**Project Report**

**On**

**“Exploratory Data Analysis on Amazon Sales Data”**

Submitted for partial fulfilment of requirement for the award of course of

**Data Scientist**

Submitted by

**Yagnik Patel**

**2025**

#### Problem Statement & Objective

#### ****Problem Statement:****

To analyze Amazon's sales transaction data in order to uncover key factors that impact revenue, customer behavior, and operational efficiency. The goal is to extract actionable business insights that can guide strategic decisions related to sales performance, marketing, customer engagement, and discounting practices.

#### ****Objectives:****

* **Sales Performance Analysis:**  
  Identify top-performing product categories, regions, and customer segments that drive the most revenue.
* **Customer Behavior Insights:**  
  Understand customer purchase patterns, repeat behavior, and peak buying periods to improve targeting and engagement.
* **Operational Efficiency:**  
  Analyze order quantities, shipping methods, and delivery performance to spot inefficiencies and optimize fulfillment.
* **Marketing & Promotional Effectiveness:**  
  Evaluate the impact of discounts on profit margins and identify optimal discount strategies that boost sales without hurting profitability.
* **Product Mix Optimization:**  
  Recommend the best-performing products and combinations to refine the overall product strategy for higher profitability.

**Introduction**

**Exploratory Data Analysis (EDA)**

Exploratory Data Analysis (EDA) is an approach to analyzing data sets to summarize their main characteristics using statistical methods and data visualization techniques.  
It is a crucial step in any data analysis process, enabling analysts to uncover patterns, detect anomalies, test hypotheses, and extract valuable business insights.

**EDA Pipeline**

1. **Data Acquisition and Objective**  
   a. Obtain Amazon sales data (CSV format)  
   b. Define the business objectives for analysis  
   c. Choose appropriate tools and programming environment (Python)
2. **Data Loading/Reading**  
   a. Load CSV data directly into Jupyter Notebook using Python  
   b. Prepare the data for further cleaning and analysis
3. **Familiarize with Data & Identify Target Variable**  
   a. Explore dataset structure (column names, data types)  
   b. Identify target variables relevant to business objectives (e.g., Sales, Profit, Quantity)
4. **Data Preparation & Transformation**  
   a. Perform data cleaning  
   b. Handle missing values (if any)  
   c. Remove duplicates and irrelevant data  
   d. Convert and format data types (categorical, numerical, datetime)
5. **Feature Engineering**  
   a. Create new variables like Year, Month, Week, Discount %, Profit Margin, etc.
6. **Data Analysis & Visualization**  
   a. Univariate Analysis  
   i. Numerical variables (mean, median, standard deviation)  
   ii. Categorical variables (frequency distribution)  
   b. Bivariate & Multivariate Analysis  
   i. Identify patterns across multiple dimensions  
   ii. Visualizations (bar charts, pie charts, boxplots, heatmaps, line plots)
7. **Summary and Suggestions**  
   a. Derive key business insights and offer actionable recommendations

**About the Company**

* Amazon is a global e-commerce and technology company operating in multiple regions.
* The platform offers a wide variety of products and services to a large customer base.
* The analysis aims to explore Amazon’s sales data to derive insights into performance, customer behavior, and operational efficiency.
* The objective is to support data-driven decision-making using historical transaction records.

### ****Tools and Platforms Used in Project****

#### ****Why Python?****

Python is a powerful, high-level, open-source programming language widely used for data analysis, visualization, and automation.  
It provides a rich ecosystem of libraries such as **Pandas, NumPy, Matplotlib, and Seaborn**, which simplifies data manipulation and visualization tasks.  
Python is one of the most in-demand tools for Data Science, especially for projects involving structured data analysis like sales and profitability studies.

#### ****Platforms Used****

* **Jupyter Notebook** – A web-based application used for writing and running Python code, data cleaning, analysis, and visualization.
* **Visual Studio Code (VS Code)** – A lightweight code editor for editing Python files, especially .ipynb files with Jupyter extension support.
* **CSV Files** – Data was sourced in comma-separated value (.csv) format, making it simple to load and process directly in Python without a database.

#### ****Versions of Platform****

* **Jupyter Notebook** – 7.1.3
* **Visual Studio Code** – 1.89.1
* **Python** – 3.12.2

### ****Chapter 1: Data Loading/Reading****

#### ****Load Data in Python (CSV File)****

The Amazon sales dataset used in this project is in **CSV (Comma-Separated Values)** format. The data is directly loaded into Python using the **Pandas** library, which provides efficient tools for reading, processing, and analyzing structured data.

#### ****Import Necessary Libraries****

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import warnings

warnings.filterwarnings("ignore")

%matplotlib inline

* **pandas (pd)** – For reading CSV and managing data in DataFrame format
* **numpy (np)** – For numerical operations
* **matplotlib.pyplot (plt)** and **seaborn (sns)** – For creating visualizations
* **warnings** – To suppress unnecessary warning messages

#### ****Load CSV Data into Jupyter Notebook****

# Load the CSV file into a DataFrame

df = pd.read\_csv("Amazon Sales Data.csv")

# Display the first 5 rows

df.head()

* The dataset is now loaded into a variable named df, which holds all the transaction data.
* The first few rows are displayed using df.head() to verify successful loading.

#### ****Interpretation****

* Data has been successfully imported into the Jupyter Notebook using **pd.read\_csv()**.
* It is now ready for exploration, cleaning, and transformation.
* From this step onward, all analysis will be done using **Python only**—no database or external connections are used.

### ****Chapter 2: Familiarize with Data & Identifying the Target Variable****

#### ****Explore the Provided Data (Column Names, Data Types)****

Before performing any cleaning or analysis, it's important to understand the structure of the dataset. This includes identifying the number of rows and columns, checking data types, and reviewing basic statistics.

# Display the first and last few rows

df.head()

df.tail()

# Check the shape (rows, columns)

df.shape

# Check total number of data elements

df.size

# Display info about column data types and non-null counts

df.info()

#### ****Overview of Data****

* The dataset contains information about individual transactions on Amazon, including product details, customer location, quantity, sales, and profit.
* The data is in tabular format, which is structured and ready for processing.
* Preliminary checks help in identifying columns that may need cleaning, renaming, or transformation.

#### ****Interpretation****

* **Structured Data**: The dataset is organized into rows and columns, with each row representing one transaction.
* **Dimensions**: Based on the .shape output, we get the total number of records (1,28,975) and attributes (24).
* **Data Types**: A mix of object, int64, and float64 types are observed, indicating both categorical and numerical variables.
* **Non-null Count**: The .info() function shows whether any columns contain missing (null) values.
* **Target Variables**: Based on our business objective, the following are identified as key metrics for analysis:
  + Sales
  + Quantity
  + Order Date (for time-based analysis)

### ****Chapter 3: Data Preparation & Transformation****

#### ****Data Cleaning****

The raw data must be cleaned before performing any analysis. This includes identifying and handling missing values, duplicates, and correcting data formats.

# Check for missing values

df.isnull().sum()

# Check for duplicates

df.duplicated().sum()

# Remove duplicate rows

df = df.drop\_duplicates()

#### ****Handle Missing Values****

* The .isnull().sum() function helps identify columns with missing values.
* If the number of missing values is minimal, we can drop them.
* If a column has significant missing values but is important, we can use imputation methods (mean, median, mode) or fillna method.
* In this dataset, we found some missing values so we were replace them with fillna method like  
   df['ship-city'].fillna(method='ffill',inplace=True)

df['ship-state'].fillna(method='ffill',inplace=True)

df['Amount'].fillna(df['Amount'].mean(),inplace=True)

df['Courier\_Status'].fillna(df['Courier\_Status'].mode()[0],inplace=True)

#### ****Data Reduction: Remove Unwanted Space, Columns or Rows****

* Check for columns that are irrelevant to analysis and drop them.
* In the data set we found some trilling spaces in the column name so we can remove this spaces like

df.columns = df.columns.str.strip()

* Unwanted Column or Rows, In this case, there are some unwanted columns so we can delete the unwanted columns like this

df.drop(columns = ['index','currency','ship-country','promotion-ids','fulfilled-by','Unnamed: 22','ship-postal-code'],inplace=True)

* Duplicates, if found, were removed to avoid skewing the analysis.

df.duplicated().sum(); This shows number of duplicated rows

df = df.drop\_duplicates(); This removes duplicated rows

* Interpretation:
  + It is noted that there are 6 rows which are repeated. We have removed duplicated rows

#### ****Format Data Types (Numerical & Categorical Variables)****

# Check current data types

df.info()

# Convert date columns to datetime format

df['Date'] = pd.to\_datetime(df['Date'])

# Convert object-type categorical columns to 'category' for memory optimization

cat\_cols = df.select\_dtypes(include='object').columns

for col in cat\_cols:

df[col] = df[col].astype('category')

#### ****Rename Columns****

# Example: remove special characters or spaces in column names

df.columns = df.columns.str.strip() - This remove unwanted space from columns name.

df.columns = df.columns.str.replace(' ','\_') - This covert space(‘ ‘) into (‘\_’) underscores.

df.columns; This display columns for cross verifying that renaming step is performed

* Interpretation:
  + Feedback column is renamed by removing space in column

#### ****Change Data Types and Memory Optimization****

* Converting object types to category reduces memory usage.
* Converting date columns allows time-based operations like extracting year, month, etc.
* After optimization, the memory footprint of the DataFrame is significantly reduced.

#### ****Feature Engineering (Create New Features/Variables)****

New columns are created to enrich the dataset and enable deeper insights.

# Extracting time-based features

df['Year'] = df['Date'].dt.year

df['Month'] = df['Date'].dt.month

#### ****Interpretation****

* Data is now clean, consistent, and structured.
* All missing and duplicate values are handled.
* New features such as Year, Month will help derive actionable business insights in the upcoming analysis.

### ****Chapter 4: Data Analysis & Visualization****

#### ****Overview of Data Before Analysis****

* After data wrangling, we can check the structure and columns once before proceeding with detailed analysis.
* df.columns — This command displays all columns available in the DataFrame.

#### ****Interpretation:****

**Description of Variables**

|  |  |  |
| --- | --- | --- |
| **Variables / Columns** | **Description** | **Unique Values** |
| **Order\_ID** | Unique identifier for each order transaction | 120,378 |
| **Date** | Date when the order was placed | 91 |
| **Status** | Order status (e.g., Delivered, Returned, Cancelled, etc.) | 13 |
| **Fulfilment** | Fulfillment method used (e.g., Amazon, Merchant fulfilled) | 2 |
| **Sales\_Channel** | Channel used for sale (e.g., Amazon.in, other marketplaces) | 2 |
| **ship-service-level** | Shipping level selected (e.g., Standard, Express) | 2 |
| **Style** | Product style/variant (e.g., color, design) | 1,377 |
| **SKU** | Stock Keeping Unit – internal product identifier | 7,195 |
| **Category** | Broad product category (e.g., Apparel, Electronics) | 9 |
| **Size** | Size of the product (e.g., S, M, L, XL) | 11 |
| **ASIN** | Amazon Standard Identification Number (product-level unique code) | 7,190 |
| **Courier\_Status** | Shipment progress as reported by courier (e.g., In Transit, Delivered) | 3 |
| **Qty** | Quantity of product ordered in a transaction | 10 |
| **Amount** | Transaction amount for the order | 1,411 |
| **ship-city** | City to which the product was shipped | 8,955 |
| **ship-state** | State to which the product was shipped | 69 |
| **B2B** | Indicates if order was Business-to-Business or retail (Yes/No or binary) | 2 |
| **Year** | Extracted year from the order date | 1 |
| **Month** | Extracted month from the order date | 4 |

|  |  |
| --- | --- |
|  |  |
|  |  |
| Univariate analysis We need to perform univariate analysis on relevent columns in Amazon Sales data for more targetted approach Summary statistics  * df.describe().T; We generate summary statistics for numerical columns in data df. and transpose the output * Interpretation: * Summary statistics provide valuable insights of data distribution to understand our data better * Notable Observations * Order Quantity: Range from single unit to 15 units * Sales:Miximum sales value is upto 5584 Rupees   Note: Numerical summary stats on categorical variable in Amazon Sales data set are not yielding any valuable insights Univariate Analysis of Categorical ColumnsDistribution of order status |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

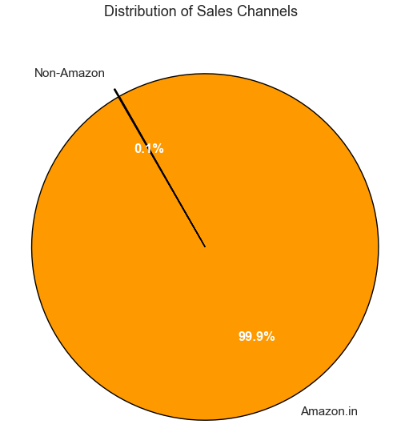
**Order Status Distribution**  
• **Shipped** orders dominate at **60.3%**, indicating a strong fulfillment rate.  
• **22.3%** of orders are **delivered**, showing successful last-mile completion.  
• **14.2%** of orders are **cancelled**, suggesting potential issues in the order process.  
• Other statuses each account for less than **2%**, with minimal operational impact.

#### Distribution of order Fulfilment

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

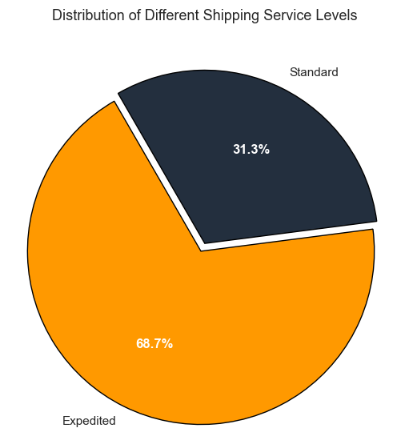
**Distribution of Different Fulfilment Types**  
• **Amazon** handles the majority of fulfilment at **69.5%**, showing a strong preference for in-house logistics.  
• **Merchant-fulfilled** orders account for **30.5%**, indicating a notable but smaller share.

#### Distribution of order Sales Channels



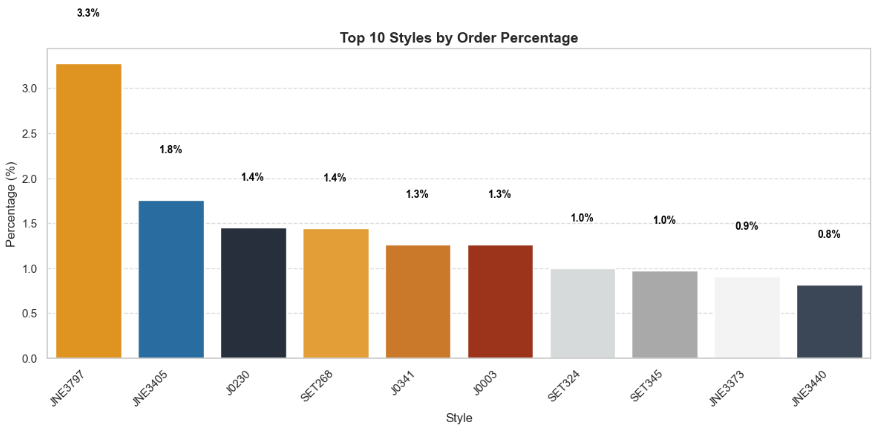
**Distribution of Sales Channels**  
• **Amazon.in** accounts for an overwhelming **99.9%** of sales, making it the dominant sales channel.  
• **Non-Amazon** sales contribute only **0.1%**, indicating minimal external channel activity.

#### Distribution of order Shipping Service



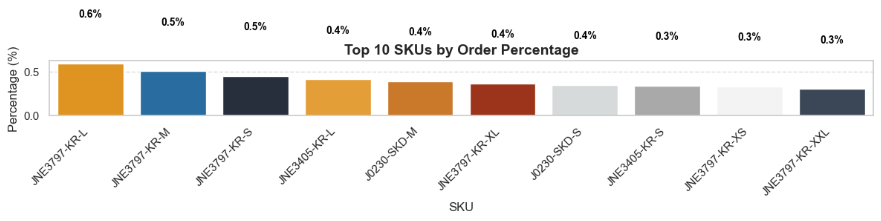
**Distribution of Different Shipping Service Levels**  
• **Expedited** shipping represents **68.7%**, indicating it is the most preferred service level.  
• **Standard** shipping accounts for **31.3%**, showing a significant but lesser usage.

#### Distribution of Top 10 Styles



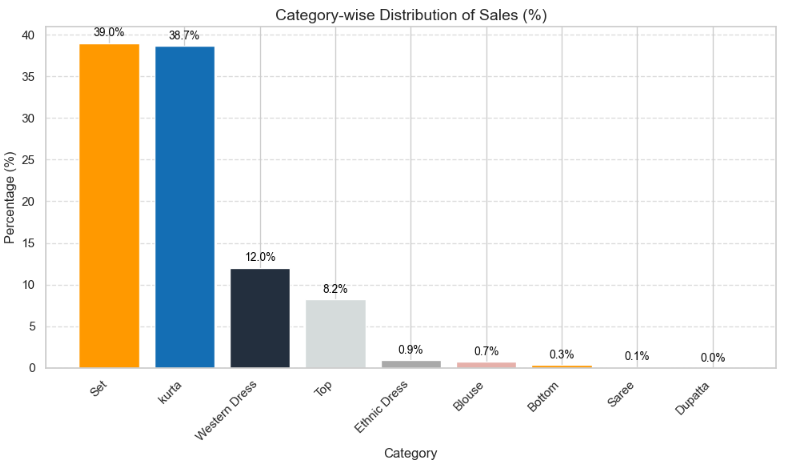
**Top 10 Styles by Order Percentage**  
• **JNES787** is the top-selling style with **3.3%** of total orders.  
• **JNES3405** follows with **1.8%**, and **JD230**, **SET286** both account for **1.4%**.  
• The rest—**JD0341**, **JD003**, **SET324**, **SET345**, **JNES7373**, and **JNES3440**—each contribute between **1.3%** and **0.8%**.

#### Distribution of Top 10 SKU



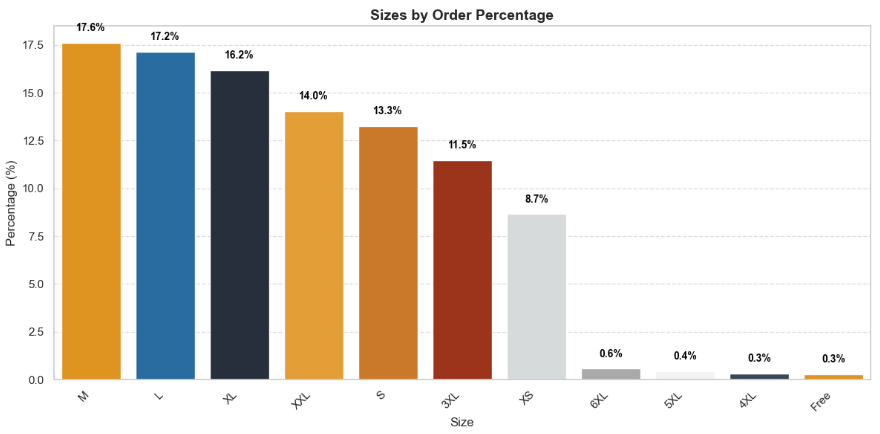
**Top 10 SKUs by Order Percentage**  
• **JNES797-KR-L** is the most ordered SKU, accounting for **0.6%** of total orders.  
• **JNES797-KR-M** and **JNES797-KR-S** follow closely with **0.5%** each.  
• Other SKUs, including **JNES3405-KR-L**, **JD230-SKD-M**, and **JNES797-KR-XL**, each contribute around **0.4%**.

#### Distribution of Category



**Category-wise Distribution of Sales (%)**  
• The majority of sales come from the **Set** and **Kurta** categories, contributing **39.0%** and **38.7%**, respectively.  
• **Western Dress** accounts for **12.0%**, followed by **Top** at **8.2%**, showing notable demand in these categories as well.  
• Sales for other categories like **Ethnic Dress (0.9%)**, **Blouse (0.7%)**, **Bottom (0.3%)**, **Saree (0.1%)**, and **Dupatta (0.0%)** are relatively negligible.

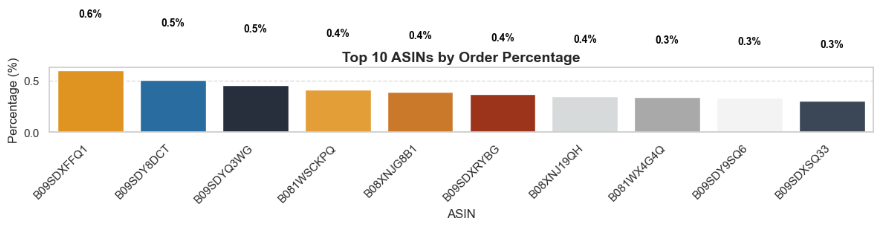
#### Distribution of Size



### Sizes by Order Percentage

* **M (17.6%)**, **L (17.2%)**, and **XL (16.2%)** are the most ordered sizes.
* **XXL (14.0%)**, **S (13.3%)**, and **3XL (11.5%)** also have significant shares.
* **XS** accounts for **8.7%** of the orders.
* Larger sizes like **6XL (0.6%)**, **5XL (0.4%)**, **4XL (0.3%)**, and **Free size (0.3%)** have minimal order percentages.

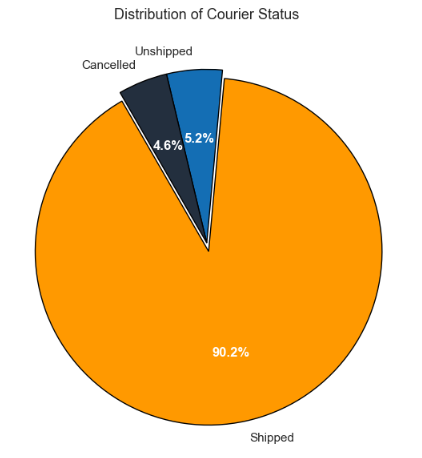
#### Distribution of order Top 10 ASIN



### Top 10 ASINs by Order Percentage

* **B09SDXFFQ1** has the highest order share at **0.6%**.
* **B09SDY8DCT (0.5%)** and **B09SDYQ3WG (0.5%)** follow next in order volume.
* ASINs **B081WSCXPO**, **B06XXJVG81**, and **B09SDXRYBG** each have an order share of **0.4%**.
* ASINs **B08XXJJ9QH** and **B081WX4G4Q** also contribute **0.4%** each.
* The lowest order percentages are shared by **B09SDY8SQ6** and **B09SDXSQ33**, each at **0.3%**.

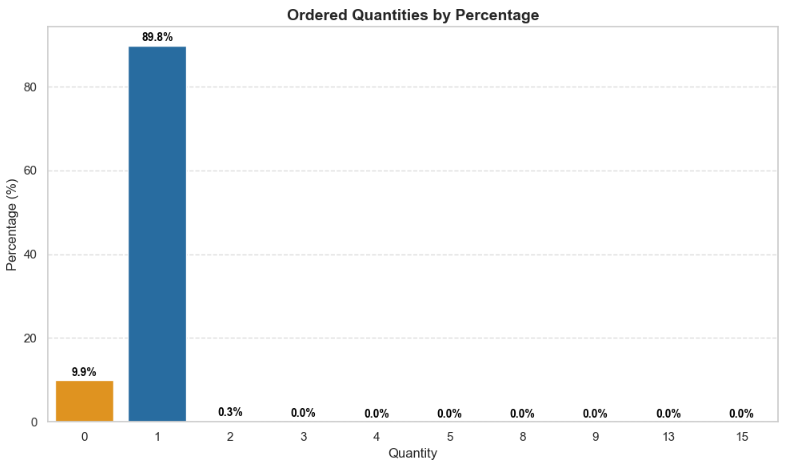
#### Distribution of Courier Status



### Distribution of Courier Status

* The majority of orders are **Shipped**, accounting for **90.2%** of the total.
* **Unshipped** orders make up **5.2%**, indicating a small portion of pending shipments.
* **Cancelled** orders represent **4.6%**, reflecting a relatively low cancellation rate.

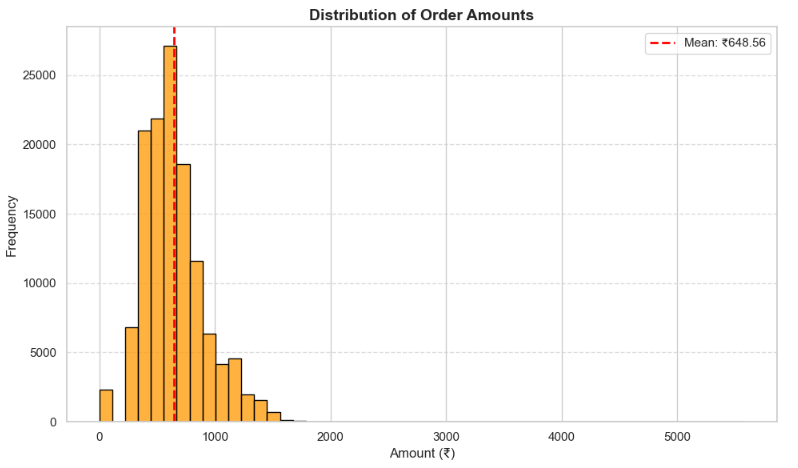
#### Distribution of order QTY



### Ordered Quantities by Percentage

* A large majority of orders consist of **quantity 1**, contributing **89.8%** of total orders.
* **Quantity 0** follows with **9.9%**, likely representing cancelled or unfulfilled items.
* Very few orders have higher quantities: **Quantity 2** makes up just **0.3%**, and all quantities above 2 contribute **0.0%**, indicating negligible bulk ordering behavior.

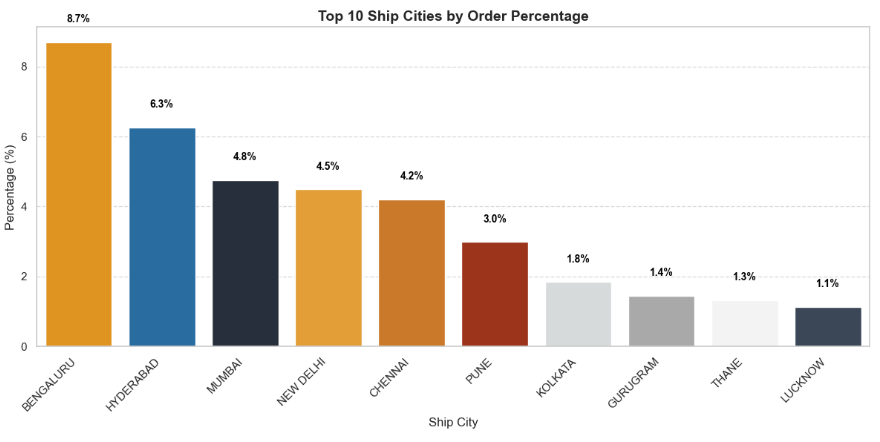
#### Distribution of Amount



### Distribution of Order Amounts

* The order amounts are heavily concentrated between ₹300 and ₹900, with a peak frequency around ₹650.
* The **mean order amount is ₹648.56**, shown by the red dashed line, indicating most customers spend around this value.
* Very few orders exceed ₹2000, suggesting **high-value orders are rare**, and pricing is generally affordable for the majority.

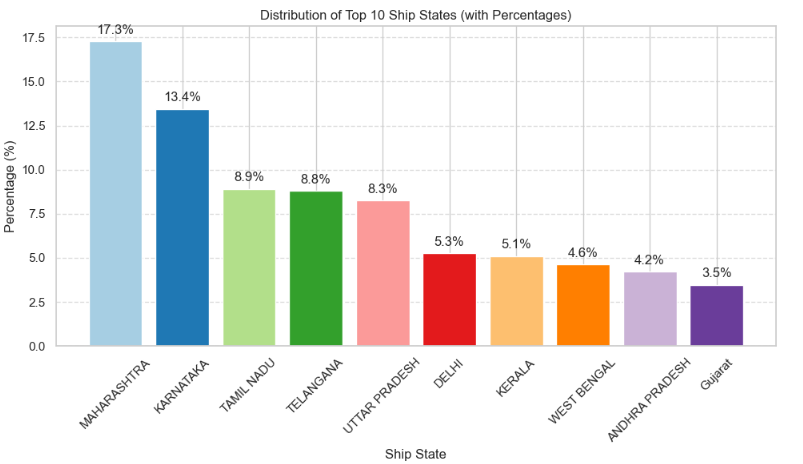
#### Distribution of Top 10 Cities



### Top 10 Ship Cities by Order Percentage

* **Bengaluru** leads with **8.7%** of total orders, followed by **Hyderabad (6.3%)**, and **Mumbai (4.8%)**, indicating strong customer bases in these metro cities.
* Cities like **New Delhi (4.5%)**, **Chennai (4.2%)**, and **Pune (3.0%)** also contribute significantly to orders.
* **Kolkata, Gurugram, Thane, and Lucknow** show **lower order percentages (below 2%)**, presenting potential growth opportunities through localized marketing or delivery enhancements.

#### Distribution of Top 10 States



### Top 10 Ship States by Order Percentage

* **Maharashtra** dominates with **17.3%** of total orders, followed by **Karnataka (13.4%)**, and **Tamil Nadu (8.9%)**, showcasing strong demand in southern and western regions.
* **Telangana (8.8%)**, **Uttar Pradesh (8.3%)**, and **Delhi (5.3%)** also contribute significantly, indicating pan-India customer engagement.
* States like **Kerala, West Bengal, Andhra Pradesh, and Gujarat** account for **below 6%** each, signaling untapped potential for growth through targeted campaigns or regional influencer partnerships.

## 

## Multivariate analysis

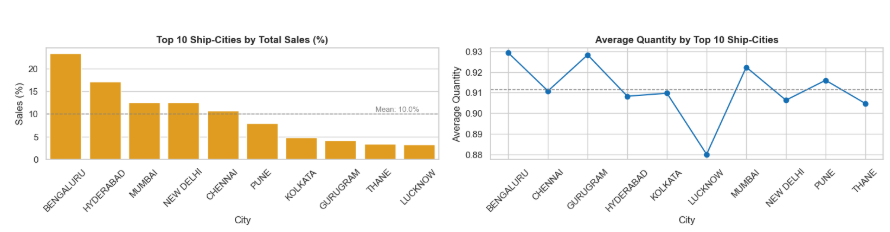
Target variables split by State

## 

**Top 10 Ship-States Analysis**

* **Sales (%):**
  + *Highest:* Maharashtra
  + *Above Avg:* Karnataka, Telangana, Uttar Pradesh, Tamil Nadu
  + *Below Avg:* Delhi, Kerala, West Bengal, Andhra Pradesh, Haryana
* **Average Quantity:**
  + *Highest:* Haryana, Karnataka
  + *Lowest:* Kerala
  + *Others:* Around average

Target variables split by Cities



**Top 10 Ship-Cities Analysis**

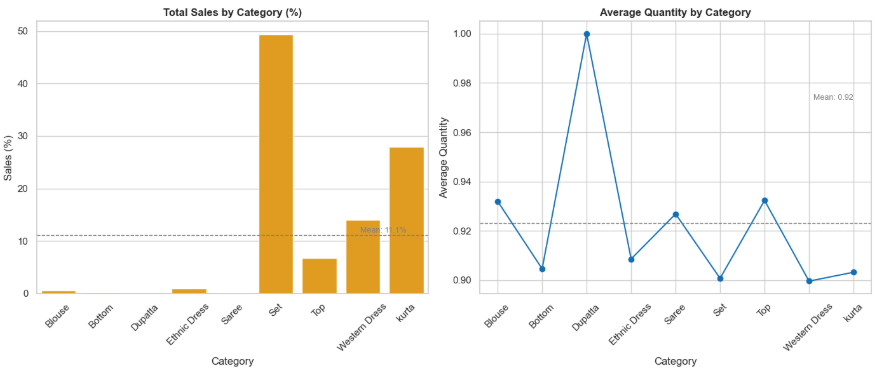
**Sales (%):**

* **Highest**: Bengaluru
* **Above Avg** Hyderabad, Mumbai, New Delhi, Chennai
* **Below Avg**: Pune, Kolkata, Gurugram, Thane, Lucknow

**Average Quantity:**

* **Highest**: Bengaluru, Gurugram, Mumbai
* **Lowest**: Lucknow
* **Others**: Around average

Target variables split by Category



**Top 10 Product Categories Analysis**

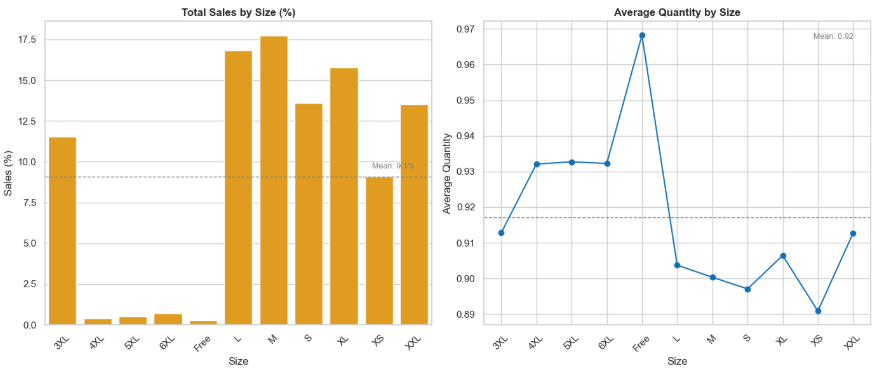
**Sales (%):**

* **Highest**: Set
* **Above Avg**: Set, Kurta, Western Dress
* **Below Avg**: Saree, Blouse, Dupatta, Top, Ethnic Dress, Bottom

**Average Quantity:**

* **Highest**: Dupatta
* **Lowest**: Set, Western Dress
* **Others**: Around average

Target variables split by Size



**Top 10 Sizes Analysis**

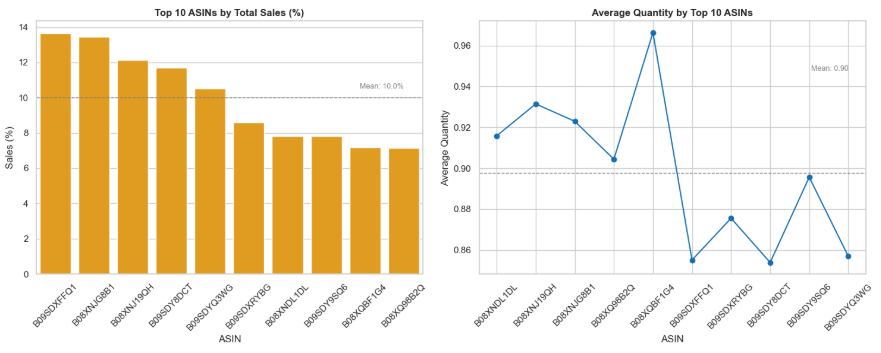
**Sales (%):**

* **Highest**: M
* **Above Avg**: L, S, XL, XXL, 3XL
* **Below Avg**: 4XL, 5XL, 6XL, Free, XS

**Average Quantity:**

* **Highest**: Free
* **Lowest**: XS,S, M
* **Others**: Around average

Target variables split by Top 10 ASIN



**Top 10 ASINs Analysis**

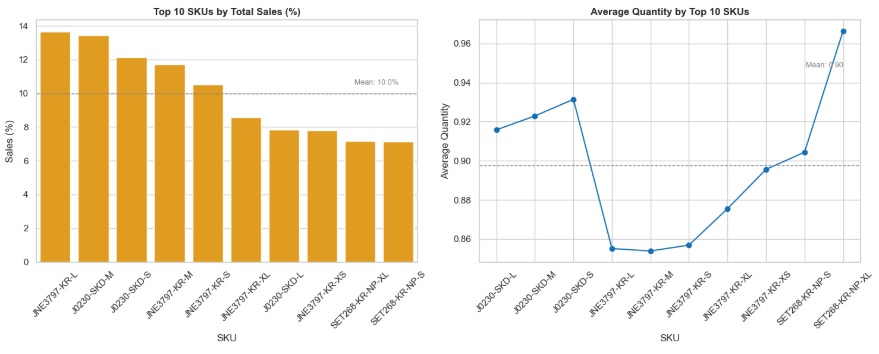
**Sales (%):**

* **Highest**: B09SDXFFQ1, B08XNJG8B1
* **Above Avg**: B08XNJ19QH, B09SDY8DCT, B09SDYQ3WG
* **Others**: Below Avg

**Average Quantity:**

* **Highest**: B08XQBF1G4
* **Above Avg**: B08XNDL1DL, B08XNJ19QH, B08XNJG8B1
* **Others**: Below Avg

Target variables split by Top 10 SKU



**Top 10 SKUs Analysis**

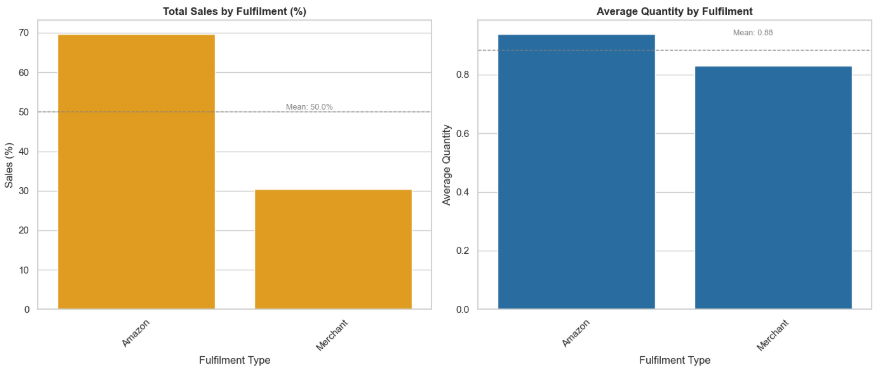
**Sales (%):**

* **Highest**: JNE7371-KR-L ,J0230-SKD-M
* **Above Avg**: J0230-SKD-S, JNE3797-KR-M, JNE3797-KR-M
* **Others**: Below Avg

**Average Quantity:**

* **Highest**: SET268-KR-NP-XL
* **Above Avg**: J2030-SKD-L, J0230-SKD-M, J0230-SKD-S
* **Others**: Below Avg

Target variables split by Fulfilment



**Fulfillment Type Analysis**

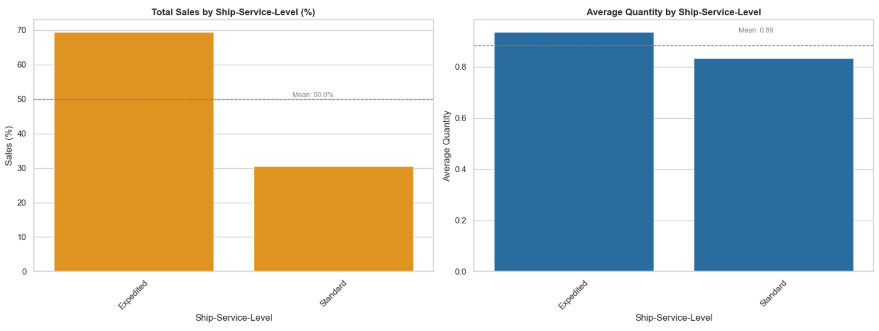
**Sales (%):**

* **Amazon Fulfilled**: ~70%
* **Merchant Fulfilled**: ~30%

**Average Quantity:**

* **Amazon**: ~0.91
* **Merchant**: ~0.89

Target variables split by Ship-Service



**Shipping Service Level Analysis**

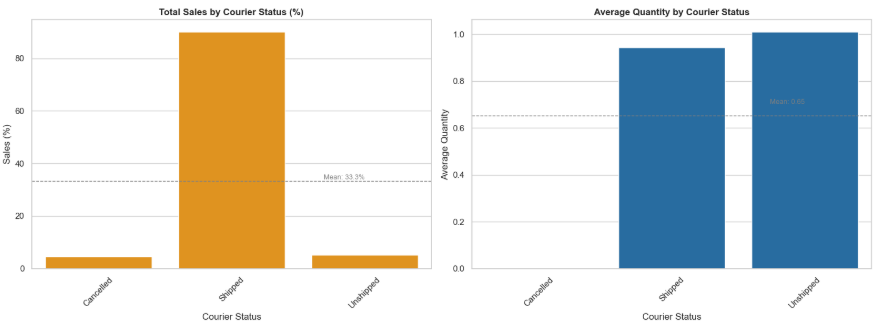
**Sales (%):**

* **Expedited**: ~70%
* **Standard**: ~30%

**Average Quantity:**

* **Expedited**: ~0.91
* **Standard**: ~0.89

Target variables split by Courier Status



**Courier Status Analysis**

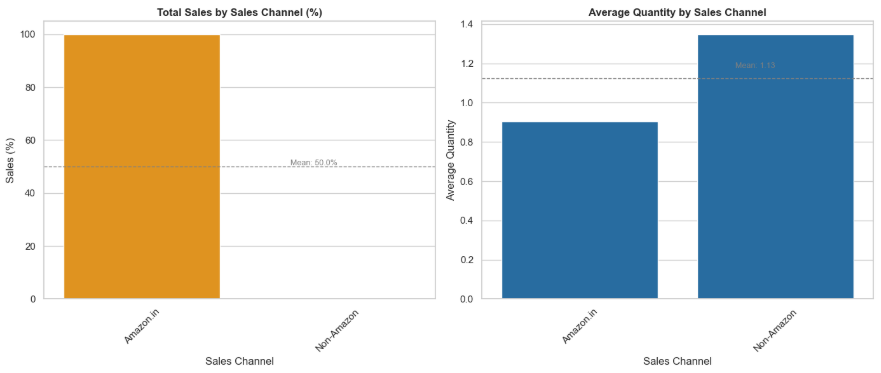
**Sales (%):**

* **Shipped**: ~90%
* **Unshipped**: ~5%
* **Cancelled:** ~5%

**Average Quantity:**

* **Highest**: Unshipped
* **Shipped**: Above Avg

Target variables split by Sales Channel



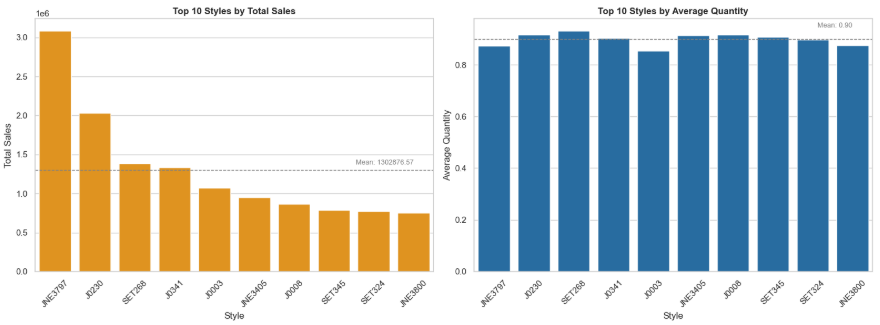
**Sales (%):**

* **Amazon.in**: ~100%
* **Non-Amazon**: ~0%

**Average Quantity:**

* **Highest**: Non-Amazon (~1.34)
* **Lowest**: Amazon.in (~0.92)

Target variables split by Top Styles



**Top 10 Styles Analysis**

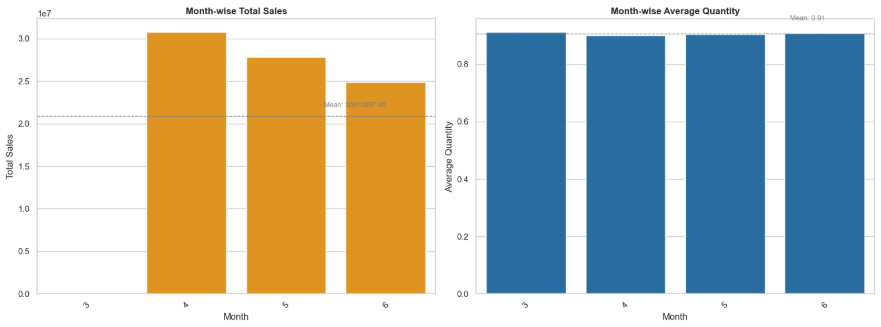
**Sales:**

* **Highest**: JNE1707 (~3M)
* **Above Avg**: J2020, SET288, J2041
* **Below Avg**: J3003, JNE2405, J2008, SET345, SET324, JNE3000

**Average Quantity:**

* **Highest**: SET288, J2020, SET345, JNE2405 (~0.93)
* **Lowest**: J3003 (~0.87)
* **Others**: Around average (~0.90)

Target variables split by Month



**Month-wise Analysis**

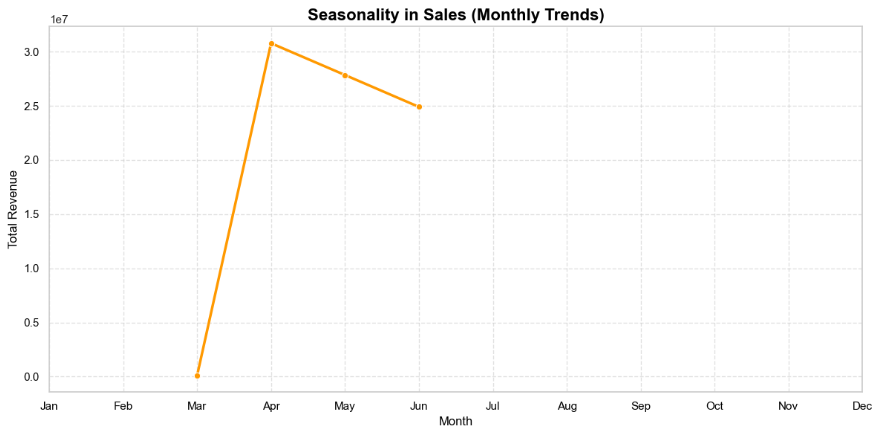
**Sales:**

* **Highest**: April (~31M)
* **Above Avg**: April, May
* **Below Avg**: June (~24M), March (~20M)

**Average Quantity:**

* **All Months**: Around average (~0.91)
  + Slight variation but generally consistent

Seasonality Trend Month wise

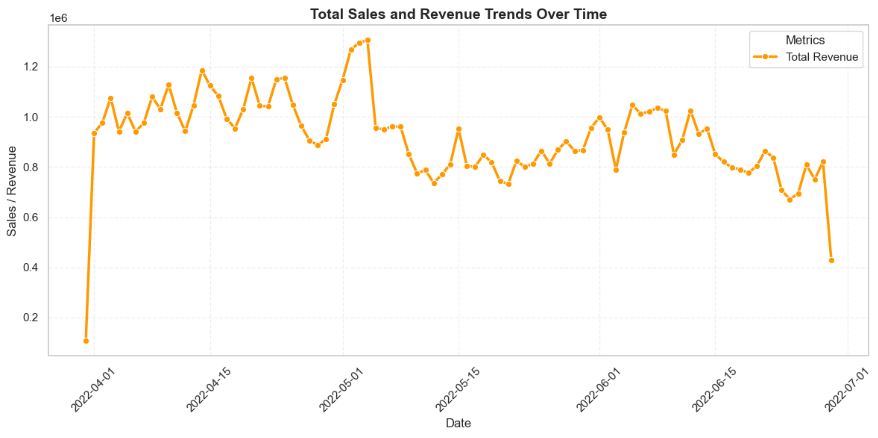
****

**Seasonality in Sales**

**Observation:**

* **Strong Surge**: Sales sharply **increased in April** (from near-zero in March to ~31M).
* **Gradual Decline**: A **steady drop** observed from **April → May → June**, indicating a **post-peak slowdown**.

Seasonality Trend Day wise

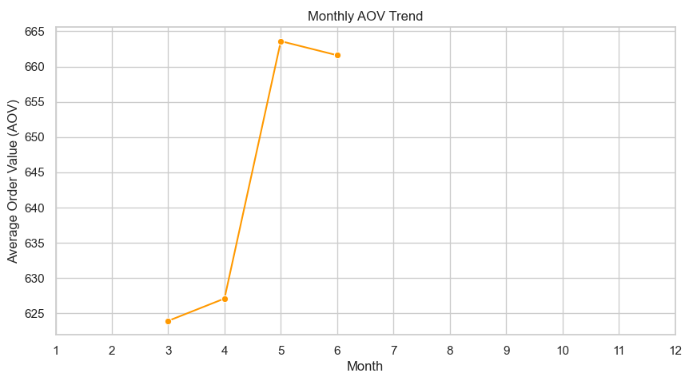


**Daily Sales & Revenue Trend (Apr–Jun 2022)**

**Insights:**

* **Early April Surge**: A **sharp spike** is seen at the **start of April**, marking the beginning of a high-revenue period.
* **Peak Revenue**: Late April shows the **highest daily revenue**, exceeding ₹1.3M.
* **May Dip**: Post-April, there’s a **notable drop** in revenue levels—though fluctuations continue.
* **June Plateau**: June maintains a **moderate but declining trend** with occasional brief spikes.
* **Month-End Dips**: Both **April 30th and June 30th** show significant revenue drops.

Monthly Average order value Trend

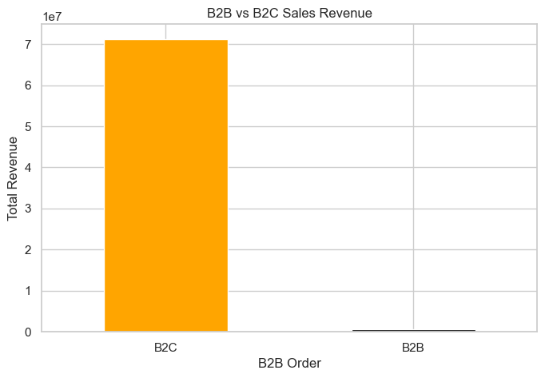


**Monthly AOV Trend (Mar–Jun)**

**Insights:**

* **Steady Start**: AOV begins at around **₹624** in **March**, with a small rise to ₹627 in **April**.
* **Sharp Surge**: A significant **spike in May** pushes AOV to **~₹664**, the highest across all months.
* **Minor Dip**: In **June**, AOV slightly drops but remains high at **~₹662**, indicating stability.

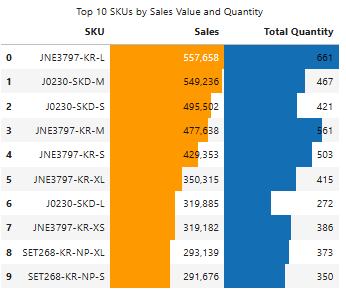
B2B Sales Trend



**B2B vs B2C Sales Revenue**

**Insights:**

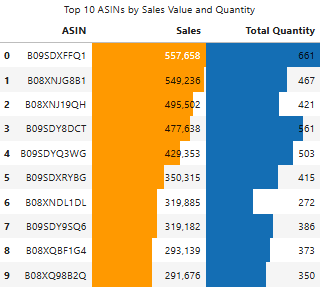
* **B2C Dominance**:
  + **B2C (Business-to-Consumer)** orders generate over **₹70 million** in total revenue.
  + Represents **almost 100%** of the total revenue.
* **Negligible B2B**:
  + **B2B (Business-to-Business)** contribution is extremely minimal, nearly **insignificant** in comparison.



**Top 10 SKUs by Sales & Quantity**

**Key Observations:**

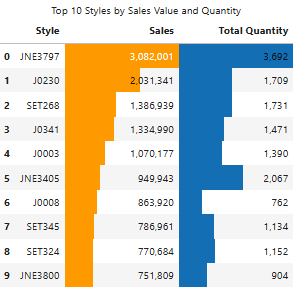
* **Best Performer**:
  + **SKU: JNE3797-KR-L** leads in both **Sales (~₹5.6L)** and **Quantity (661 units)**.
* **Top Contenders**:
  + **J0230-SKD-M** and **J0230-SKD-S** also show high sales (~₹5.5L and ₹4.9L) with solid quantities (467 and 421 units respectively).
* **Consistent Demand**:
  + **JNE3797** styles appear **multiple times**, indicating high product popularity and variant-level success.
* **Trailing SKUs**:
  + **SET268-KR-NP-XL** and **SET268-KR-NP-S** rank lower in both metrics, but still contribute notably to volume.



**Top 10 ASINs by Sales & Quantity**

**Key Insights:**

* **Top ASIN Performer**:
  + **B09SDXFFQ1** tops the chart with the **highest sales (~₹5.6L)** and **highest quantity (661 units)**.
* **Close Competitors**:
  + **B08XNJG8B1** and **B08XNJ19QH** follow closely with strong sales (>₹4.9L) and substantial order volumes.
* **Frequent Leaders**:
  + ASINs starting with **B09SD** appear multiple times, suggesting high-performing product lineups.
* **Mid-Tier Movers**:
  + ASINs like **B09SDXYRBG** and **B08XNDL1DL** show moderate sales (~₹3.5–3.2L) with proportionate volumes.
* **Volume Focused SKUs**:
  + **B08XQ88B2Q** has the **lowest sales** but still notable **quantity**, showing potential for price increase or bundling.



**Top 10 Styles by Sales & Quantity**

**Key Highlights:**

* **Top Performer – JNE3797**
  + Dominates with **₹30.8L in sales** and **3,692 units sold**, far ahead of others.
* **Strong Contenders**:
  + **J0230** and **SET268** rank 2nd and 3rd in sales (~₹20.3L & ₹13.9L respectively), each with ~1,700 units sold.
* **Balanced Styles**:
  + Styles like **J0341** and **J0003** maintain high sales and good volume, indicating both value and volume contributions.
* **High Volume, Moderate Sales**:
  + **JNE3405** sold **2,067 units**, yet ranks 5th in revenue (~₹9.5L), hinting at lower average selling price.
* **Styles Needing Review**:
  + **J0008** and **SET324** have lower sales and quantity, suggesting the need for promotions or product optimization.