**Sample Submission - Retail Sales Data Analysis**

**1. Data Acquisition and Exploration**

Python

import pandas as pd

# Download and read the CSV file (replace with your download path)

data\_path = "retail\_sales.csv"

df = pd.read\_csv(data\_path)

# Explore the data

print(df.head()) # View the first few rows

print(df.tail()) # View the last few rows

print(df.info()) # Check data types and missing values

print(df.describe()) # Get summary statistics

**2. Data Cleaning with Generative AI**

* Missing Values: We identified missing values in the 'discount' column using df['discount'].isna().sum(). Generative AI suggested using df['discount'].fillna(df['discount'].mean()) to impute missing values with the mean. We implemented the following code (alternative approach):

Python

# Fill missing values in 'discount' with median

df['discount'].fillna(df['discount'].median(), inplace=True)

* Outlier Detection: We explored outliers in the 'sales\_amount' column using generative AI which suggested boxplot visualization and IQR (Interquartile Range) method for outlier detection. However, we opted for a simpler approach to identify potential outliers:

Python

# Identify potential outliers (values exceeding 1.5 times the IQR above the 75th percentile)

Q1 = df['sales\_amount'].quantile(0.25)

Q3 = df['sales\_amount'].quantile(0.75)

IQR = Q3 - Q1

potential\_outliers = df[df['sales\_amount'] > (Q3 + 1.5 \* IQR)]

# Further analysis or handling of outliers can be done here (optional)

print(f"Number of potential outliers: {len(potential\_outliers)}")

**3. Data Visualization with Generative AI**

* We chose to visualize the relationship between 'product\_category' and 'sales\_amount'.
* Generative AI suggested using a bar chart with Matplotlib. We implemented the following code with customizations:

Python

import matplotlib.pyplot as plt

# Create a bar chart

plt.figure(figsize=(8, 6))

plt.bar(df['product\_category'].unique(), df.groupby('product\_category')['sales\_amount'].sum())

plt.xlabel('Product Category')

plt.ylabel('Total Sales Amount')

plt.title('Sales by Product Category')

plt.xticks(rotation=45, ha='right') # Rotate category labels for better readability

plt.tight\_layout()

plt.show()

**4. Reflection**

* We addressed missing values in the 'discount' column by filling them with the median value. This approach assumes the missing values are randomly distributed and avoids introducing bias from the mean.
* While generative AI suggestions can be helpful starting points, it's important to understand the data and choose appropriate techniques. For example, we opted for a simpler outlier detection method for this initial exploration.
* The bar chart reveals that some product categories contribute significantly more to total sales compared to others. This could be a starting point for further analysis into high-performing categories or potential areas for improvement.

**Note:**

This is a sample submission and the data cleaning and visualization techniques might require adjustments based on the actual data in "retail\_sales.csv". The focus is on demonstrating the thought process and exploration with generative AI suggestions.