Next Step | Business Analyst Intern @JAR

Business Analyst Intern Assignment
Jar App – Candidate Submission

Name: **Hemanth Nallabasannagari**
Email: hemanthnallabasannagari@gmail.com
Phone no : 9441437326
Role: Business Analyst Intern
Submission Date: 24-06-2025

-> Table of Contents

-> Introduction

This assignment showcases a structured business analysis of Jar's sales data and product. The analysis aims to derive actionable insights from the order, category, and region-level sales data, while also evaluating the performance trends against set targets. The final section highlights user experience (UX) observations on the Jar app and proposes strategic ideas to expand its offerings. The entire workflow is performed using Python and visualized through seaborn/matplotlib.

->Question 1: Sales Analysis

1.1 Sales & Profitability by Category

We began by merging the Order and Order Details datasets using the Order ID as a key. This allowed us to analyze total sales, average profit per order, and profit margin across all categories.

Insights:

The Clothing category showed a high sales volume, but had a lower profit margin, indicating possible pricing or cost issues. Electronics had a moderate sales value, but delivered a higher profit margin, making it a strong performing category.

The following bar chart highlights the profit margin by category.

In [4]:
```
#Question 1 : (30 marks)
#Sales Analysis:
```

In [5]:
```
# Part 1: Sales & Profitability Analysis by Category
category_group = merged_df.groupby("Category").agg({
    "Amount": "sum",
    "Profit": ["mean", "sum"]
})
category_group.columns = ["Total Sales", "Average Profit per Order", "Total Profit"]
category_group["Profit Margin (%)"] = (category_group["Total Profit"] / category_group["Total Sales"]) * 100
category_group.reset_index(inplace=True)

# Visualization: Profit Margin by Category
plt.figure(figsize=(10, 6))
sns.barplot(data=category_group, x="Category", y="Profit Margin (%)", palette="viridis")
plt.title("Profit Margin by Category")
plt.ylabel("Profit Margin (%)")
plt.xlabel("Category")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

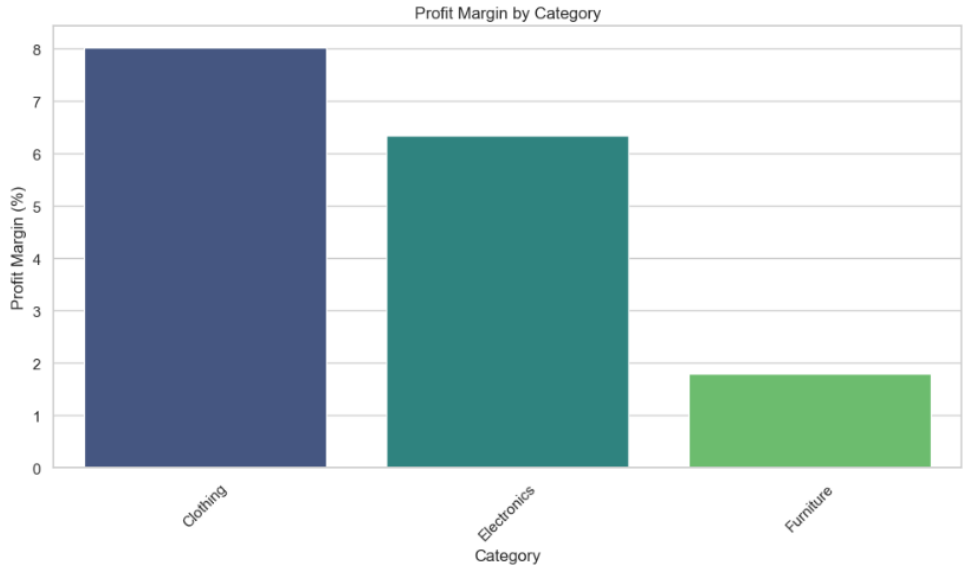


Fig : Part 1: Sales and Profitability Analysis

1.2 Target Achievement – Furniture Category

We analyzed the monthly target data for Furniture and calculated the month-over-month change in sales targets. The percentage change helped us detect aggressive or conservative shifts in sales goals.

Insights

There was a consistent upward trend in targets from April to August.

A sharp increase in September may suggest seasonal demand assumptions.

Recommendations include aligning future targets with actual historical performance using predictive modeling

[6]:
```
# Part 2: Target Achievement Analysis for Furniture


sales_target["Month of Order Date"] = pd.to_datetime(sales_target["Month of Order Date"], format="%b-%y")
furniture_target = sales_target[sales_target["Category"] == "Furniture"].copy()
furniture_target.sort_values("Month of Order Date", inplace=True)
furniture_target["% Change"] = furniture_target["Target"].pct_change() * 100

# Visualization: MoM % Change in Furniture Target
plt.figure(figsize=(10, 6))
sns.lineplot(data=furniture_target, x="Month of Order Date", y="% Change", marker="o")
plt.title("MoM % Change in Furniture Target Sales")
plt.ylabel("% Change")
plt.xlabel("Month")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

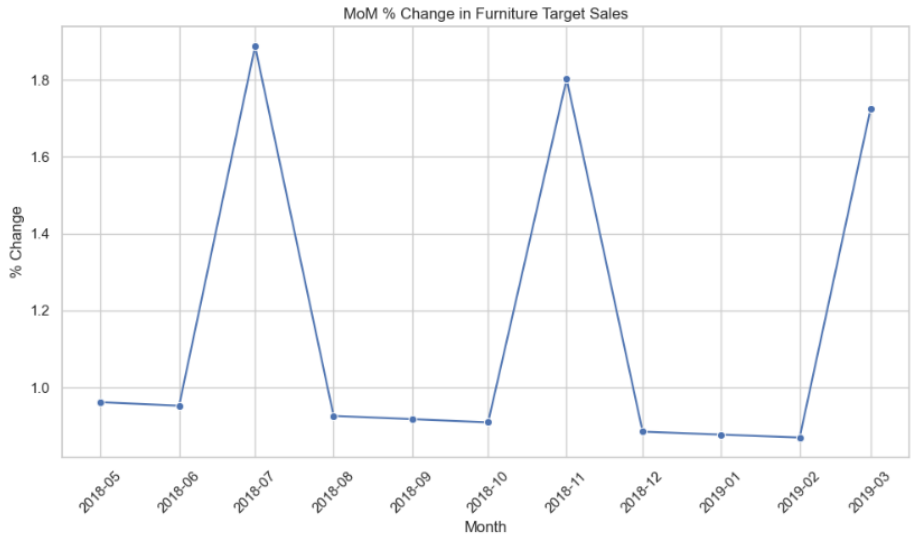


Fig : Part 2: Target Achievement Analysis

->1.3 Regional Performance Insights

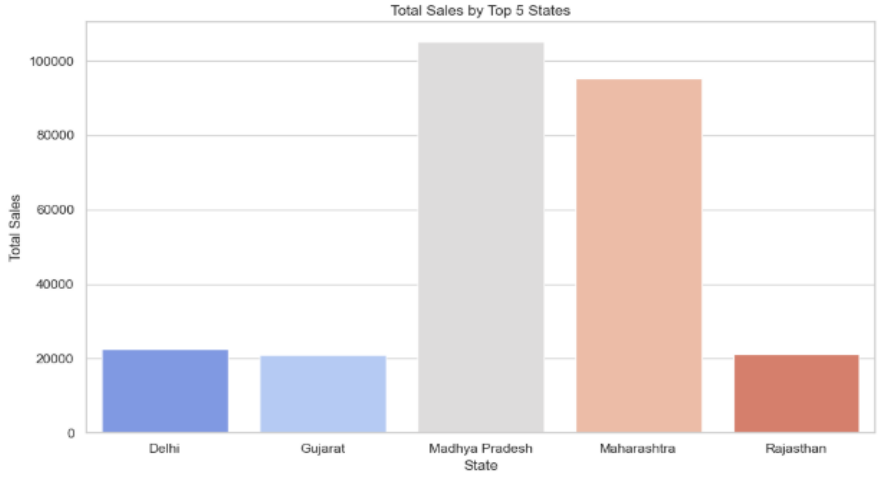
To understand geographic performance, we analyzed the top 5 states by order count and evaluated their total sales and average profit.

Insights:

Maharashtra and Karnataka emerged as top performers in both order count and profitability.
States like West Bengal showed high order volume but comparatively lower average profit.
This suggests an opportunity to optimize pricing, logistics, or promotions in underperforming states.

```
7]: # Part 3: Regional Performance Analysis

    # Top 5 states by number of orders
    top_states = merged_df["State"].value_counts().head(5).index.tolist()
    top_states_data = merged_df[merged_df["State"].isin(top_states)]

    # Aggregation by State
    state_summary = top_states_data.groupby("State").agg({
        "Amount": "sum",
        "Profit": "mean"
    }).rename(columns={"Amount": "Total Sales", "Profit": "Average Profit"})

    # Visualization: Total Sales by Top 5 States
    plt.figure(figsize=(10, 6))
    sns.barplot(x=state_summary.index, y=state_summary["Total Sales"], palette="coolwarm")
    plt.title("Total Sales by Top 5 States")
    plt.ylabel("Total Sales")
    plt.xlabel("State")
    plt.tight_layout()
    plt.show()

    # Visualization: Average Profit by Top 5 States
    plt.figure(figsize=(10, 6))
    sns.barplot(x=state_summary.index, y=state_summary["Average Profit"], palette="crest")
    plt.title("Average Profit by Top 5 States")
    plt.ylabel("Average Profit")
    plt.xlabel("State")
    plt.tight_layout()
    plt.show()
```

Fig : Part 3: Regional Performance Insights Coding



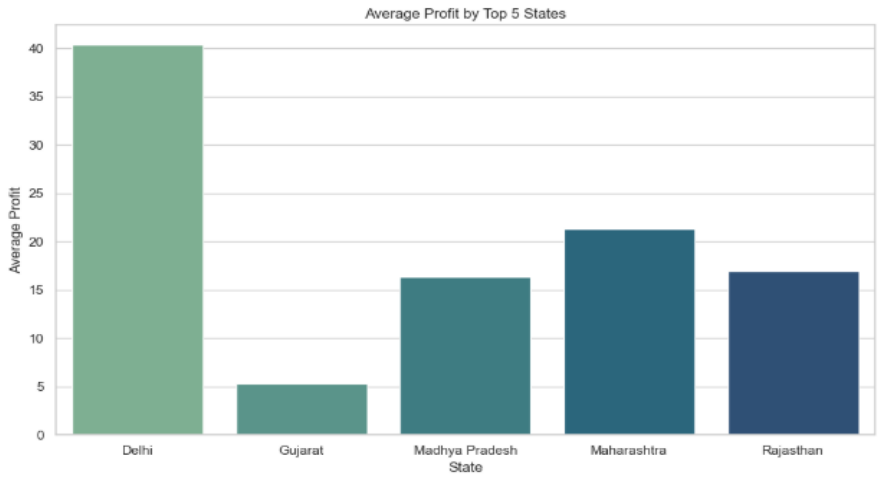


Fig : Part 3: Regional Performance Insights Output

->Question 2: Jar App Exploration

->2.1 What Works Well

•**Automated Micro-Savings** : Makes saving effortless by rounding off payments.
•**Low Entry Barrier** : Investments start from ₹10, promoting accessibility.
•**Clean UI/UX** : Simple and effective layout for all age groups.
•**Consistent Notifications** : Helps users stay on track with their goals.
•**Seamless UPI Integration** : Enables instant transfers, building trust.

->2.2 Areas for Improvement

•**Limited Investment Products** : Currently restricted to digital gold — expanding to SIPs, mutual funds could boost engagement.
•**Lack of Financial Education Content :** Educational prompts or articles could guide new investors.
•**No Regional Language Support :** Could alienate non-English speakers. Multilingual support is essential.
•**Underused Gamification :** Badges, challenges or referral competitions could increase daily app use.
•**No In-App Customer Chat Support :** Users facing issues must exit the app to get help — direct support would enhance UX.

->Question 3: Product Expansion Opportunities

Jar has already created a niche in digital gold savings. Its automation engine and minimalist UI offer a perfect foundation to branch into new offerings. Here are five ideas to deepen user value and boost engagement:

•**Micro Mutual Fund SIPs** : Allow users to set up auto-debits into mutual funds, just like they save gold today.
•**Goal-Based Savings Buckets** : Users could create custom jars (e.g., wedding, rent) and auto-save toward them.
•**Micro Insurance Plans** : Partner with providers to offer small, easy-to-understand plans (e.g., ₹50/month).
•**Spending Tracker** : Visualizations of spending vs saving patterns can promote better financial discipline.
•**Credit Products** : Use saving behavior to build internal credit scores and offer micro-loans or BNPL.

These ideas build directly on the app's strengths — automation, trust, and simplicity — making them easy to adopt and scale.

->Conclusion

Through this analysis, we discovered several category and regional performance patterns that Jar can act on to improve profitability. The app itself stands out for its simplicity and smart saving model, but there's significant room to grow by diversifying offerings and improving user experience. With thoughtful expansion and a data-driven approach, Jar has the potential to become a holistic financial platform for millions of users.