Ecommerce Churn Rate

Python

1. change the column name form Upper with spaces to lower case with underscore (snake_casing)
2. checking duplicate rows
3. checking for null values
4. Droppping null value rows
5. Standardising Date columns
6. Removing negative values in price and quantity
7. Ensure churn flag is an integer (not float)
8. checking the shape of the dataset after cleaning & before adding new column
9. Adding column of days a customer first signup to last login
10. Ensuring all tenure days are positive as last login date !< signup date
11. Connecting to PostgreSQL

SQL

1. Acquisition Quality Analysis (Churn by Marketing Channel)

Why we are doing this (Business Decision): Marketing teams often celebrate high acquisition numbers, but if those users churn immediately, the Return on Ad Spend (ROAS) is negative. This query identifies which channels bring in "loyal" customers versus "low-quality" customers. If "Paid Ads" has a 60% churn rate while "Referral" has only 10%, you should shift budget to Referrals.

How we do it: We group customers by their Marketing_Channel and calculate the churn percentage for each.

Insight to look for: A specific channel (e.g., "Influencer Promo") might have high volume but disproportionately high churn.

2. The "Price Sensitive" Churn (Impact of Discounts)

Why we are doing this (Business Decision): Businesses often use heavy discounts to acquire users. The risk is attracting "bargain hunters" who leave as soon as the discounts stop. This query helps decide if your pricing strategy is sustainable. If users who *never* received a discount have lower churn, your product value is strong. If only discounted users stay, you have a pricing problem.

How we do it: We segment customers based on whether they have utilized discounts (Discount_Applied) and compare churn rates between the two groups.

3. Operational Friction Analysis (Delivery Time Impact)

Why we are doing this (Business Decision): Churn isn't always about the product; sometimes it's about the logistics. If customers receiving items in 7+ days churn at double the rate of those receiving them in 2 days, you have a clear business case to invest in faster logistics or new warehouse locations.

How we do it: We create "bins" or categories for Delivery_Time_Days and calculate the churn rate for each bucket.

4. High-Value Customer Loss (VIP Churn)

Why we are doing this (Business Decision): Losing 10 "New" customers might cost you $100, but losing 1 "VIP" customer might cost you $1,000. Not all churn is created equal. This analysis focuses retention efforts where they matter most financially. If VIP churn is spiking, you need an immediate "white glove" retention campaign.

How we do it: We filter by Customer_Segment and look at the revenue impact of the churned users in those segments.

5. The "Critical Window" (Tenure Analysis)

Why we are doing this (Business Decision): When do customers leave? Is it after the first week (onboarding issue) or after 6 months (engagement issue)? Knowing the "Tenure Drop-off" allows you to set up automated emails or offers *just before* that critical moment to save them.

How we do it: We analyze the tenure_days of **churned customers only** to find the average lifespan or group them into tenure buckets.

6. Predicting "At-Risk" Customers (RF Analysis)

This is a **Proactive** analysis. Instead of looking at who *has* churned, we look at active customers (Churn_Flag = 0) who are exhibiting behavior similar to those who left.

Why we are doing this (Business Decision): It is 5x–25x cheaper to retain an existing customer than to acquire a new one. By identifying customers who used to buy frequently but haven't logged in recently, you can trigger specific "Win-Back" email campaigns (e.g., "We miss you, here is 10% off") *before* they officially churn.

How we do it: We use a simplified **RF (Recency, Frequency)** approach:

1. **Recency:** How many days has it been since they last logged in?
2. **Frequency:** How many total purchases have they made?
3. **Risk Logic:**
   - **High Risk VIP:** Customers who buy often (Frequency > 5) but haven't visited in a long time (Recency > 60 days). These are your top priority to save.
   - **Drifting:** Customers with average frequency who haven't visited recently.

o **Safe:** Customers who visited recently.

Note: In this query, we assume the "Current Date" is the latest date available in your dataset.

Interpretation of Results for Action:

- **High Risk VIP (Critical):** Export this list immediately. These people know your product and like it (high frequency), but something stopped them. Send a personalized email or a high-value discount.

- **At Risk (Drifting):** These might be casual shoppers who forgot about you. A gentle nudge or a newsletter might be enough.

- **Warning Zone:** Keep an eye on them; perhaps they just need a "New Arrivals" notification

7. The "Behavioral Cooling" Analysis (Basket Size Decline)

Why we are doing this (Business Decision): Customers rarely just "vanish." They usually disengage slowly—a phenomenon known as **Behavioral Cooling**. A customer who used to buy 20 items at a time but dropped to buying 2 items just before they stopped logging in was signalling dissatisfaction.

- **The Goal:** Prove that shrinking basket sizes is a leading indicator of churn.

- **The Action:** If the data proves this, you can trigger a "Volume Discount" (e.g., "Buy 5 items, get 20% off") the moment a VIP's average basket size drops by 10%.

How we do it: We compare the **First Basket Size** (quantity of items in their first order) vs. the **Last Basket Size** (quantity in their last order) specifically for customers who have already churned.

Insights to look for:

- **Negative % Change (e.g., -30%):** This confirms "Cooling." Users gradually lost interest. You need to intervene earlier in the lifecycle.

- **No Change (0%):** The churn was sudden (e.g., bad customer service experience, price hike, or competitor offer) rather than a gradual loss of interest.

8. Cross-Category "Share of Wallet" Analysis

Why we are doing this (Business Decision): Sometimes users don't stop buying *everything*; they just stop buying your *high-margin* items. A customer might still buy "Office Supplies" (low margin) but stop buying "Electronics" (high margin). This query checks if churners abandoned specific categories first.

How we do it: We look at the total revenue per category for churned vs. active customers to see if the "Product Mix" is different.

Interpretation: If Home Decor has a much higher share in Churned customers (+5%) than Active customers, it might mean your Home Decor products are low quality and causing people to leave!

9. Acquisition Seasonality (The "Holiday Shopper" Effect)

Why we are doing this (Business Decision): Many businesses see a spike in signups during holidays (e.g., Black Friday, Christmas). However, these customers often exhibit the highest churn rates because they came for a specific deal or gift, not for the long-term product value.

- **The Problem:** If your "December Cohort" has a 90% churn rate, your marketing team shouldn't be celebrating the "record-breaking signups" in Q4.

- **The Decision:** Adjust your Customer Lifetime Value (CLV) models. Don't spend the same amount of ad money acquiring a "December user" as you do a "July user" if the December user is worth half as much.

How we do it: We group customers by the *Month* of their Signup_Date and calculate the churn rate for each month.

Insights to look for:

- **The "January Hangover":** If December signups have high churn, it confirms they were just holiday shoppers.

- **The "Summer Slump":** If July signups have *low* churn, it means people who sign up in the summer (when sales are usually slow) are your most deliberate and loyal customers. You should target them more aggressively.

10. Cohort Analysis (The "Retention Curve")

Since we are looking at *when* people signed up, the most powerful way to visualize this is a **Cohort Analysis**. This answers: *"Of the people who signed up in January, how many were still active in February, March, April…?"*

Why we are doing this: This detects **Product Changes**. If the "March Cohort" has much better retention than the "January Cohort," it proves that a feature you released in March actually improved the product stickiness.

How we do it: We calculate the time difference (in months) between their Signup_Date and Last_Login_Date for all active users.