data is tightly clustered around the median.

CC_Agent_Score

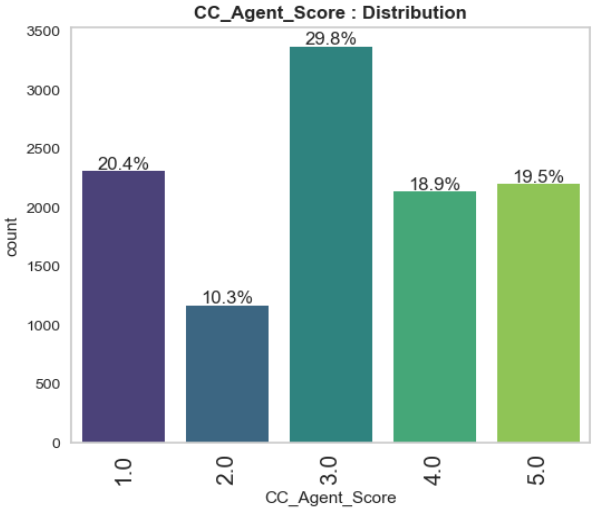


Figure 21- Univariate Analysis - Agent_Score

- 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.

Rev_growth_yoy

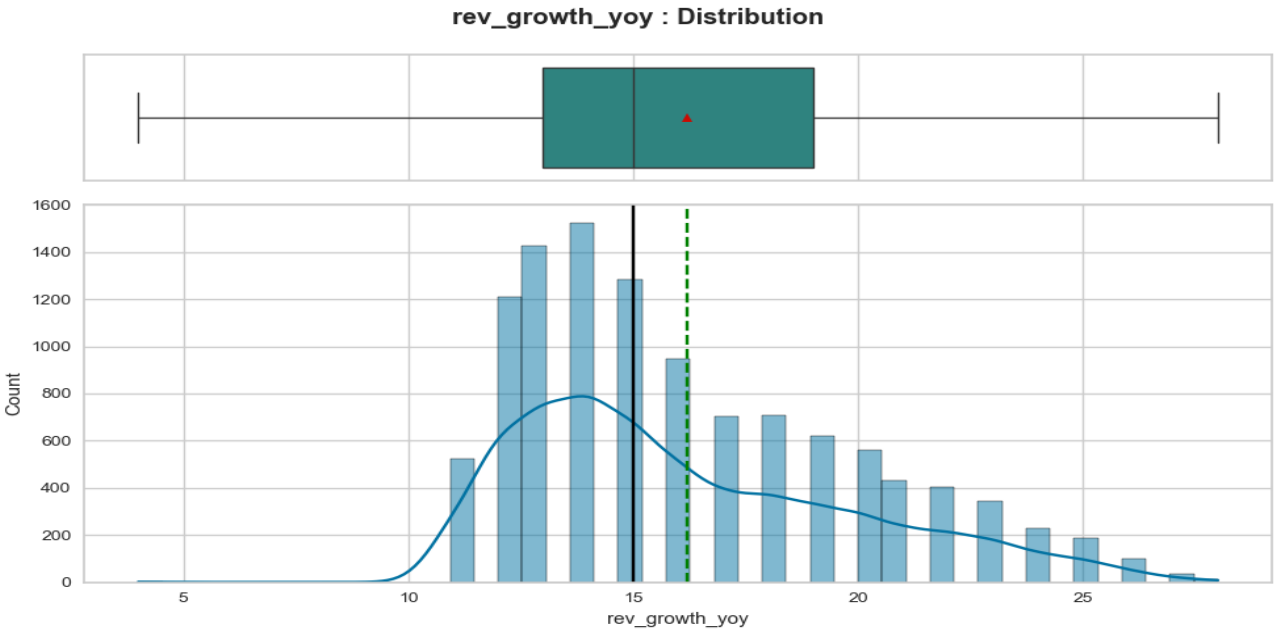


Figure 22- Univariate Analysis - rev_growth_yoy

Statistics	
count	11257
mean	16.19339
std	3.757721
min	4
25%	13
50%	15
75%	19
max	28

Table 11 - rev_growth_yoy statistics

- 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.

Login_device

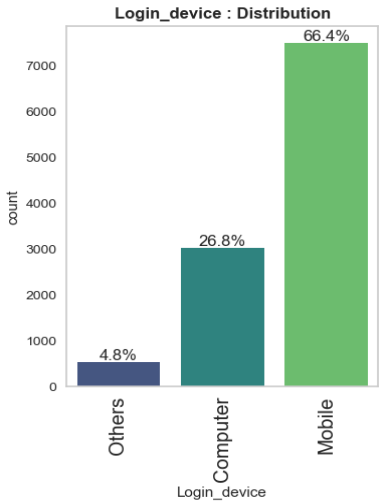


Figure 23 - Login_device

- 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.

Coupon_used_for_payment

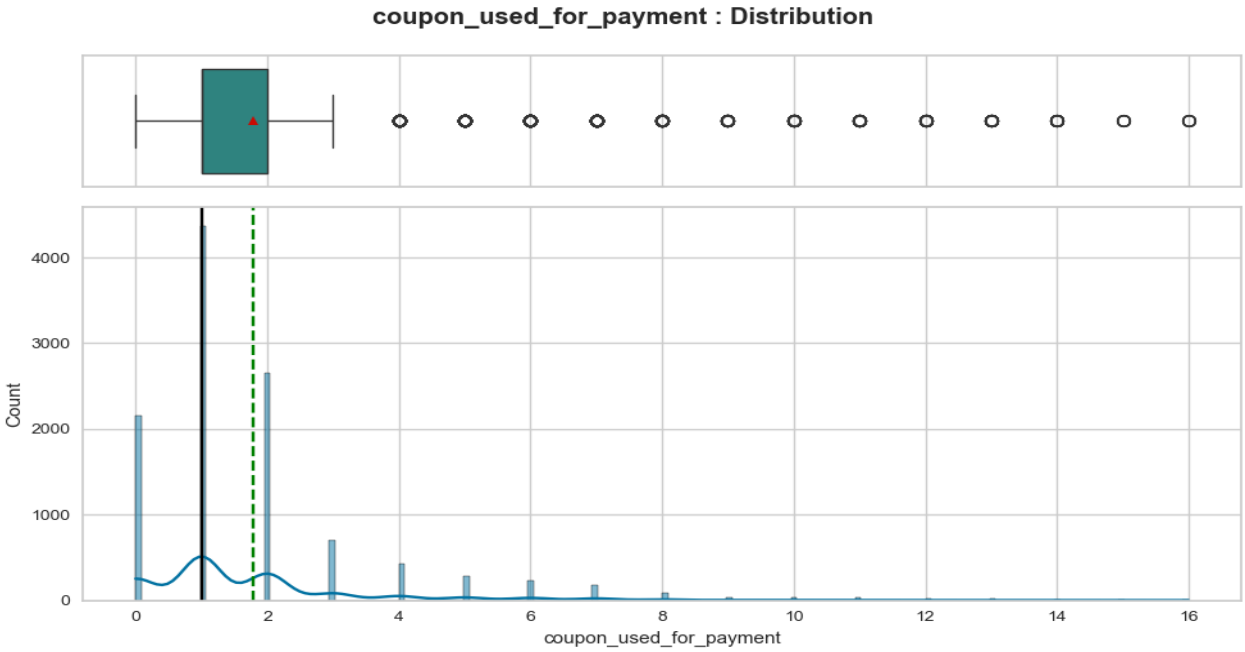


Figure 24- Univariate Analysis - coupon_usage

Statistics	
count	11257
mean	1.790619
std	1.969551
min	0
25%	1
50%	1
75%	2
max	16

Table 12 - Coupon_usage Statistics

- 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.

cashback

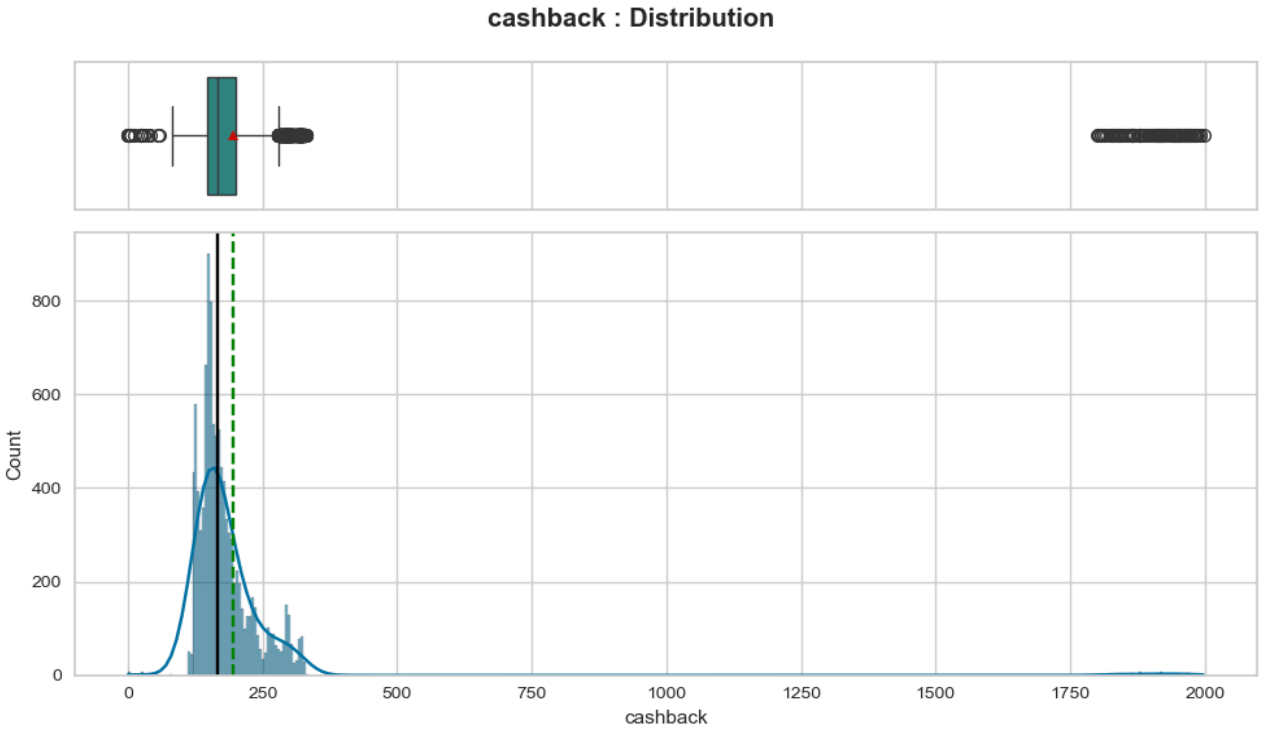


Figure 25- Univariate Analysis - cashback

Statistics	
count	10787
mean	196.2364
std	178.6605
min	0
25%	147.21
50%	165.25
75%	200.01
max	1997

Table 13 - casback statistics

- 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.

Bivariate Analysis

Churn Vs Numerical Columns

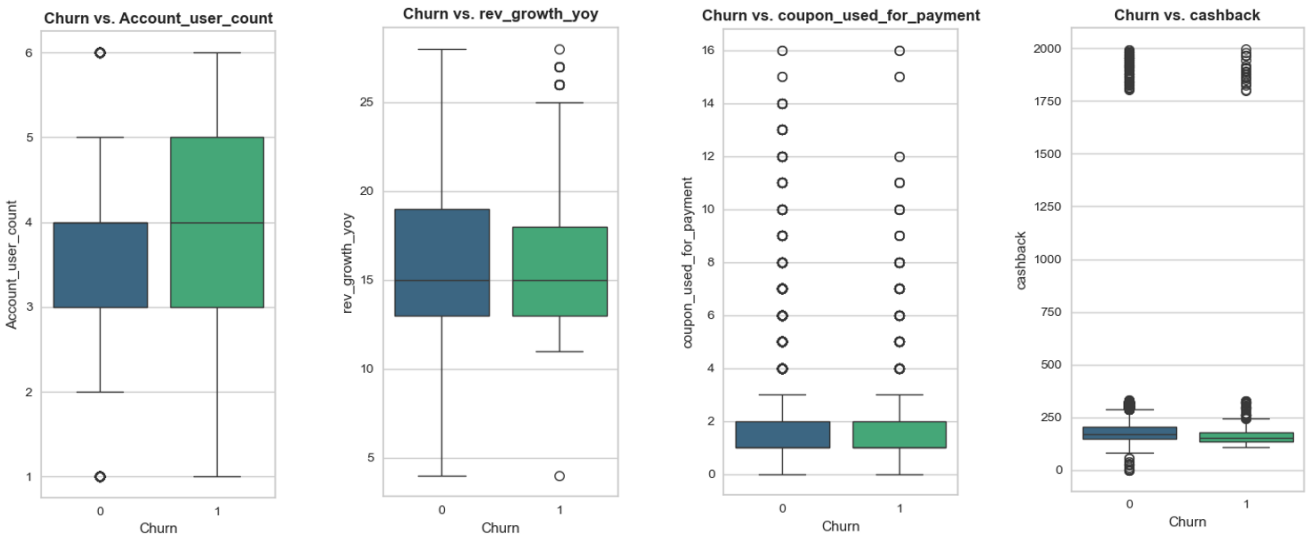


Figure 26 - Bivariate Analysis - Churn vs Numerical Columns

Account_user_count vs. Churn

- No strong trend, but higher user counts seem to have a wider spread for churned customers.
- Most Customers Have 3 or 4 Users on Their Account
- Lower Churn in Accounts with 1, 2, 5, or More Users
- Accounts with 3-4 users should be targeted for retention strategies, as they exhibit a higher churn rate

rev_growth_yoy vs. Churn

- Similar distributions for both churned and retained customers.
- Might not be a strong predictor of churn on its own.

coupon_used_for_payment vs. Churn

- Churned customers tend to use slightly fewer coupons on average.
- Some customers with very high coupon usage also churn, but the difference isn't stark.

cashback vs. Churn

- Cashback amounts are similar for both churned and non-churned customers.
- Some high cashback recipients still churn, suggesting cashback alone isn't a retention guarantee.

Churn Vs Categorical Columns

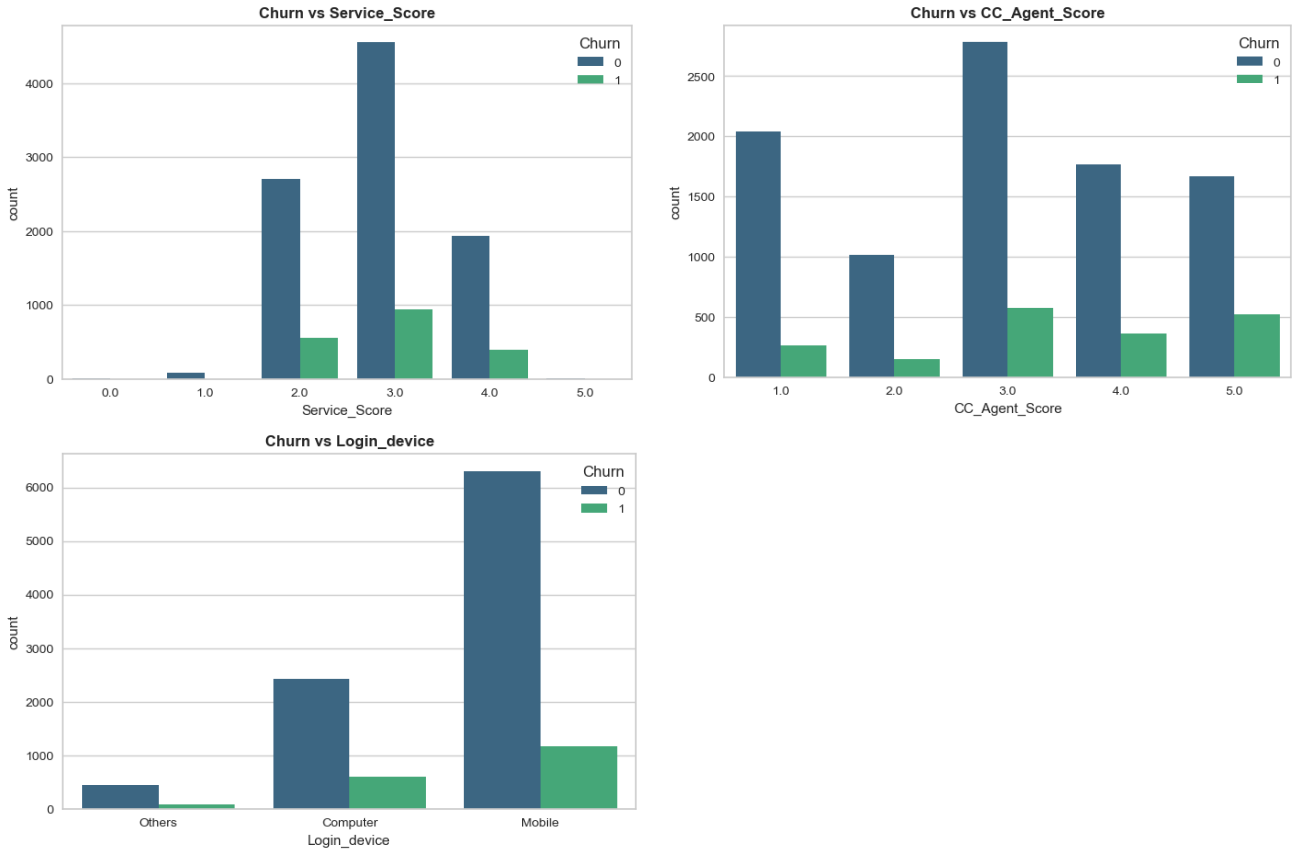


Figure 27 - Bivariate Analysis - Churn vs Categorical Columns

Churn vs. Service Score

- **Most customers (both churned and retained) have a service score of 3**, indicating this is the most common experience level.
- **Higher churn among lower service scores (2-3)**, suggesting dissatisfaction with service quality.
- **Customers with high service scores (4+) churn less**, reinforcing that strong service leads to retention.

Actionable Insight:

- Focus on improving service for customers with scores 2-3 to reduce churn.
- Investigate root causes of low ratings (e.g., support delays, product issues).
- Maintain high service standards for top-rated customers to sustain loyalty.

Churn vs. CC Agent Score

- **Most customers rate agents as "3"**, the neutral/moderate experience level.
- **Churn occurs across all agent scores**, but is **highest for scores 3 and 5**—indicating potential dissatisfaction even at high ratings.
- **Lower churn for scores 2 & 4**, suggesting these interactions may be more stable.

Actionable Insight:

- Investigate why **scores 3 and 5** correlate with higher churn—possible inconsistencies in service quality.
- Enhance agent training, especially for handling complex issues (score 5 may reflect unresolved complaints).
- Implement sentiment analysis on customer feedback to identify pain points.

Churn vs. Login Device

- **Mobile users dominate the platform but churn more** than computer/other device users.
- **Lower churn among desktop users**, possibly due to better usability or engagement.
- **Potential drivers of mobile churn:**
  Poor app performance, UI/UX friction, or unmet expectations.
  Differences in usage patterns (e.g., mobile users may expect faster, seamless experiences).

Actionable Insight:

- Optimize mobile experience (e.g., app speed, navigation, notifications).
- Conduct surveys to understand mobile users' pain points.
- Test personalized retention strategies (e.g., targeted offers, in-app support).

Multivariate Analysis

Revenue Analysis Across City Tiers & Account Segments

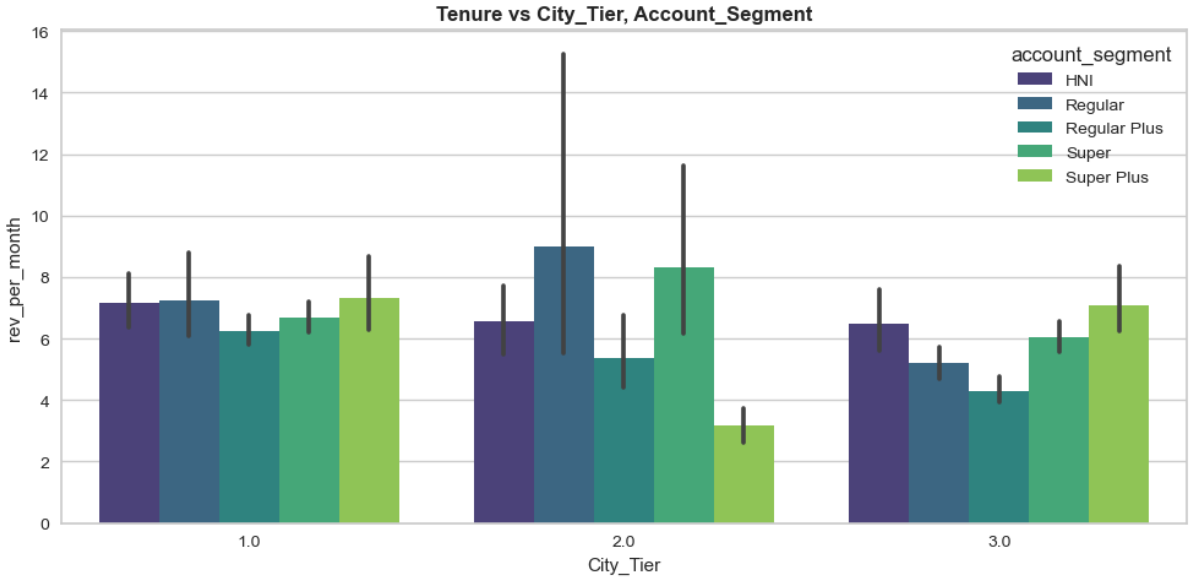


Figure 28 - Multivariate Analysis - Tenure Vs City_tier, account_segment

Revenue varies significantly by city tier and account segment.

- **City Tier 1** shows stable revenue across all segments with minor fluctuations.
- **City Tier 2** has the highest revenue from "Super Plus" customers, while "Regular Plus" lags behind.
- **City Tier 3** generates the lowest revenue, especially in "Regular" and "Regular Plus" segments.
- **"Super Plus" and "Super" segments drive the highest revenue in all city tiers.**
- **"Regular Plus" and "Regular" segments underperform**, with noticeable variations by location.

- **HNI customers** maintain steady revenue but don't lead in earnings.
- **City Tier 2 displays the widest revenue gap**—"Super Plus" excels while "Regular Plus" struggles.
- **City Tier 3 consistently underperforms**, suggesting regional economic challenges.
- **Lower-revenue segments (Regular/Regular Plus) and Tier 3 cities are at higher churn risk.**
- **"Super Plus" and "Super" customers are high-value—retention efforts should prioritize them.**
- **Geo-targeted strategies can optimize revenue—Tier 2 for growth, Tier 3 for stabilization.**

Key Actions:

- **Enhance Tier 3 engagement** with localized offers to boost revenue.
- **Upsell "Regular Plus" in Tier 2** to premium tiers for better retention.
- **Investigate churn drivers** in low-revenue groups to reduce attrition.

Revenue Analysis Across Tenure & Account Segments

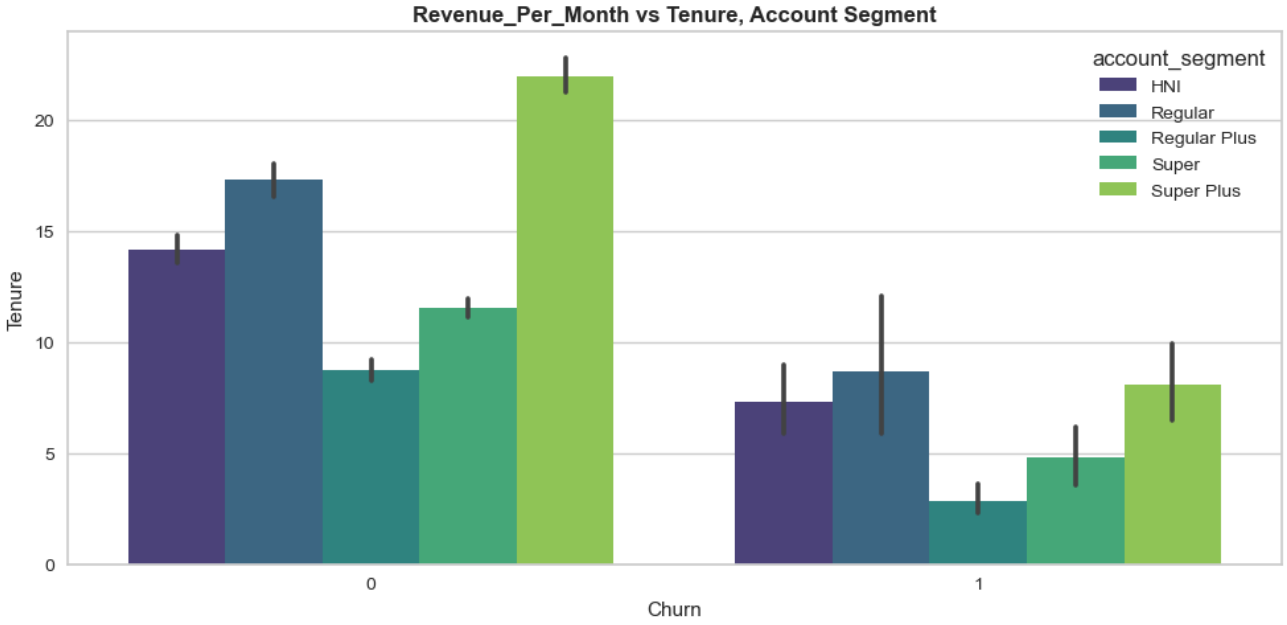


Figure 29 - Multivariate Analysis - Rev_per_month Vs Tenure, account_segment

Tenure Trends by Churn Status

- Retained customers consistently show higher tenure across all account segments.
- Churned customers exhibit significantly lower tenure, indicating newer customers are more likely to leave.

Account Segment Influence

- Non-churn group: "Super Plus" has the longest tenure, followed by "Regular" and "HNI."
- Churn group: "Super Plus" and "Super" maintain slightly higher tenure than other segments, but still below retention benchmarks.
- "Regular Plus" customers display low tenure in both groups, suggesting higher volatility.

Revenue and Tenure Relationship

- Longer-tenured customers generate higher revenue, especially in "Super Plus" and "Regular" segments.
- Churned customers have both lower tenure and lower revenue, indicating short-term engagement.

Strategic Insights for Retention

- **Early-stage customers are high-risk:** Implement **onboarding incentives** and **early engagement programs** to reduce churn.

- **Premium segments ("Super Plus," "Super," "HNI") retain longer but still churn:** Focus on **long-term loyalty programs** and **exclusive benefits**.
- **"Regular Plus" needs proactive retention:** Offer **personalized incentives** to improve engagement and tenure.
- **Prioritize high-value, long-tenure customers** in retention strategies to maximize revenue stability.

Customer Distribution & Marital Status Trends

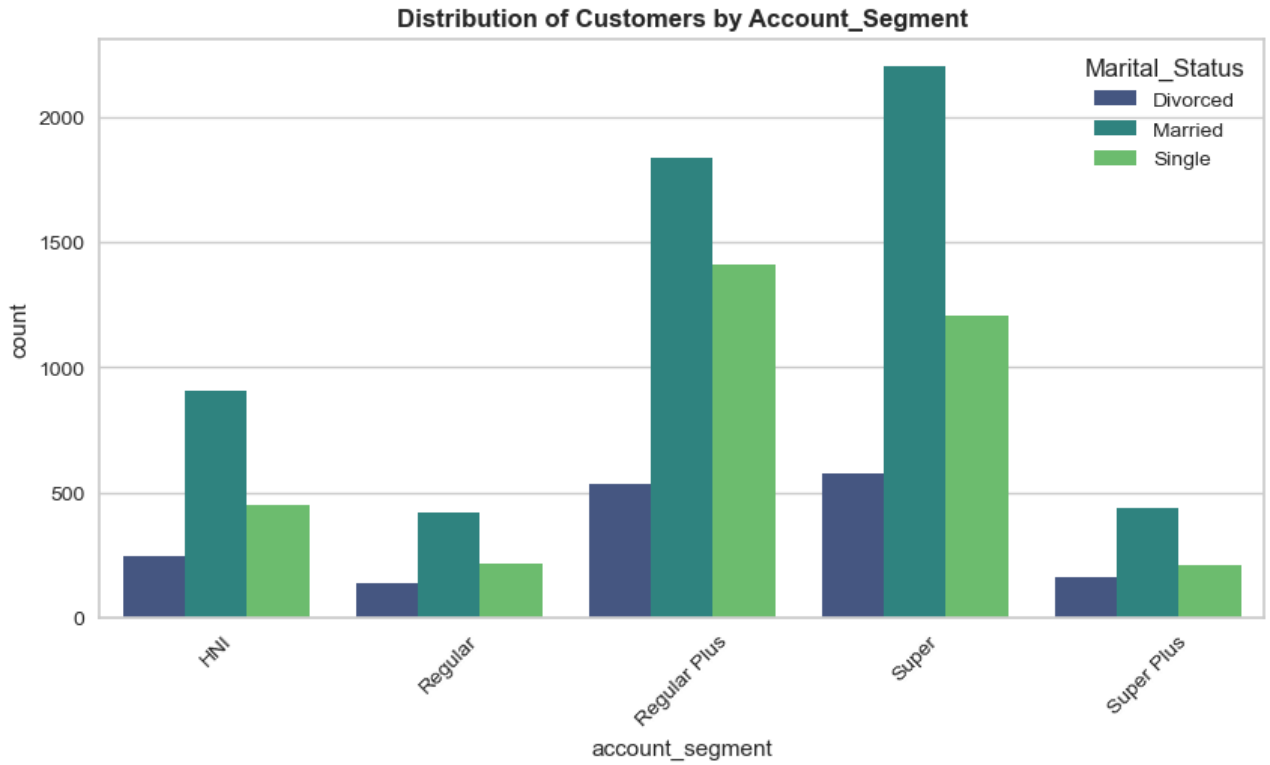


Figure 30 - Multivariate Analysis - Customers vs account_segment

General Customer Distribution

- **Highest Customer Concentration:** "Super" and "Regular Plus" segments dominate in customer numbers.
- **Lowest Customer Base:** "HNI" and "Super Plus" segments have the fewest customers.

Marital Status Breakdown

- **Married customers** form the **largest group across all segments**.
- **Single customers** are the **second-largest segment**, followed by a small proportion of **Divorced customers**.

Segment-Specific Insights

- **Super Segment:**
  - Highest customer volume, primarily **Married**, with **Single** as the next largest group.
- **Regular Plus Segment:**
  - Similar marital distribution to "Super," with **Married** leading and **Single** following.
- **HNI & Super Plus Segments:**
  - Smaller customer base but still **Married-dominated**, mirroring overall trends.
- **Regular Segment:**
  - Most **balanced distribution** but with fewer customers than "Super" and "Regular Plus."

Strategic Implications for Retention & Engagement

- **Married Customers (Primary Target):**
  - Focus on **family-oriented benefits**, long-term value, and stability-driven messaging.

- **Single Customers (High Potential):**
  - Particularly prominent in **"Super" and "Regular Plus"**—ideal for **flexible plans, lifestyle perks, and short-term incentives**.
- **Divorced Customers (Niche but Important):**
  - Though small in number, they exist in all segments—**personalized financial or support-based offers** may improve retention.

Missing Values in Dataset

The below Table represents the Missing values in each Features in the Dataset.

Feature	Missing_Values	%_Missing_Values
rev_per_month	791	7.02%
cashback	473	4.2%
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%

Table 14 - Missing Values Check

Data Preprocessing Details

1. Feature Encoding

Variable	Encoding Type	Description
Payment_Method	One-Hot Encoding	Created binary columns: Payment_COD, Payment_UPI, Payment_Wallet, etc.
Tenure_Group	Ordinal Encoding	Low = 0, Medium = 1, High = 2
Marital_Status	One-Hot Encoding	Binary columns: Marital_Status_Single, Marital_Status_Married
City_Tier	Ordinal Encoding	Tier 1 = 0, Tier 2 = 1, Tier 3 = 2

Table 15 - Feature Encoding

2. Feature Scaling

Scaler Used	Reason
StandardScaler	Normally distributed; scaled to mean=0, std=1 for SVM convergence.

Table 16 - Feature Scaling

SMOTE Oversampling Results

Class Distribution Before SMOTE:

- Non-Churn: 1214 (83.2%)
- Churn: 5992 (16.8%)

Class Distribution After SMOTE:

- Non-Churn: 5992 (50%)
- Churn: 5992 (50%)

Initial Model Building - Detailed

Logistic Regression (statsmodel)

```
                           Logit Regression Results
==============================================================================
Dep. Variable:                  Churn   No. Observations:
7206
Model:                          Logit   Df Residuals:
7182
Method:                           MLE   Df Model:
23
Date:                Tue, 01 Apr 2025   Pseudo R-squ.:
0.3394
Time:                        19:51:58   Log-Likelihood:                -
2158.6
converged:                       True   LL-Null:                       -
3267.6
Covariance Type:            nonrobust   LLR p-value:
0.000
==============================================================================
                               coef    std err          z      P>|z|
[0.025      0.975]
------------------------------------------------------------------------------
const                       -2.5404      0.058    -43.507      0.000      -
2.655      -2.426
City_Tier                    0.2555      0.044      5.819      0.000
0.169       0.342
CC_Contacted_LY             -0.7358      0.141     -5.216      0.000      -
1.012      -0.459
Service_Score                0.0222      0.043      0.510      0.610      -
0.063       0.107
Account_user_count           0.3034      0.044      6.901      0.000
0.217       0.390
account_segment             -0.4383      0.045     -9.768      0.000      -
0.526      -0.350
```

```
CC_Agent_Score              0.3828      0.041      9.412      0.000
0.303       0.463
rev_per_month               1.2534      0.135      9.288      0.000
0.989       1.518
rev_growth_yoy             -0.1152      0.041     -2.821      0.005      -
0.195      -0.035
coupon_used_for_payment     1.5011      0.171      8.759      0.000
1.165       1.837
Day_Since_CC_connect       -0.3887      0.051     -7.636      0.000      -
0.488      -0.289
cashback                    0.3682      0.127      2.902      0.004
0.120       0.617
Loyalty_Score              -2.2778      0.311     -7.321      0.000      -
2.888      -1.668
Tenure_Group                0.8902      0.053     16.667      0.000
0.785       0.995
Payment_COD                 0.2125      0.039      5.402      0.000
0.135       0.290
Payment_DC                  0.0713      0.047      1.530      0.126      -
0.020       0.163
Payment_UPI                -0.0257      0.043     -0.603      0.546      -
0.109       0.058
Payment_Wallet              0.1371      0.047      2.910      0.004
0.045       0.229
Gender_Male                 0.1633      0.040      4.096      0.000
0.085       0.241
Marital_Status_Married     -0.1048      0.059     -1.773      0.076      -
0.221       0.011
Marital_Status_Single       0.3699      0.056      6.601      0.000
0.260       0.480
Login_device_Mobile        -0.2838      0.041     -6.981      0.000      -
0.364      -0.204
Login_device_Others        -0.1443      0.042     -3.428      0.001      -
0.227      -0.062
Complain_ly_1.0             0.7124      0.037     19.350      0.000
0.640       0.785
==============================================================================
```

- **Pseudo R-squared:** 0.3394 (pseudo R-squared value indicates the goodness of fit (closer to 1 is better)).
- **LLR p-value:** 0.000 The p-value of the Likelihood Ratio Test. The model is significantly better than the null model (p-value < 0.05).
- **City_Tier, Account_user_count, CC_Agent_Score, rev_per_month, coupon_used_for_payment, cashback, Tenure_Group, Payment_COD, Payment_Wallet, Gender_Male, Marital_Status_Single, Complain_ly_1.0:** Positive coefficients increase the likelihood of the target being 1 (e.g., churn).
- **CC_Contacted_LY, account_segment, rev_growth_yoy, Day_Since_CC_connect, Loyalty_Score, Login_device_Mobile, Login_device_Others :** These are having Negative coefficients decrease the likelihood.
- Negative values of the coefficient shows that probability of customer being a defaulter decreases with the increase of corresponding attribute value.
- Positive values of the coefficient show that that probability of customer being a defaulter increases with the increase of corresponding attribute value.
- p-value of a variable indicates if the variable is significant or not. If we consider the significance level to be 0.05 (5%), then any variable with a p-value less than 0.05 would be considered significant.
- But these variables might contain multicollinearity, which will affect the p-values.

- We will have to remove multicollinearity from the data to get reliable coefficients and p-values.
- There are different ways of detecting (or testing) multi-collinearity, one such way is the Variation Inflation Factor.
- **Variance Inflation factor**: Variance inflation factors measure the inflation in the variances of the regression coefficients estimates due to collinearity that exist among the predictors. It is a measure of how much the variance of the estimated regression coefficient βk is "inflated" by the existence of correlation among the predictor variables in the model.
- General Rule of thumb: If VIF is 1 then there is no correlation among the kth predictor and the remaining predictor variables, and hence the variance of β̂k is not inflated at all. Whereas if VIF exceeds 5, we say there is moderate VIF and if it is 10 or exceeding 10, it shows signs of high multi-collinearity. But the purpose of the analysis should dictate which threshold to use.

Multi Collinearity Check for Each features

```
No. of Features in X_train: 24

Index(['const', 'City_Tier', 'CC_Contacted_LY', 'Service_Score',
       'Account_user_count', 'account_segment', 'CC_Agent_Score',
       'rev_per_month', 'rev_growth_yoy', 'coupon_used_for_payment',
       'Day_Since_CC_connect', 'cashback', 'Loyalty_Score', 'Tenure_Group',
       'Payment_COD', 'Payment_DC', 'Payment_UPI', 'Payment_Wallet',
       'Gender_Male', 'Marital_Status_Married', 'Marital_Status_Single',
       'Login_device_Mobile', 'Login_device_Others', 'Complain_ly_1.0'],
      dtype='object')
```

Checking VIF for each Features:

feature	VIF
Loyalty_Score	17.111444
coupon_used_for_payment	5.888786
CC_Contacted_LY	4.283434
rev_per_month	4.08359
cashback	3.609709
Marital_Status_Single	2.17528
Marital_Status_Married	2.151073
Payment_Wallet	1.599868
Payment_DC	1.373348
City_Tier	1.372658
Day_Since_CC_connect	1.291693
Payment_COD	1.186906
Service_Score	1.182889
Payment_UPI	1.156801
Account_user_count	1.150753
Tenure_Group	1.150003
Login_device_Mobile	1.133422
Login_device_Others	1.128356
rev_growth_yoy	1.024399
account_segment	1.02419
CC_Agent_Score	1.017373

Gender_Male	1.012549
Complain_ly_1.0	1.011868
const	1

Table 17 - VIF Check before SMOTE

Removing features with High VIF values one at a time

1. Removing Loyalty_Score with VIF = 17.111444

Re-fitting the model on the updated X_train

```
                           Logit Regression Results
==============================================================================
Dep. Variable:                  Churn   No. Observations:
7206
Model:                          Logit   Df Residuals:
7183
Method:                           MLE   Df Model:
22
Date:                Tue, 01 Apr 2025   Pseudo R-squ.:
0.3261
Time:                        19:51:58   Log-Likelihood:                -
2202.1
converged:                       True   LL-Null:                       -
3267.6
Covariance Type:            nonrobust   LLR p-value:
0.000
==============================================================================
                               coef    std err          z      P>|z|
[0.025      0.975]
------------------------------------------------------------------------------
const                       -2.5196      0.058    -43.587      0.000       -
2.633      -2.406
City_Tier                    0.2749      0.043      6.339      0.000
0.190       0.360
CC_Contacted_LY              0.2586      0.038      6.744      0.000
0.183       0.334
Service_Score                0.0422      0.043      0.983      0.326       -
0.042       0.126
Account_user_count           0.3011      0.044      6.900      0.000
0.216       0.387
account_segment             -0.4455      0.044    -10.114      0.000       -
0.532      -0.359
CC_Agent_Score               0.3806      0.040      9.482      0.000
0.302       0.459
rev_per_month                0.3106      0.039      7.954      0.000
0.234       0.387
rev_growth_yoy              -0.1211      0.040     -3.000      0.003       -
0.200      -0.042
coupon_used_for_payment      0.2959      0.047      6.359      0.000
0.205       0.387
Day_Since_CC_connect        -0.3968      0.050     -7.898      0.000       -
0.495      -0.298
cashback                    -0.4652      0.058     -7.990      0.000       -
0.579      -0.351
Tenure_Group                 1.1025      0.047     23.214      0.000
1.009       1.196
Payment_COD                  0.2079      0.039      5.391      0.000
0.132       0.283
```

```
Payment_DC                  0.0772      0.046      1.677      0.093      -
0.013       0.167
Payment_UPI                -0.0073      0.042     -0.174      0.861      -
0.090       0.075
Payment_Wallet              0.1262      0.047      2.711      0.007
0.035       0.217
Gender_Male                 0.1765      0.039      4.484      0.000
0.099       0.254
Marital_Status_Married     -0.1035      0.058     -1.771      0.077      -
0.218       0.011
Marital_Status_Single       0.3738      0.055      6.747      0.000
0.265       0.482
Login_device_Mobile        -0.2901      0.040     -7.226      0.000      -
0.369      -0.211
Login_device_Others        -0.1461      0.042     -3.504      0.000      -
0.228      -0.064
Complain_ly_1.0             0.7116      0.036     19.581      0.000
0.640       0.783
```

Checking VIF for each Features:

feature	VIF
Marital_Status_Single	2.174993
Marital_Status_Married	2.151058
Payment_Wallet	1.599867
Payment_DC	1.372853
City_Tier	1.371342
Day_Since_CC_connect	1.290158
coupon_used_for_payment	1.253505
cashback	1.247629
Payment_COD	1.186897
Service_Score	1.179781
Payment_UPI	1.155807
Account_user_count	1.150729
Login_device_Mobile	1.133368
Login_device_Others	1.128278
Tenure_Group	1.118496
rev_per_month	1.069248
rev_growth_yoy	1.02418
account_segment	1.021223
CC_Contacted_LY	1.02084
CC_Agent_Score	1.016809
Gender_Male	1.011981
Complain_ly_1.0	1.011837
const	1

Table 18 - Checking VIF before SMOTE 2

Removing features with p value >0.05 which are insignificant

- Let's remove the insignificant features (p-value>0.05).

- **Service_Score, Payment_DC, Payment_UPI and Marital_Status_Married** have a high p-value. So, they are not significant and we'll drop them.
- But sometimes p-values change after dropping a variable. So, we'll not drop all variables at once.
- Instead, we will do the following repeatedly using a loop:
- Build a model, check the p-values of the variables, and drop the column with the highest p-value.
- Create a new model without the dropped feature, check the p-values of the variables, and drop the column with the highest p-value.
- Repeat the above two steps till there are no columns with p-value > 0.05.
- The above process can also be done manually by picking one variable at a time that has a high p-value, dropping it, and building a model again. But that might be a little tedious and using a loop will be more efficient.

```
No. of Significant Features: 19

['const', 'City_Tier', 'CC_Contacted_LY', 'Account_user_count',
'account_segment', 'CC_Agent_Score', 'rev_per_month', 'rev_growth_yoy',
'coupon_used_for_payment', 'Day_Since_CC_connect', 'cashback',
'Tenure_Group', 'Payment_COD', 'Payment_Wallet', 'Gender_Male',
'Marital_Status_Single', 'Login_device_Mobile', 'Login_device_Others',
'Complain_ly_1.0']
```

The above columns are the significant columns

```
No. of Features in updated Train dataset: 19
```

Re-fitting the model on the updated X_train

```
                           Logit Regression Results
==============================================================================
Dep. Variable:                  Churn   No. Observations:
7206
Model:                          Logit   Df Residuals:
7187
Method:                           MLE   Df Model:
18
Date:                Tue, 01 Apr 2025   Pseudo R-squ.:
0.3249
Time:                        19:51:59   Log-Likelihood:                -
2206.1
converged:                       True   LL-Null:                       -
3267.6
Covariance Type:            nonrobust   LLR p-value:
0.000
==============================================================================
                             coef    std err          z      P>|z|
[0.025      0.975]
------------------------------------------------------------------------------
const                     -2.5169      0.058    -43.599      0.000      -
2.630      -2.404
City_Tier                  0.2756      0.043      6.390      0.000
0.191       0.360
CC_Contacted_LY            0.2595      0.038      6.786      0.000
0.185       0.334
```

```
Account_user_count          0.3123      0.042      7.478      0.000
0.230       0.394
account_segment            -0.4454      0.044    -10.146      0.000      -
0.531      -0.359
CC_Agent_Score              0.3799      0.040      9.508      0.000
0.302       0.458
rev_per_month               0.3124      0.039      8.026      0.000
0.236       0.389
rev_growth_yoy             -0.1189      0.040     -2.962      0.003      -
0.198      -0.040
coupon_used_for_payment     0.3042      0.046      6.622      0.000
0.214       0.394
Day_Since_CC_connect       -0.3976      0.050     -7.932      0.000      -
0.496      -0.299
cashback                   -0.4558      0.057     -7.957      0.000      -
0.568      -0.344
Tenure_Group                1.1009      0.047     23.223      0.000
1.008       1.194
Payment_COD                 0.1844      0.035      5.245      0.000
0.116       0.253
Payment_Wallet              0.1019      0.043      2.379      0.017
0.018       0.186
Gender_Male                 0.1744      0.039      4.443      0.000
0.097       0.251
Marital_Status_Single       0.4503      0.037     12.191      0.000
0.378       0.523
Login_device_Mobile        -0.2901      0.040     -7.241      0.000      -
0.369      -0.212
Login_device_Others        -0.1452      0.042     -3.490      0.000      -
0.227      -0.064
Complain_ly_1.0             0.7126      0.036     19.632      0.000
0.641       0.784
==============================================================================
```

Converting coefficients to odds

- The coefficients of the logistic regression model are in terms of log(odd), to find the odds we have to take the exponential of the coefficients.
- Therefore, **odds = exp(b)**
- The percentage change in odds is given as **odds = (exp(b) - 1) * 100**

feature	Odds	Change_odd%
const	0.080713	-91.928698
City_Tier	1.317361	31.736133
CC_Contacted_LY	1.296239	29.623933
Account_user_count	1.366571	36.657063
account_segment	0.640596	-35.940398
CC_Agent_Score	1.462075	46.207548
rev_per_month	1.366669	36.666896
rev_growth_yoy	0.887871	-11.212908
coupon_used_for_payment	1.35557	35.556997
Day_Since_CC_connect	0.671923	-32.80771
cashback	0.633924	-36.607648
Tenure_Group	3.006736	200.673576
Payment_COD	1.202526	20.252577

Payment_Wallet	1.107293	10.729314
Gender_Male	1.190506	19.050613
Marital_Status_Single	1.568739	56.873903
Login_device_Mobile	0.748206	-25.179408
Login_device_Others	0.864875	-13.512548
Complain_ly_1.0	2.039333	103.933346

Table 19 - Converting Coefficients to Logodds

Coefficient interpretations

- **City_Tier**: Holding all other features constant a unit change in City_Tier will increase the odds of a customer being churned by 1.31 times or a 31.73% increase in odds.
- **CC_Contacted_LY**: Holding all other features constant a unit change in CC_Contacted_LY will increase the odds of a customer being churned by 1.29 times or a 29.62% increase in the odds.
- **Account_user_count**: Holding all other features constant a unit change in no. of users in an account will increase the odds of a customer being churned by 1.37 times or a 36.66% increase in the odds.
- **account_segment**: Holding all other features constant a unit change in account_segment will decrease the odds of a customer being churned by 0.64 times or a 35.94% decrease in the odds.
- Interpretation for other attributes can be done similarly.

Checking Logistic Regression model performance on the training set

Logistic Regression - Performance Metrics

Model	Accuracy	Recall	Precision	F1-Score	ROC-AUC
LogisticRegression(statsmodel)	0.882182	0.456343	0.745626	0.566173	0.865096

Table 20 - Log Regression - Performance on Train set

Logistic Regression - Confusion Matrix

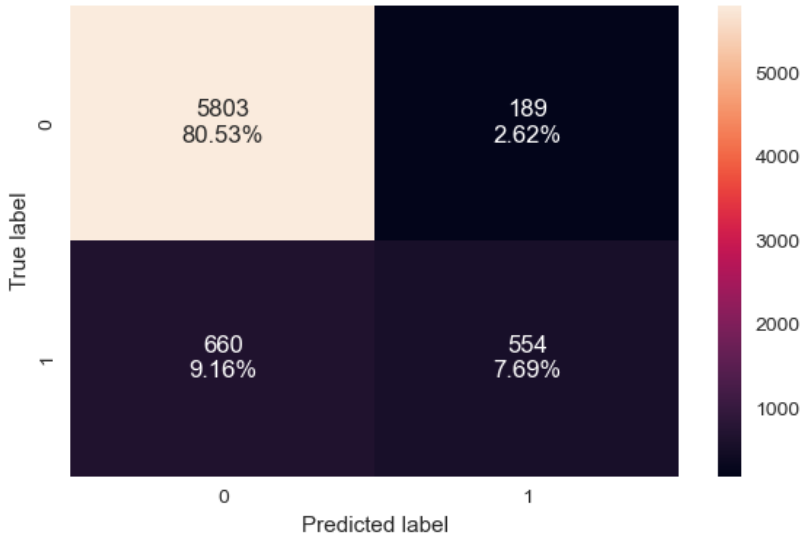


Figure 31 - Log Regression - Confusion Matrix on Train Set

Logistic Regression - ROC-AUC Curve

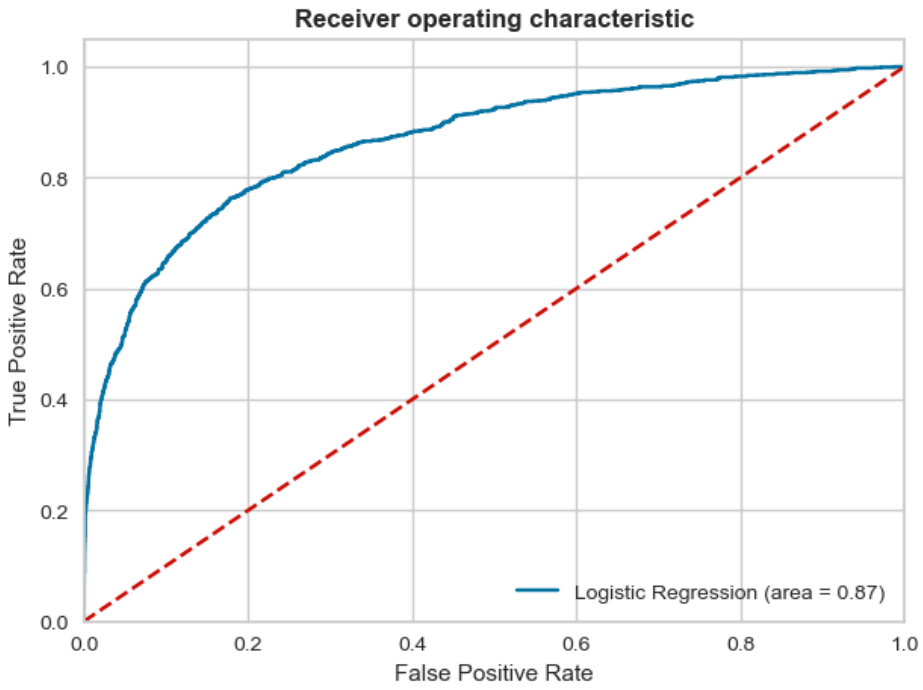


Figure 32 - Log Regression ROC-AUC Curve

Model has an AUC of 0.87, which is relatively good. This means, overall, the model is fairly good at ranking actual churners above non-churners.

Optimal threshold from ROC-AUC Curve : 0.206

Logistic Regression - Performance Metrics with Optimal Threshold

Model	Accuracy	Recall	Precision	F1-Score	ROC-AUC
LogisticRegression(statsmodel)	0.81113	0.762768	0.463232	0.576408	0.865096

Table 21 - - Log Regression - Performance with Optimal Threshold

Logistic Regression - Confusion Matrix with Optimal Threshold

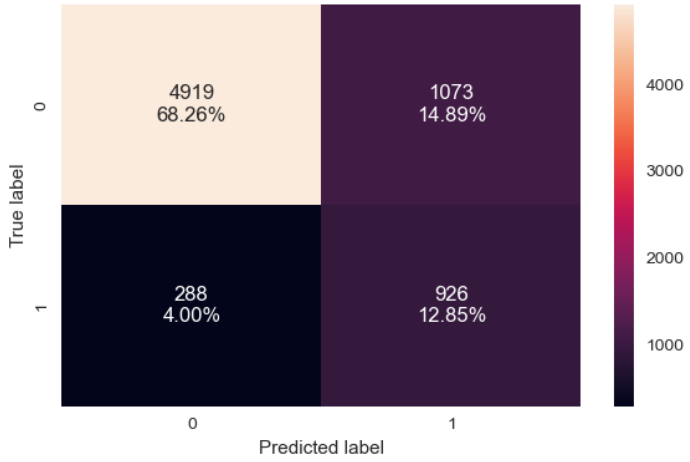


Figure 33 - Log Regression - Performance with Optimal Threshold

- Model performance has improved significantly.
- Model is giving a recall of 0.76 as compared to initial model which was giving a recall of 0.46.
- Precision has decreased from 0.75 to 0.46.

Logistic Regression - ROC-AUC Curve with Optimal Threshold

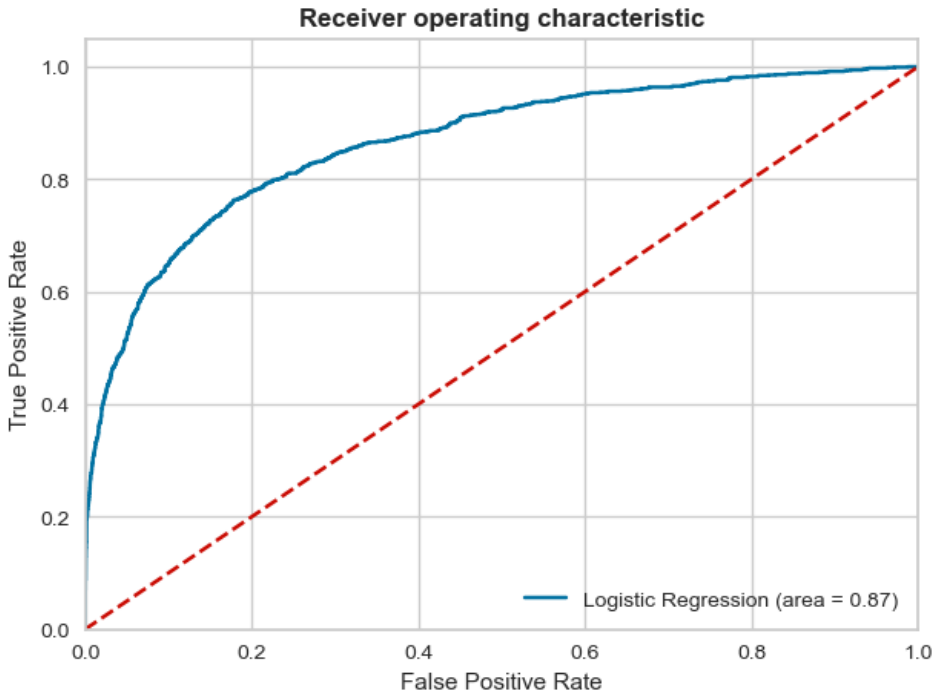


Figure 34 - Log Regression - ROC-AUC with Optimal Threshold

Checking Precision-Recall curve to see if we can find a better threshold

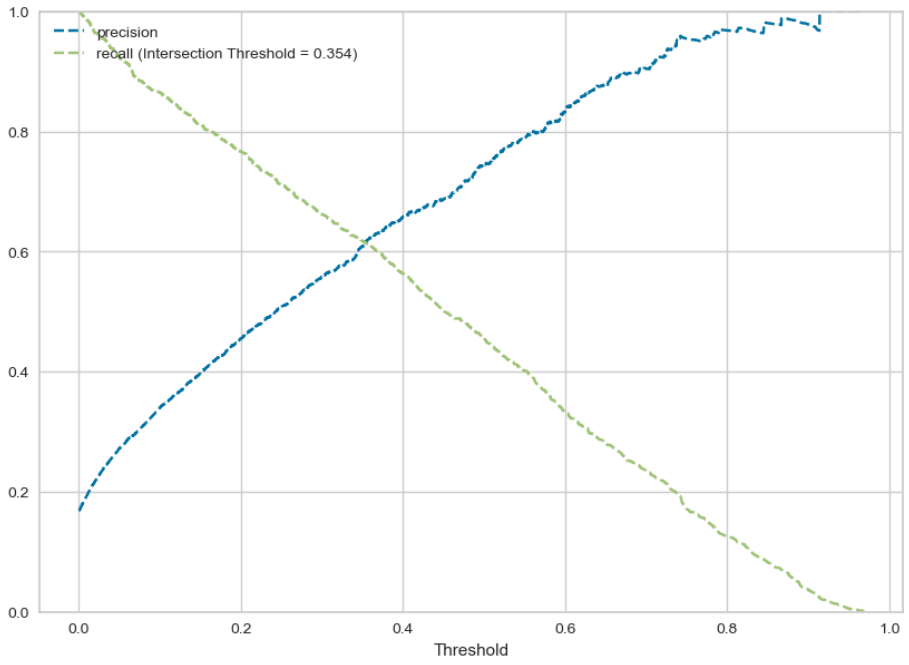


Figure 35 - Log Regression - Precision-Recall Curve for Best threshold

```
Threshold where Precision ~ Recall: 0.354
```

At threshold around 0.35 we will get equal precision and recall but taking a step back and selecting value around 0.30 will provide a higher recall and a good precision.

Logistic Regression - Performance Metrics with Best Threshold

Model	Accuracy	Recall	Precision	F1-Score	ROC-AUC
LogisticRegression(statsmodel)	0.854704	0.66145	0.558026	0.605352	0.865096

Figure 36 - Log Regression - Performance with Best Threshold

Logistic Regression - Confusion Matrix with Best Threshold

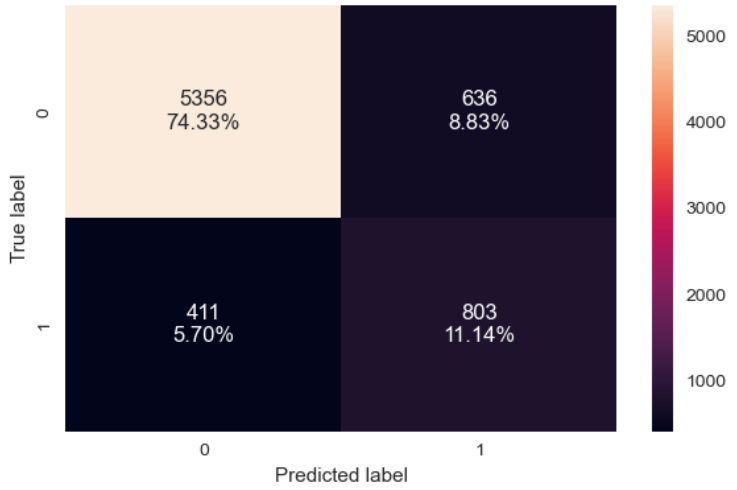


Figure 37 - Log Regression - Confusion Matrix with Best threshold

Checking Logistic Regression model performance on the Validation set

Logistic Regression - Performance Metrics

Model	Accuracy	Recall	Precision	F1-Score	ROC-AUC
LogisticRegression(statsmodel)	0.846837	0.669967	0.53562	0.595308	0.863944

Table 22 - Log Regression - Performance on Validation set

Logistic Regression - Confusion Matrix

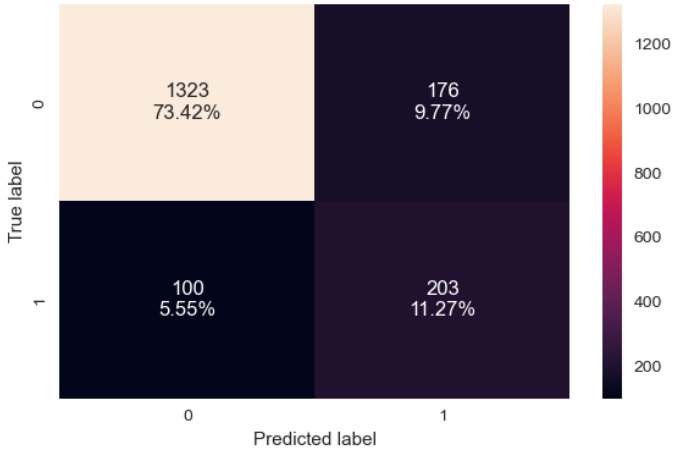


Table 23 - Log Regression - Confusion Matrix on Validation set

Logistic Regression - ROC-AUC Curve

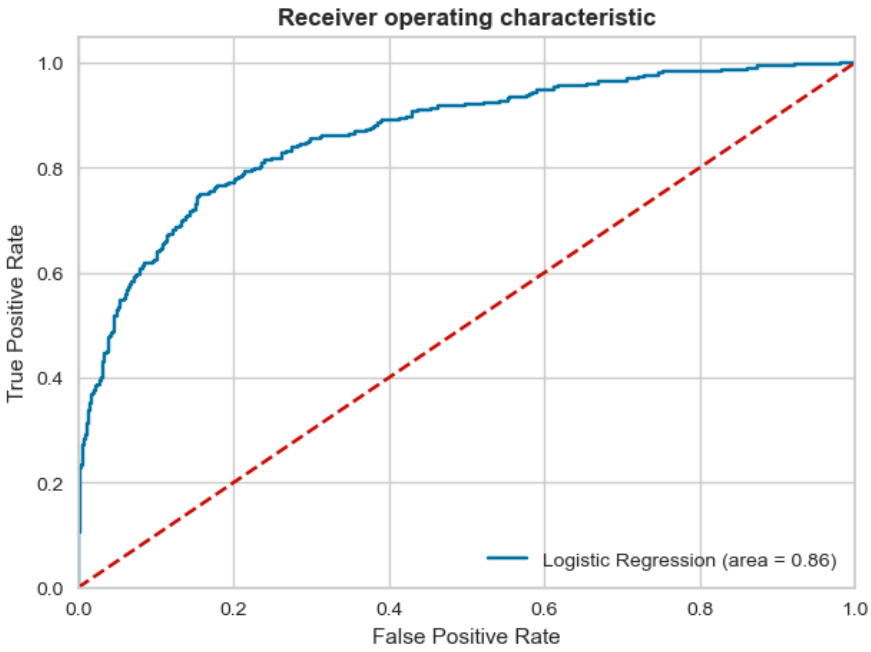


Figure 38 - Log Regression - ROC-AUC

Initial Model Building (Other Models)

Checking various Models performance on the Training set

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
DecisionTree	1	1	1	1	1
SVM	0.796281	0.44169	0.792422	0.567217	0.869773
RandomForest	1	1	1	1	1
AdaBoost	0.889259	0.747031	0.518122	0.611868	0.888891
GBM	0.913128	0.849169	0.588962	0.695525	0.942098
XGBoost	0.999445	1	0.996705	0.99835	1
LogisticRegression(statsmodel)	0.854704	0.558026	0.66145	0.605352	0.865096

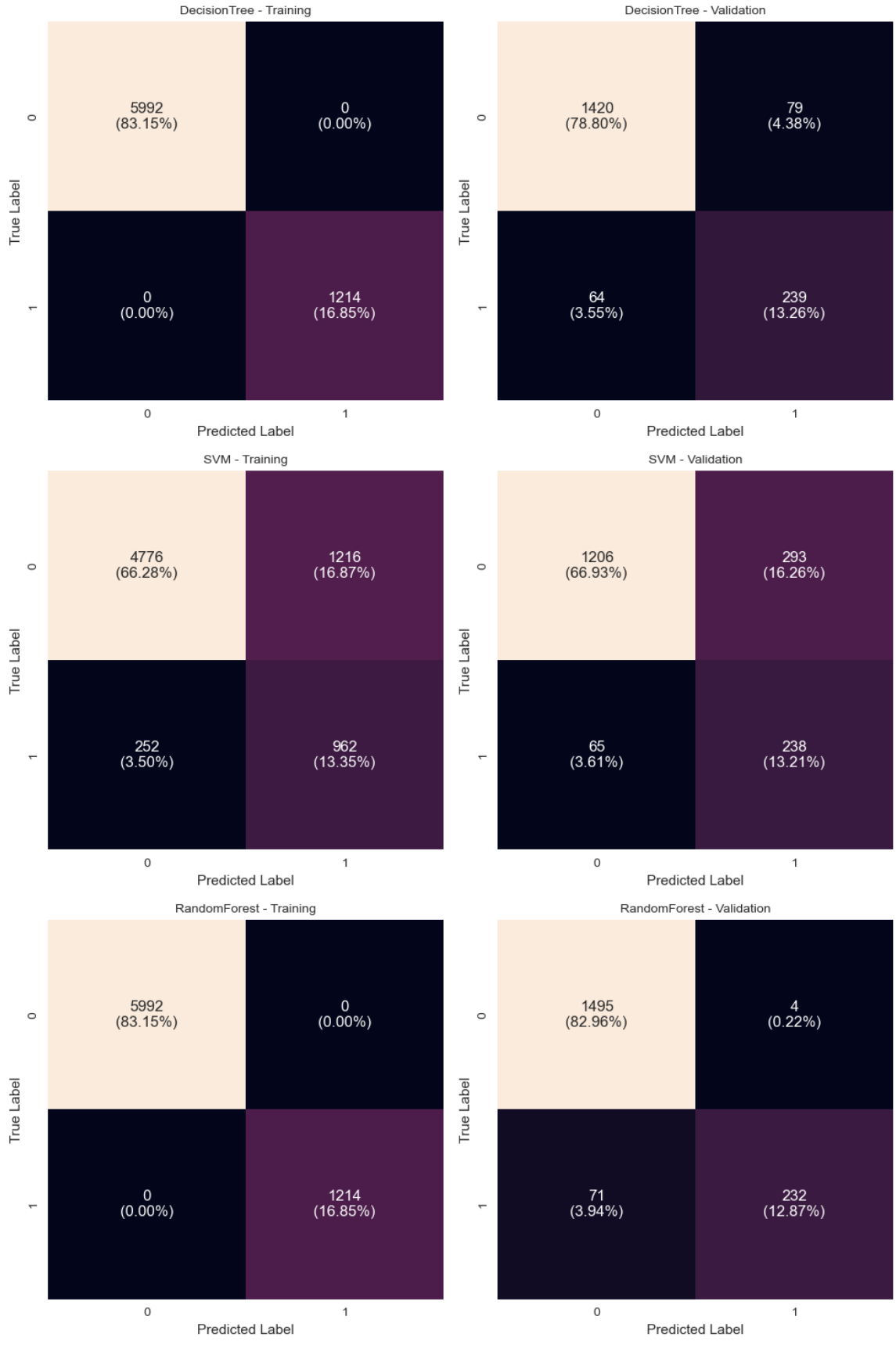
Table 24 - Initial Models Performance on Train Set

Checking various Models performance on the Validation set

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
DecisionTree	0.920644	0.751572	0.788779	0.769726	0.868039
SVM	0.801332	0.448211	0.785479	0.570743	0.868189
RandomForest	0.95838	0.983051	0.765677	0.860853	0.991365
AdaBoost	0.883463	0.704846	0.528053	0.603774	0.889043
GBM	0.897336	0.775701	0.547855	0.642166	0.927184
XGBoost	0.954495	0.907749	0.811881	0.857143	0.986021
LogisticRegression(statsmodel)	0.854704	0.558026	0.66145	0.605352	0.865096

Table 25 - Initial Models Performance on Validation Set

Confusion Matrix for Other models for Training and Validation Sets



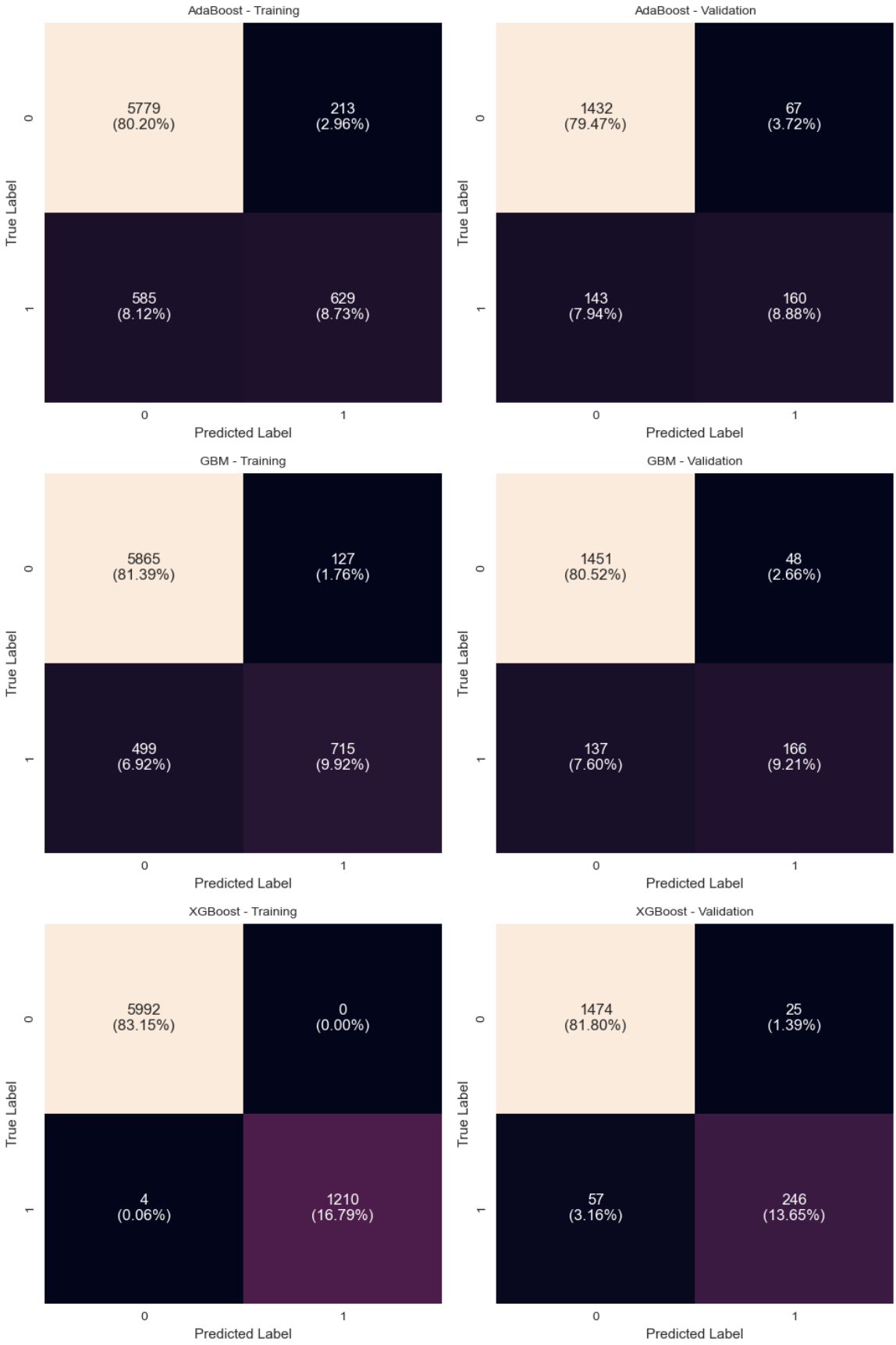


Figure 39 - Confusion Matrices for Various Models

Performance Comparison for All the Models

Model	Training Score	Validation Score	Difference
RandomForest	1	0.765677	0.234323
DecisionTree	1	0.788779	0.211221
XGBoost	0.996705	0.811881	0.184824
GBM	0.588962	0.547855	0.041107
SVM	0.792422	0.785479	0.006943
LogisticRegression(statsmodel)	0.66145	0.669967	-0.00852
AdaBoost	0.518122	0.528053	-0.00993

Table 26 - Performance Comparison for Various Models

From the above results:

- XGBoost shows the best validation performance (0.8118) while maintaining relatively good generalization (difference of 0.185), but it still shows some overfitting tendency.
- DecisionTree and RandomForest have near-perfect training scores (Accuracy, Precision, Recall, and F1-score all close to 1). However, their validation recall scores drop significantly (e.g., DecisionTree from 1.00 → 0.788 and RandomForest from 1.00 → 0.766), indicating potential overfitting.
- Out of all these, SVM shows excellent consistency between training and validation (difference < 0.01)
- AdaBoost and GBM have relatively smaller gaps between training and validation scores, indicating better generalization. But the recall scores for these are near to 0.50 which will cause random predictions.
- Comparatively, GBM has a reasonable validation recall (0.5479) and a moderate training score (0.5889), showing a balance between learning and avoiding overfitting.

In churn prediction, recall is critical as we need to minimize false negatives (missed churn cases). Models with high recall and low overfitting risk are preferable.

Recommended Models for Further Tuning:

- **Primary Choice:** XGBoost (best validation performance)
- **Secondary Choice:** SVM (most stable performance)

XGBoost

- High recall on training data (0.9967) but a notable drop in validation (0.8118).
- Still performs better than DecisionTree/RandomForest in validation.
- With proper regularization (tree depth, learning rate tuning), it can be optimized.

Support Vector Machine (SVM)

- Balanced training-validation recall (0.792422 → 0.785479).
- Less prone to overfitting compared to DecisionTree/RandomForest.
- Can improve with hyperparameter tuning (e.g., learning rate, number of estimators).

We will focus on XGB and SVM for its balance and generalization ability. And we will consider XGB if tuned properly to reduce overfitting.

Model Interpretation

Now let's interpret the best-performing models using feature importance

Feature	Importance
Tenure_Group	0.22929
Complain_ly_1.0	0.109944
CC_Agent_Score	0.046421
Marital_Status_Single	0.045495
account_segment	0.04498
City_Tier	0.04307
Login_device_Mobile	0.039551
Payment_UPI	0.038953
Day_Since_CC_connect	0.036227
Payment_Wallet	0.03567
rev_per_month	0.033325
Payment_COD	0.032825
Payment_DC	0.030788
Account_user_count	0.028471
Marital_Status_Married	0.027904
coupon_used_for_payment	0.027311
rev_growth_yoy	0.027066
CC_Contacted_LY	0.02626
Gender_Male	0.022931
cashback	0.020699
Service_Score	0.01965
Loyalty_Score	0.0174
Login_device_Others	0.015768

Table 27 - Model Interpretation

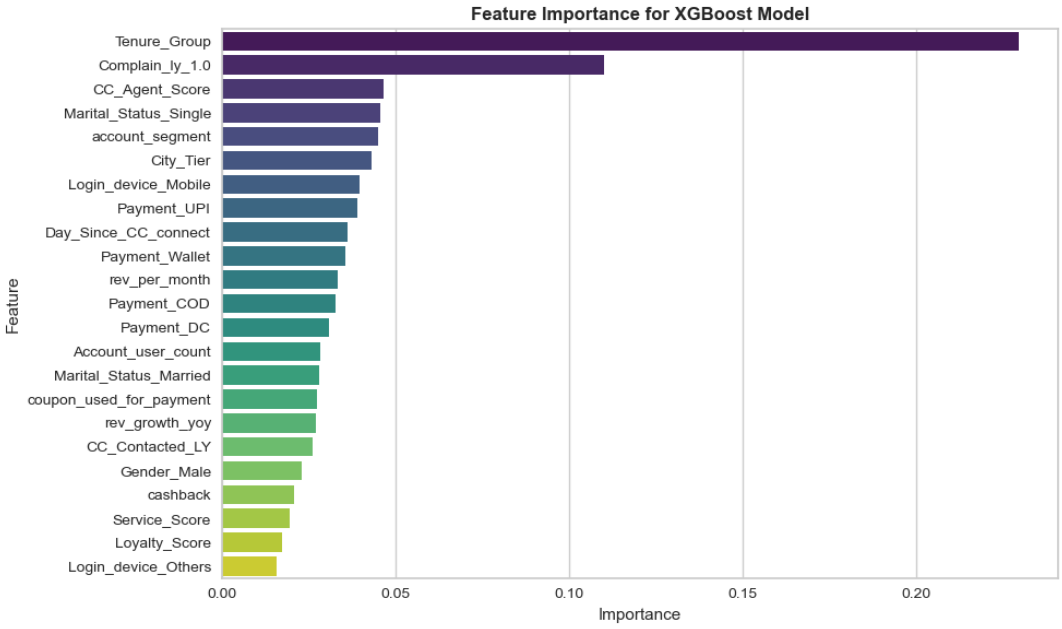


Figure 40 - Model Interpretation

Top 5 Most Important Features:

- **Tenure_Group**: Customer tenure is the most influential factor in predicting churn. This suggests that how long a customer has been associated with the company significantly impacts churn probability.
- **Complain_ly_1.0**: Customers who complained in the last year are highly likely to churn, indicating dissatisfaction.
- **CC_Agent_Score**: The quality of customer care interactions (e.g., service ratings) plays a critical role in retention.
- **Marital_Status_Single**: Single customers may have a higher churn rate compared to married ones.
- **account_segment**: Account segment is also an important factor in predicting churn.

Building Logistic Regression after SMOTE

For the next round of VIF (Multicollinearity Check) after applying SMOTE, we should use the resampled dataset but only with the initially selected significant features.

```
                           Logit Regression Results
==============================================================================
Dep. Variable:                  Churn   No. Observations:
11984
Model:                          Logit   Df Residuals:
11965
Method:                           MLE   Df Model:
18
Date:                Tue, 01 Apr 2025   Pseudo R-squ.:
0.3470
Time:                        19:52:53   Log-Likelihood:                -
5424.6
converged:                       True   LL-Null:                       -
8306.7
Covariance Type:            nonrobust   LLR p-value:
0.000
==============================================================================
                               coef    std err          z      P>|z|
[0.025      0.975]
------------------------------------------------------------------------------
----------------
const                     -0.8850      0.030    -29.973      0.000      -
0.943      -0.827
City_Tier                  0.2836      0.028     10.283      0.000
0.230       0.338
CC_Contacted_LY            0.2436      0.024      9.977      0.000
0.196       0.291
Account_user_count         0.2668      0.026     10.344      0.000
0.216       0.317
account_segment           -0.5259      0.028    -19.039      0.000      -
0.580      -0.472
CC_Agent_Score             0.3693      0.025     14.611      0.000
0.320       0.419
rev_per_month              0.3164      0.025     12.517      0.000
0.267       0.366
rev_growth_yoy            -0.1649      0.025     -6.565      0.000      -
0.214      -0.116
```

```
coupon_used_for_payment      0.3312       0.029      11.384       0.000
0.274       0.388
Day_Since_CC_connect        -0.3779       0.030     -12.423       0.000       -
0.438      -0.318
cashback                    -0.3852       0.034     -11.428       0.000       -
0.451      -0.319
Tenure_Group                 1.0127       0.027      37.539       0.000
0.960       1.066
Payment_COD                  0.2378       0.023      10.265       0.000
0.192       0.283
Payment_Wallet               0.1166       0.028       4.219       0.000
0.062       0.171
Gender_Male                  0.1433       0.025       5.828       0.000
0.095       0.191
Marital_Status_Single        0.4396       0.023      18.854       0.000
0.394       0.485
Login_device_Mobile         -0.2176       0.025      -8.652       0.000       -
0.267      -0.168
Login_device_Others         -0.1278       0.026      -4.931       0.000       -
0.179      -0.077
Complain_ly_1.0              0.6691       0.023      28.886       0.000
0.624       0.715
==============================================================================
```

Multi Collinearity Check for Each features after SMOTE

feature	VIF
Day_Since_CC_connect	1.408148
Payment_Wallet	1.390741
City_Tier	1.379222
cashback	1.379155
coupon_used_for_payment	1.293285
Tenure_Group	1.232902
const	1.174913
Login_device_Mobile	1.126745
Login_device_Others	1.120814
Account_user_count	1.056922
rev_per_month	1.055298
Marital_Status_Single	1.05494
account_segment	1.038782
Payment_COD	1.035374
CC_Contacted_LY	1.032281
Complain_ly_1.0	1.031798
rev_growth_yoy	1.024113
CC_Agent_Score	1.0191
Gender_Male	1.018211

Table 28 - VIF Check for various Features after SMOTE

So there are no features with VIF >5 indicating there is no Multicollinearity observed in the resampled data.

Also there are no features in the result summary with p value > 0.05.

Coefficient Interpretation for Each features after SMOTE

feature	Odds	Change_odd%
const	0.412716	-58.728366
City_Tier	1.327858	32.785786
CC_Contacted_LY	1.275843	27.584348
Account_user_count	1.305727	30.572746
account_segment	0.591032	-40.896794
CC_Agent_Score	1.44678	44.678011
rev_per_month	1.372226	37.222649
rev_growth_yoy	0.847944	-15.205558
coupon_used_for_payment	1.392597	39.259734
Day_Since_CC_connect	0.685276	-31.472398
cashback	0.680334	-31.966648
Tenure_Group	2.75316	175.316042
Payment_COD	1.268439	26.843918
Payment_Wallet	1.123665	12.366509
Gender_Male	1.154021	15.402062
Marital_Status_Single	1.552145	55.214472
Login_device_Mobile	0.804449	-19.555063
Login_device_Others	0.880012	-11.998833
Complain_ly_1.0	1.952508	95.250812

Table 29 - Coefficient Interpretation after SMOTE

Checking Logistic Regression model performance on the training set after SMOTE

Logistic Regression - Performance Metrics

Model	Accuracy	Recall	Precision	F1-Score	ROC-AUC
LogisticRegression(statsmodel)	0.796646	0.812083	0.787761	0.799737	0.870985

Table 30 - Log Regression Performance Metrics after SMOTE

Logistic Regression - Confusion Matrix

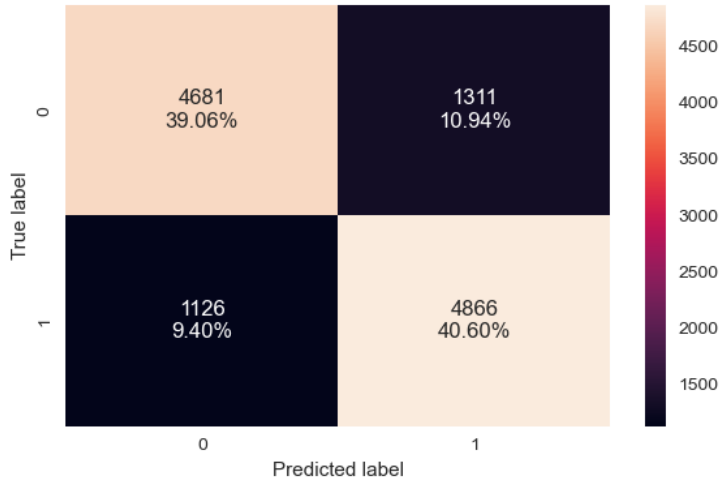


Figure 41 - Confusion Matrix for Log Regression after SMOTE

Logistic Regression - ROC-AUC Curve

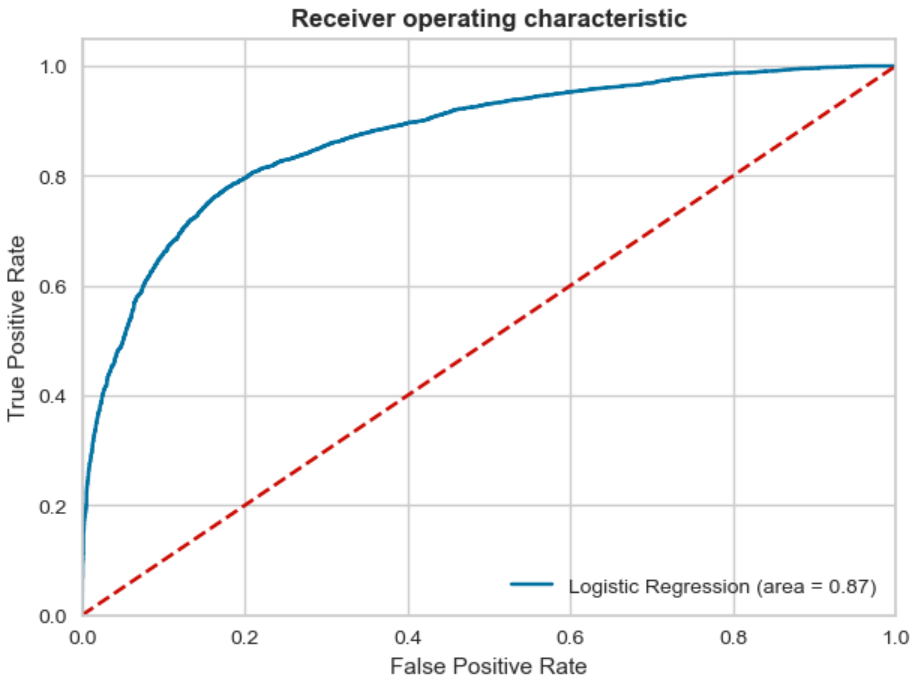


Figure 42 - Log Regression ROC-AUC Curve after SMOTE

Optimal threshold from ROC-AUC Curve : 0.543

Logistic Regression - Performance Metrics after SMOTE with Optimal Threshold

Model	Accuracy	Recall	Precision	F1-Score	ROC-AUC
LogisticRegression(statsmodel)	0.800067	0.785214	0.809254	0.797052	0.870985

Table 31 - Log Regression Performance Metrics after SMOTE with Optimal Threshold

Logistic Regression - Confusion Matrix after SMOTE with Optimal Threshold

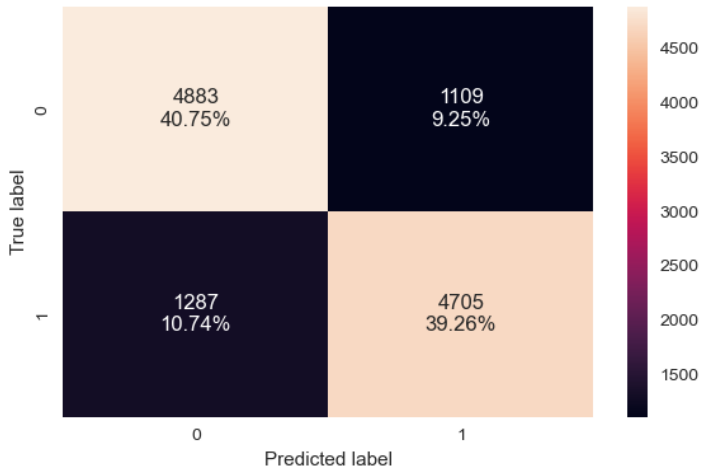


Table 32 - Log Regression Confusion Matrix after SMOTE with Optimal Threshold

Logistic Regression - ROC-AUC Curve after SMOTE with Optimal Threshold

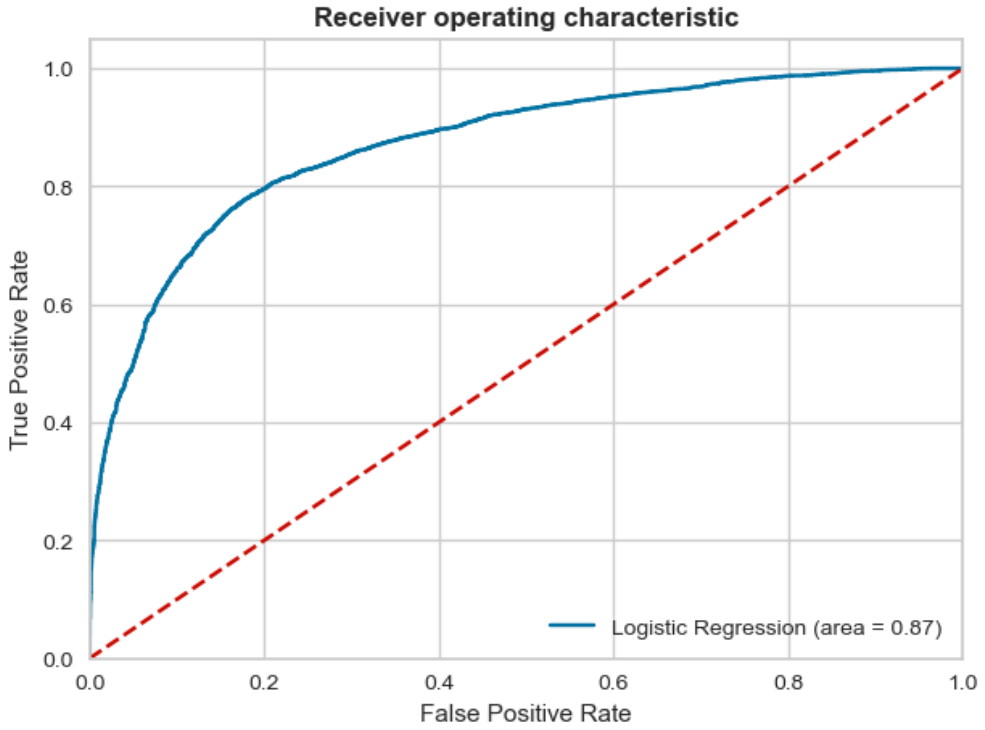


Figure 43 - Log Regression ROC-AUC after SMOTE with Optimal threshold

Checking Precision-Recall curve to see if we can find a better threshold

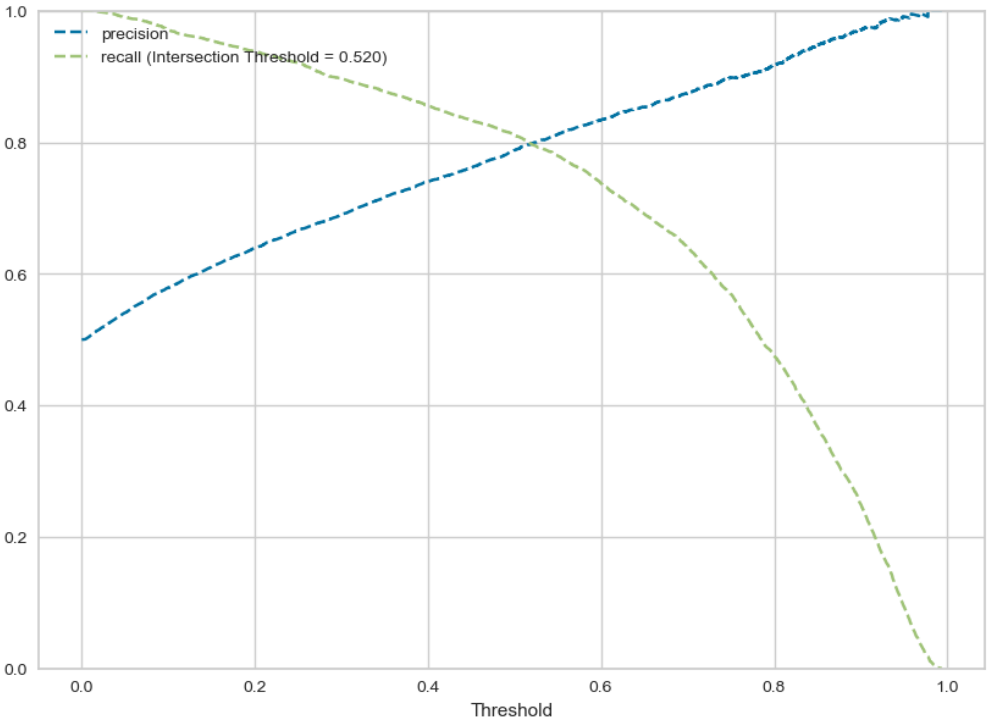


Figure 44 - Precision-Recall Curve after SMOTE

```
Threshold where Precision ~ Recall: 0.52
```

At threshold around 0.520 we will get equal precision and recall but taking a step back and selecting value around 0.50 will provide a higher recall and a good precision.

Logistic Regression - Performance Metrics after SMOTE with Best Threshold

Model	Accuracy	Recall	Precision	F1-Score	ROC-AUC
LogisticRegression(statsmodel)	0.796646	0.812083	0.787761	0.799737	0.870985

Table 33 - Log Regression Performance after SMOTE with Optimal Threshold

Logistic Regression - Confusion Matrix after SMOTE with Best Threshold

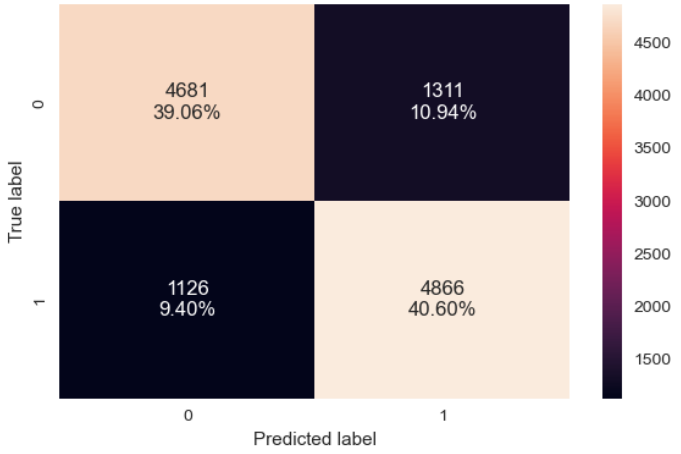


Figure 45 - Log Regression Confusion Matrix after SMOTE with best threshold

Checking Logistic Regression model performance after SMOTE on the Validation set

Logistic Regression - Performance Metrics

Model	Accuracy	Recall	Precision	F1-Score	ROC-AUC
LogisticRegression(statsmodel)	0.786349	0.805281	0.42807	0.558992	0.865688

Table 34 - Log Regression Performance after SMOTE on validation set

Logistic Regression - Confusion Matrix

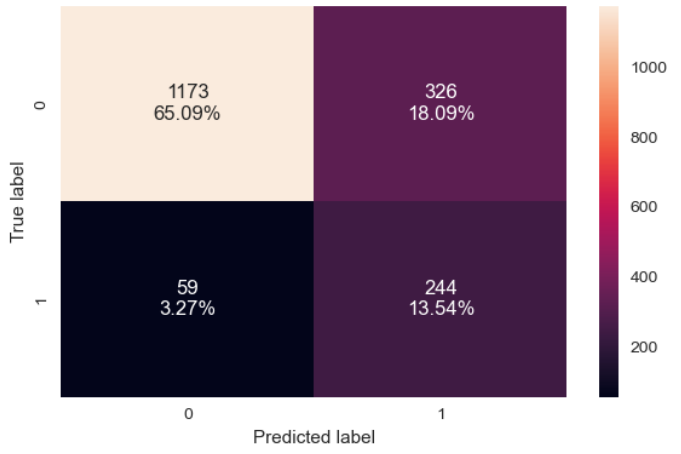


Figure 46 - Log Regression Confusion Matrix after SMOTE on validation set

Logistic Regression - ROC-AUC Curve

Figure 47 - Log Regression - ROC-AUC Curve after SMOTE

Building other models after SMOTE

Checking various Models performance with SMOTE on the Training set

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
DecisionTree	1	1	1	1	1
SVM	0.804322	0.801554	0.808912	0.805216	0.878351
RandomForest	1	1	1	1	1
AdaBoost	0.875167	0.878962	0.87016	0.874539	0.947459
GBM	0.927654	0.934248	0.92006	0.9271	0.979072
XGBoost	0.999666	1	0.999332	0.999666	0.999998
LogisticRegression(statsmodel)	0.796646	0.787761	0.812083	0.799737	0.870985

Table 35 - Various Model Performance with SMOTE on Training set

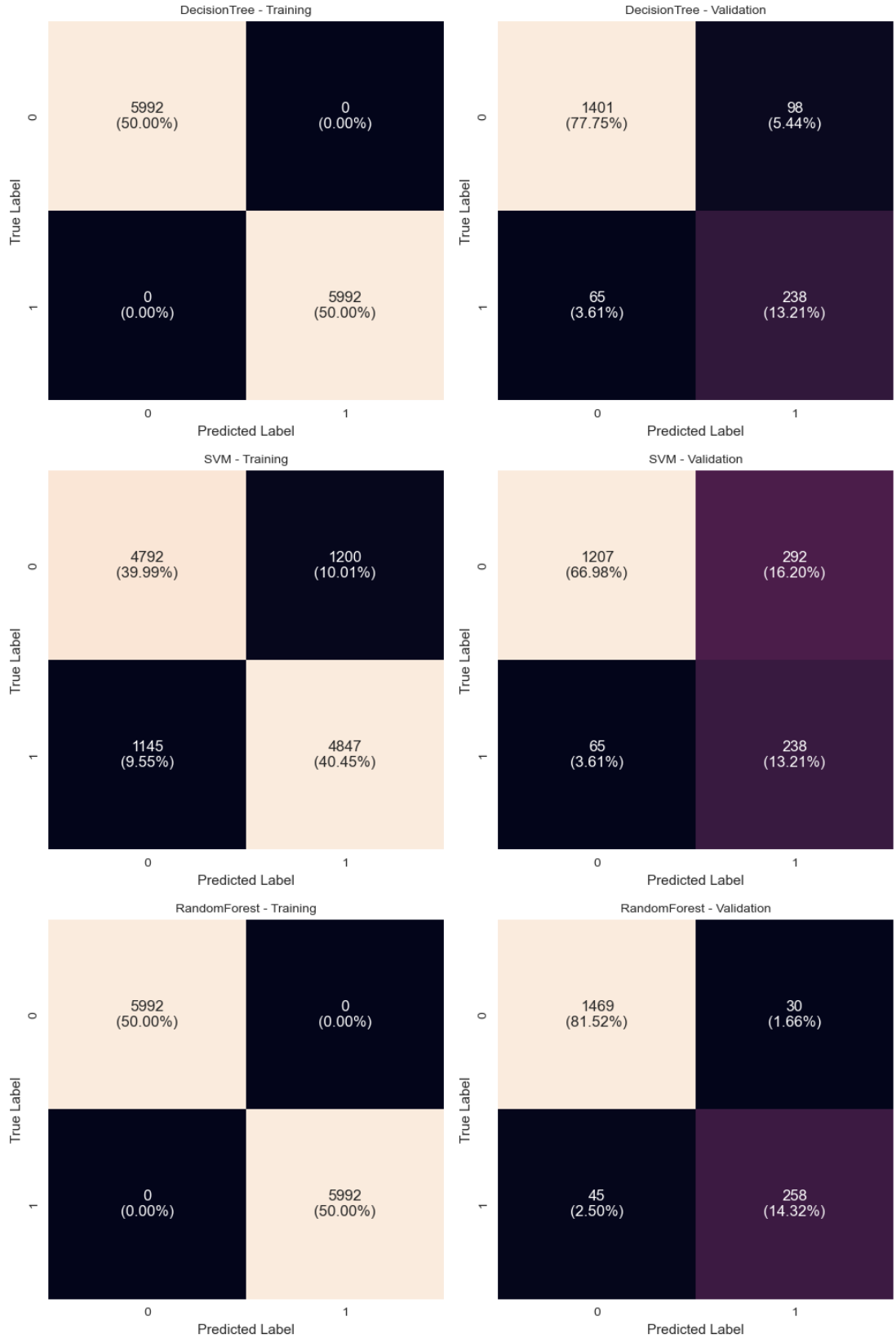
Checking various Models performance with SMOTE on the Validation set

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
DecisionTree	0.909545	0.708333	0.785479	0.744914	0.860051
SVM	0.801887	0.449057	0.785479	0.571429	0.8689
RandomForest	0.95838	0.895833	0.851485	0.873096	0.986667
AdaBoost	0.846282	0.530374	0.749175	0.621067	0.884551
GBM	0.885683	0.645646	0.709571	0.676101	0.916926

XGBoost	0.955605	0.885813	0.844884	0.864865	0.983593
LogisticRegression(statsmodel)	0.796646	0.787761	0.812083	0.799737	0.870985

Table 36 - Various Model Performance with SMOTE on Validation Set

Confusion Matrix for Other models for Training and Validation Sets after SMOTE



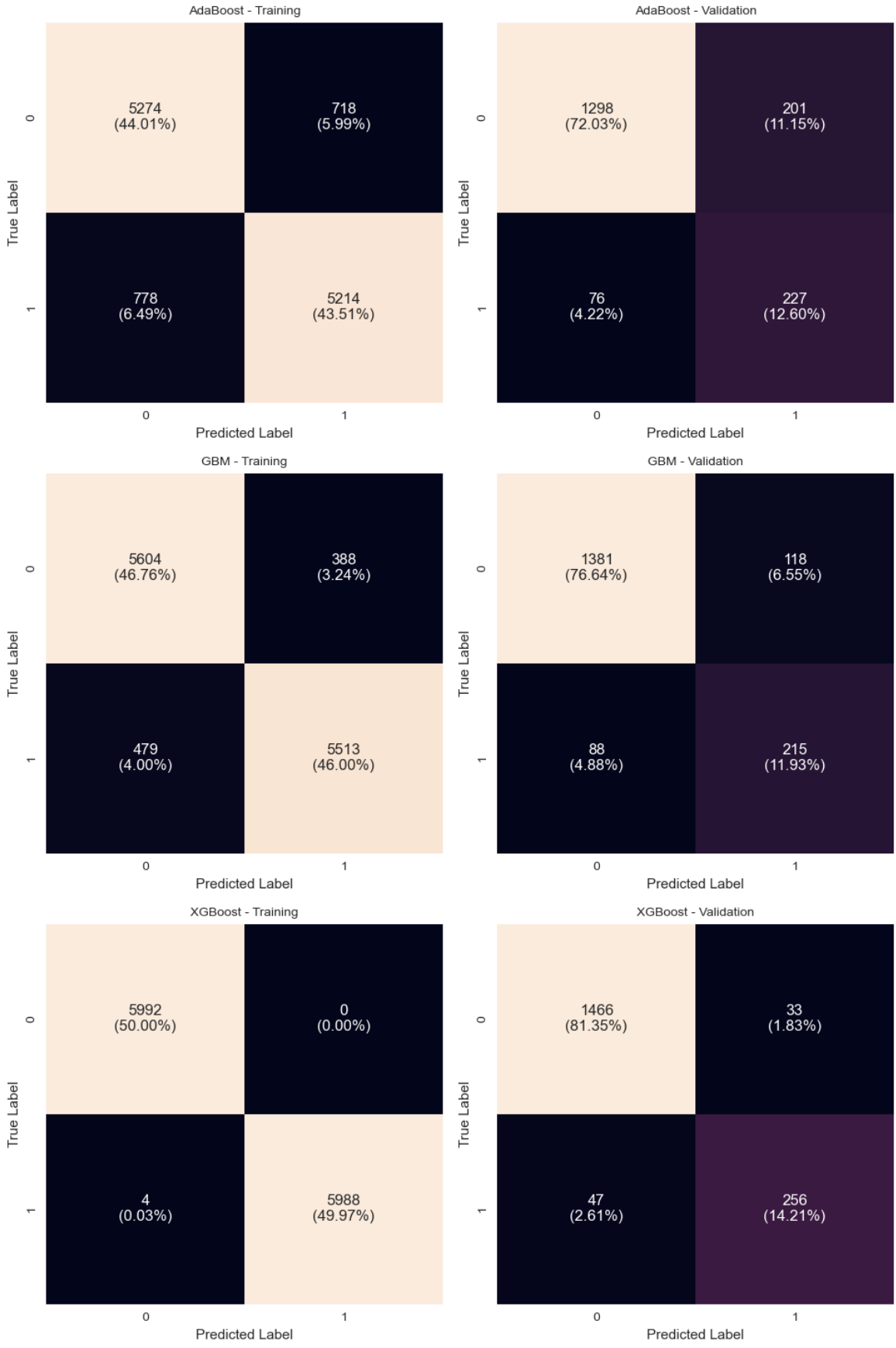


Figure 48 - Confusion Matrix for Other models for Training and Validation Sets after SMOTE

Performance Comparison for All the Models after SMOTE

Model	Training Score	Validation Score	Difference
DecisionTree	1	0.785479	0.214521
GBM	0.92006	0.709571	0.210489
XGBoost	0.999332	0.844884	0.154448
RandomForest	1	0.851485	0.148515
AdaBoost	0.87016	0.749175	0.120985
SVM	0.808912	0.785479	0.023433
LogisticRegression(statsmodel)	0.812083	0.805281	0.006802

Table 37 - Performance Comparision for Various models after SMOTE

Before Oversampling

- XGBoost (0.811881) and SVM (0.785479) have the highest validation scores. The differences between training and validation scores are small, indicating less overfitting.

After Oversampling

- **RandomForest (0.851485) and XGBoost (0.844884)** have the highest validation scores.
- RandomForest has better Generalization: It shows a smaller performance gap between training (1.000) and validation (0.851) compared to XGBoost (0.999 vs 0.845)
- Both models show more Balanced Metrics :
  - Recall: 0.851 (good at identifying churners)
  - Precision: 0.896 (not flagging too many non-churners)
- DecisionTree, RandomForest, and XGBoost have high training scores (1.000000 or close) but significantly lower validation scores, indicating overfitting.

So the best Trade-off is RandomForest and XGBoost show the best balance between training and validation performance.

- **Primary Model Choice: RandomForest**
- **Secondary Model Choice: XGBoost**

Model Hyperparameters

1. Support Vector Machine (SVM)

```
Best Parameters for SVM: {'kernel': 'rbf', 'gamma': 'scale',
'class_weight': 'balanced', 'C': 10}

Best Recall Score for SVM: 0.9958274081125024
```

2. Logistic Regression (Baseline)

```
Best Parameters: {'solver': 'liblinear', 'penalty': 'l1', 'C': 0.01}

Best Recall Score: 0.818257702230991
```

3. RandomForest

```
Best Parameters: {'n_estimators': 100, 'min_samples_split': 2,
'min_samples_leaf': 1, 'max_features': 'log2', 'max_depth': 30}

Best Recall Score: 0.973957151271023
```

4.XGBoost

```
Best Parameters: {'subsample': 0.8, 'reg_lambda': 0.5, 'reg_alpha': 0.5,
'n_estimators': 50, 'max_depth': 7, 'learning_rate': 0.1, 'gamma': 0.1,
'colsample_bytree': 1.0}

Best Recall Score: 0.9330569018979367
```

Model Performance Comparison

Performance Comparison of Tuned models on Training Set

Model	Accuracy	Precision	Recall	F1-Score
RandomForest	1	1	1	1
XGBoost	0.977887	0.977648	0.978138	0.977893
LogisticRegression	0.790971	0.77723	0.815754	0.796026
SVM	0.994993	0.992364	0.997664	0.995007

Table 38 - Performance Comparison of Tuned models on Training Set

Performance Comparison of Tuned models on Validation Set

Model	Accuracy	Precision	Recall	F1-Score
RandomForest	0.95727	0.900709	0.838284	0.868376
XGBoost	0.924528	0.761755	0.80198	0.78135
LogisticRegression	0.779134	0.418803	0.808581	0.551802
SVC	0.963374	0.881029	0.90429	0.892508

Table 39 - Performance Comparison of Tuned models on Validation Set

Based on these Performance metrics, SVC (Support Vector Classifier) model is the most optimal choice.

Model Interpretation of Best Model - Detailed

- **Highest Recall on Validation Set**

  Recall: 0.9043 (highest among all models)

  Since recall is crucial in churn prediction (to minimize false negatives and correctly identify churned customers), SVC performs best.

- **Good Precision & F1-Score**

  SVC has a high precision (0.8810) and F1-score (0.8925), showing a good balance between precision and recall.

RandomForest and XGBoost have decent recall but lower F1-scores, indicating slightly lower overall performance.

So we use SVC as the final model for customer churn prediction.

Testing the Predictive Model Against the Test Set

Model	Accuracy	Precision	Recall	F1-Score
SVC	0.968028	0.911528	0.897098	0.904255

Table 40 - Tuned SVC model performance on Test Set

Since recall is the priority (to correctly identify churned customers), the SVC model remains the best choice as it maintains a high recall (0.8971) even on the test set.

This confirms its generalizability and robustness across datasets.

Feature Importance

For the Support Vector Classifier (SVC), performing feature importance and confusion matrix analysis requires specific techniques because SVC does not inherently provide feature importance like tree-based models (e.g., Random Forest).

As we used non-linear kernels (RBF) as the best parameter, we can use permutation importance, which measures the drop in model performance when a feature's values are shuffled.

Feature	Importance
Tenure_Group	0.065142
Complain_ly_1.0	0.064298
CC_Agent_Score	0.034503
Gender_Male	0.033792
Loyalty_Score	0.033748
City_Tier	0.032682
account_segment	0.029307
rev_per_month	0.028552
Marital_Status_Single	0.026909
rev_growth_yoy	0.025577
Day_Since_CC_connect	0.024911
Login_device_Mobile	0.024556
CC_Contacted_LY	0.022202
coupon_used_for_payment	0.02087
Payment_DC	0.020382
Account_user_count	0.019716
Payment_Wallet	0.017806
Payment_COD	0.016874
Payment_UPI	0.014565
Service_Score	0.012389
cashback	0.012211

Marital_Status_Married	0.008481
Login_device_Others	0.003286

Table 41 - Feature Importance

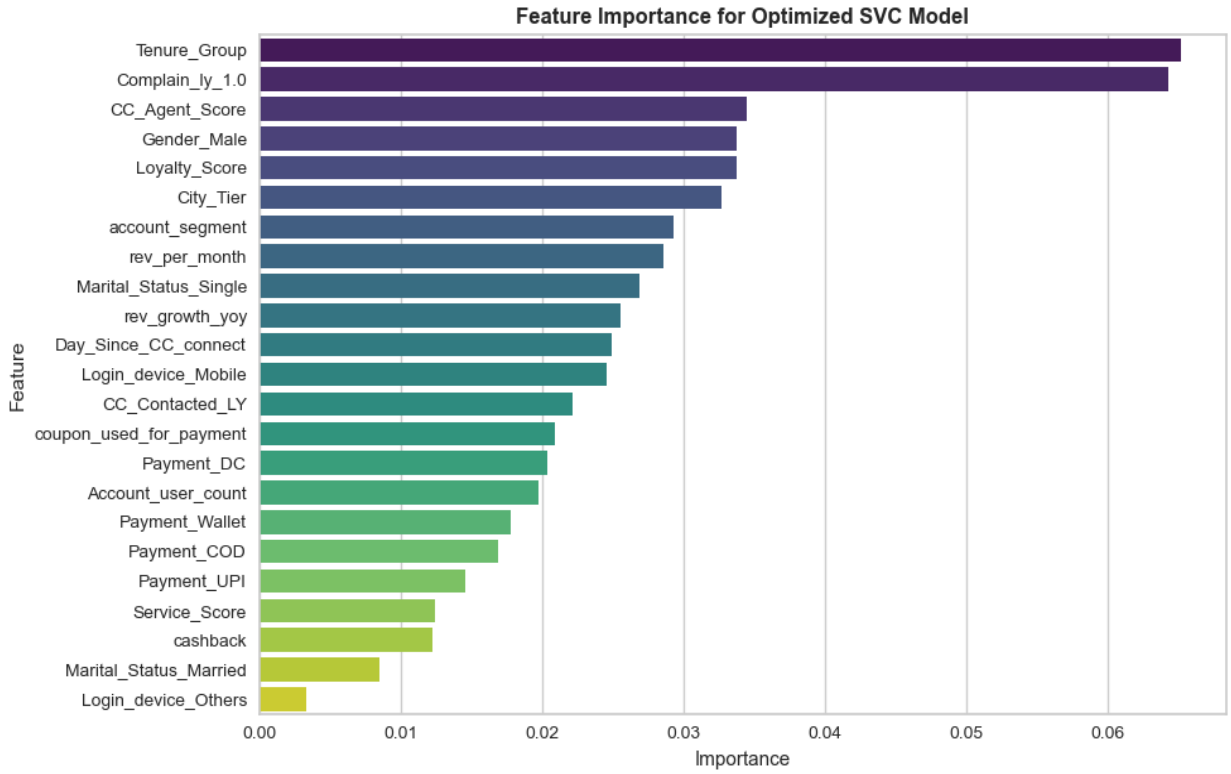


Figure 49 - Feature Importance Graph

Clustering

Checking Elbow Plot

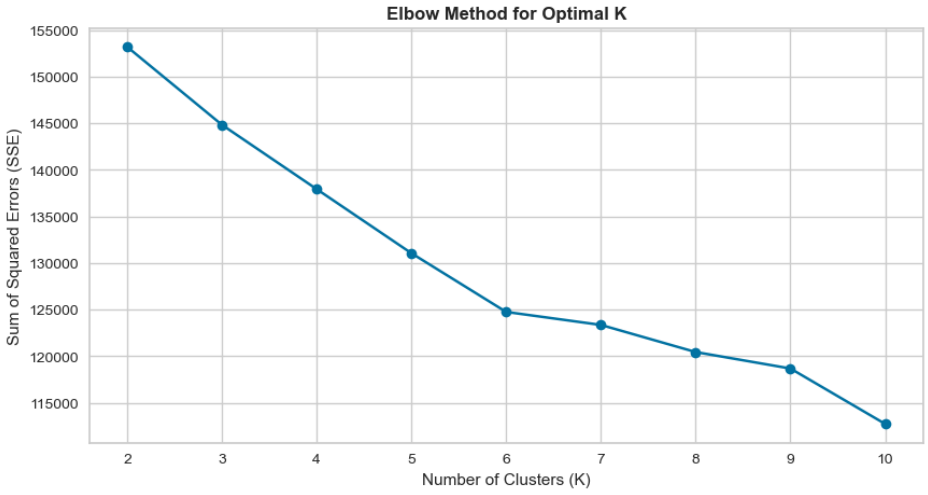


Figure 50 - Elbow Plot

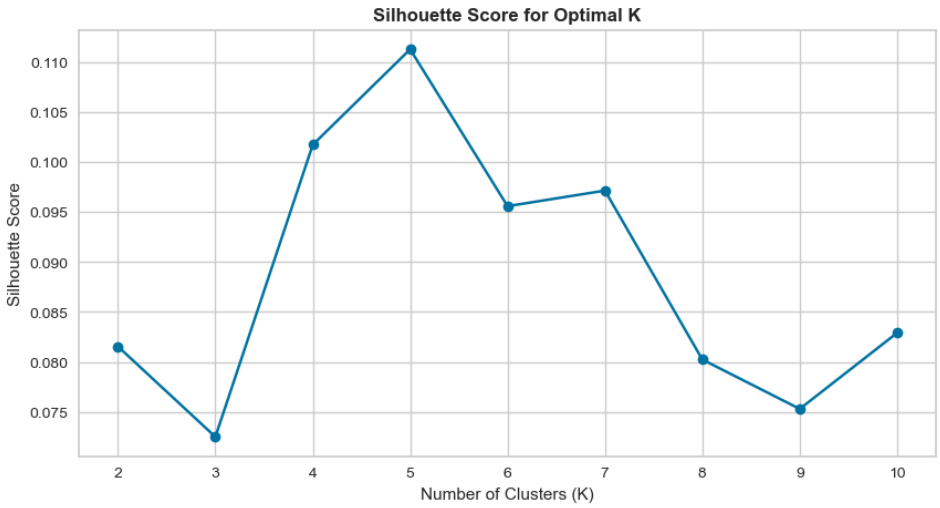


Figure 51 - Silhoutte Plot

From the two graphs:

Elbow Method: The SSE (Sum of Squared Errors) curve starts flattening significantly around K = 5 or 6, suggesting a good balance between reducing error and avoiding too many clusters.

Silhouette Score: The highest Silhouette Score is at K = 5, indicating the best cluster separation.

Cluster profiling

Cluster Profiling for Numerical Columns:

Cluster	Tenure	City_Tier	CC_Contacted_LY	Service_Score	Account_user_count	CC_Agent_Score	rev_per_month	Complain_ly	rev_growth_yoy	coupon_used_for_payment	Day_Since_CC_connect	cashback	Churn	Cluster
0	10.3833	1.4968	17.459	2.8596	3.7413	3.1435	5.1404	0.2508	15.6041	1.7319	4.5994	171.7259	0.265	0
1	15.5596	1.5092	16.2155	3.1411	3.9571	3.0923	6.6195	0.2529	16.6504	2.79	6.4628	218.6408	0.1012	1
2	11.4824	2.9892	19.2304	2.9187	3.7561	3.1098	5.1721	0.2737	15.8713	1.6558	4.8279	187.0478	0.206	2
3	8.4835	1.4686	18.4737	2.7697	3.5326	3.0006	4.5601	0.2811	16.1451	1.0075	3.3152	157.278	0.1823	3
4	10.0983	1.6358	17.8295	2.8699	3.8353	3.0665	5.0751	0.2775	16.2948	1.5405	4.2399	174.9921	0.1618	4

Cluster Profiling for Categorical Columns:

	Payment					Gender		account_segment					Marital_Status			Login_device		
	CC	COD	DC	UPI	Wallet	Female	Male	HNI	Regular	Regular Plus	Super	Super Plus	Divorced	Married	Single	Computer	Mobile	Others
Cluster																		
0	0	634	0	0	0	256	378	105	23	292	187	27	90	322	222	171	463	0
1	780	4	1037	180	4	786	1219	483	266	291	637	328	302	1220	483	598	1407	0
2	0	0	0	0	738	311	427	148	36	85	417	52	120	413	205	192	546	0
3	1390	0	1756	337	0	1371	2112	261	74	1728	1326	94	507	1743	1233	972	2511	0
4	95	25	169	24	33	125	221	45	96	50	130	25	51	178	117	0	0	346

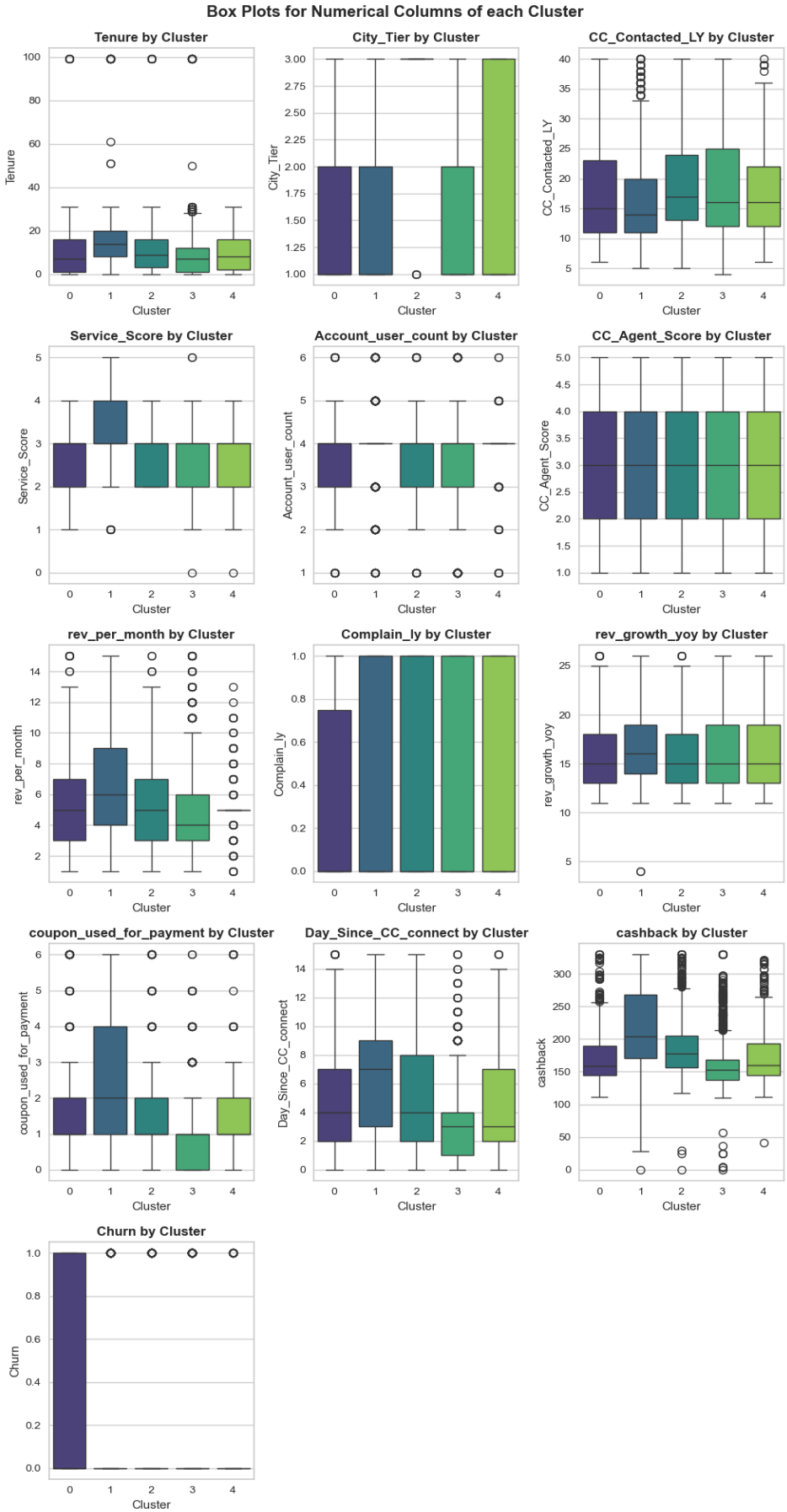


Figure 52 - Numerical Columns by Each Clusters

Figure 53 - Cluster Group by Means

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

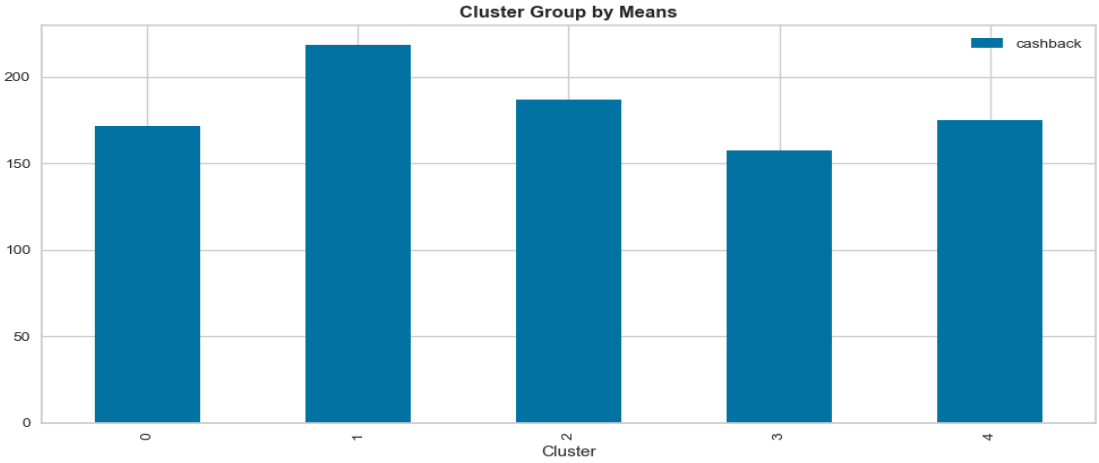


Figure 54 - Cluster Groups by Means for Cashback

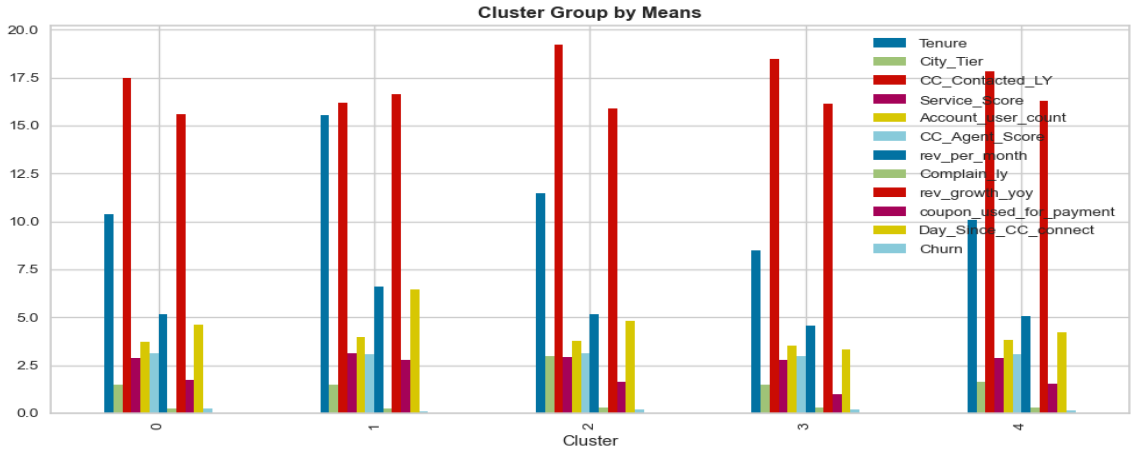


Figure 55 - Cluster Groups by Means for Other Features

Cluster Summary

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

- **Cluster 0**
  - **Tenure**: Low (3.37 years)
  - **CC_Contacted_LY**: High (18.65 contacts)
  - **Account User Count**: Moderate (3.74 users)
  - **Revenue per Month**: Low ($5.11K)
  - **Revenue Growth YoY**: Moderate (16.28%)
  - **Coupon Usage for Payment**: Low (1.20 coupons)
  - **Days Since Last CC Connect**: Low (3.17 days)
  - **Cashback**: Low ($162.62)
  - **Characteristics**: Newer customers with recent engagement, low revenue, and minimal coupon usage. Likely need proactive engagement to increase loyalty and spending.
- **Cluster 1**
  - **Tenure**: High (18.14 years)
  - **CC_Contacted_LY**: Moderate (15.19 contacts)
  - **Account User Count**: High (3.94 users)
  - **Revenue per Month**: High ($8.54K)
  - **Revenue Growth YoY**: Moderate (16.11%)
  - **Coupon Usage for Payment**: High (4.36 coupons)
  - **Days Since Last CC Connect**: High (7.93 days)
  - **Cashback**: High ($298.56)
  - **Characteristics**: Long-term customers with high revenue and coupon usage but less frequent engagement. Valuable segment requiring retention strategies to maintain loyalty.
- **Cluster 2**
  - **Tenure**: Moderate (13.33 years)
  - **CC_Contacted_LY**: High (18.40 contacts)
  - **Account User Count**: Moderate (3.63 users)
  - **Revenue per Month**: Moderate ($5.41K)
  - **Revenue Growth YoY**: Moderate (16.30%)
  - **Coupon Usage for Payment**: Moderate (1.31 coupons)
  - **Days Since Last CC Connect**: Moderate (4.28 days)
  - **Cashback**: Moderate ($181.19)
  - **Characteristics**: Mid-term customers with balanced engagement, moderate revenue, and coupon usage. Potential for growth through targeted offers.
- **Cluster 3**
  - **Tenure**: Moderate (13.24 years)
  - **CC_Contacted_LY**: High (17.84 contacts)
  - **Account User Count**: Moderate (3.58 users)
  - **Revenue per Month**: Moderate ($7.53K)
  - **Revenue Growth YoY**: Moderate (16.07%)
  - **Coupon Usage for Payment**: Moderate (1.45 coupons)
  - **Days Since Last CC Connect**: Moderate (4.66 days)
  - **Cashback**: Moderate ($182.89)

    - **Characteristics**: Mid-term customers with slightly higher revenue and engagement compared to Cluster 2. Likely to respond well to loyalty programs and personalized offers.
- **Summary**
    - **Cluster 0**: Focus on increasing engagement and revenue through targeted campaigns and promotions.
    - **Cluster 1**: Strengthen retention strategies to maintain high revenue and loyalty.
    - **Cluster 2 & 3**: Enhance customer experience with personalized offers to drive growth and increase coupon usage.

Key Insights from Clustering

- **Cluster 0 (Newer Customers)**:
    - **Characteristics**: Low tenure, recent engagement, low revenue, and minimal coupon usage.
    - **Insight**: These customers are new and have low spending but are actively engaged. They represent an opportunity to increase loyalty and revenue through targeted campaigns.
- **Cluster 1 (Long-Term Customers)**:
    - **Characteristics**: High tenure, high revenue, high coupon usage, but less frequent engagement.
    - **Insight**: These are loyal, high-value customers who generate significant revenue. However, their engagement is declining, which could lead to churn if not addressed.
- **Cluster 2 (Mid-Term Customers)**:
    - **Characteristics**: Moderate tenure, balanced engagement, moderate revenue, and coupon usage.
    - **Insight**: This segment has steady revenue and engagement but shows potential for growth through personalized offers and improved customer experience.
- **Cluster 3 (Mid-Term Customers with Higher Engagement)**:
    - **Characteristics**: Moderate tenure, higher engagement, moderate revenue, and coupon usage compared to Cluster 2.
    - **Insight**: These customers are more engaged than Cluster 2 and are likely to respond well to loyalty programs and personalized offers.