Synopsis Project: Retail Sales Data Analysis

Project Title: Retail Sales Data Analysis

The aim of this project is to conduct a comprehensive analysis of retail sales data to gain actionable insights into various business parameters like sales trends, product performance, and the impact of marketing efforts. By leveraging the power of SQL and Python (with libraries like Pandas, Matplotlib, Seaborn), we aim to explore patterns in sales data, determine factors influencing sales, and provide data-driven recommendations for improving sales strategies. This analysis covers key aspects such as daily and seasonal trends, correlation between discounts and sales, and regional performance of retail stores.

Data Overview

The dataset used in this project is sourced from a retail sales database and contains information related to various products and their sales. Key columns in the dataset include:

- Store_ID: Identifies the store where the sale took place.
- Product_ID: Identifies the product sold.
- Sale_Date: The date on which the sale occurred.
- Units_Sold: Number of units sold in each transaction.
- Sales_Revenue_USD: The total revenue generated from the sale.
- Discount_Percentage: The discount applied during the sale.
- Marketing_Spend_USD: The budget spent on marketing the product.
- Store_Location: Geographical location of the store.
- Product_Category: The category of the product.
- Day_of_the_Week: The day when the sale occurred.
- Holiday_Effect: A binary indicator (0 or 1) to denote if the sale happened on a holiday or not.

Data Cleaning and Preparation

The first phase of the project involves preparing the data for analysis. Data cleaning operations are essential to ensure accuracy in our analysis and results. This includes handling missing or incomplete data.

Missing Data Handling:

- Missing values in key columns such as Store_ID, Product_ID, Sale_Date, Units_Sold, and Sales_Revenue_USD were identified and addressed.
- For missing Store_ID and Product_ID, 'Unknown' was used as a placeholder.
- For missing Sale_Date, a default date of '1970-01-01' was assigned.
- Units_Sold and Sales_Revenue_USD were filled with zeroes to ensure no sales data was omitted from analysis.

Creating a Table for Missing Values: An important step was creating a table that captured rows with missing data, which helps in ensuring that no valuable insights are overlooked. The SQL query used to create the table is as follows:

```
CREATE TABLE IF NOT EXISTS MissingValues AS

SELECT *

FROM SalesData

WHERE Store_ID IS NULL

  OR Product_ID IS NULL

  OR Sale_Date IS NULL

  OR Units_Sold IS NULL

  OR Sales_Revenue_USD IS NULL;
```

Exploratory Data Analysis (EDA)

Once the data was cleaned, the focus shifted to exploratory data analysis (EDA). The goal of EDA is to understand the data better, uncover hidden patterns, and test hypotheses.

1. **Sales Trends Over Time**:
    - We performed a time-series analysis to observe trends in the Sales_Revenue_USD and Units_Sold over different days of the week. This helps to understand the cyclical nature of retail sales.

- SQL queries were used to calculate average sales revenue and units sold per day, which were then plotted using bar charts to visualize trends.

SQL Query for Average Sales by Day:

```
SELECT Day_of_the_Week, AVG(Sales_Revenue_USD) AS Avg_Sales_Revenue
FROM SalesData
GROUP BY Day_of_the_Week
ORDER BY FIELD(Day_of_the_Week, 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday');
```

2. Impact of Discounts on Sales:
    - We explored the correlation between discount percentage and sales revenue. A scatter plot was generated to visualize this relationship, helping to understand if higher discounts correlate with higher sales.
    - The SQL query to retrieve data for this analysis aggregates sales and discounts by product and store location.

3. Sales by Store Location:
    - A key focus was on understanding which store locations generated the highest sales. Using SQL queries, the sales revenue for each store location was summed up.
    - A pie chart was plotted to visualize the percentage contribution of each store location to the total sales revenue.

SQL Query for Sales by Location:

```
SELECT Store_Location, SUM(Sales_Revenue_USD) AS Total_Sales_Revenue
FROM SalesData
GROUP BY Store_Location;
```

4. **Seasonality and Holidays**:
    - An analysis was performed to examine the impact of holidays on sales. The Holiday_Effect column was used to segregate sales data into holiday and non-holiday sales, and the average sales revenue was calculated for both categories.

- A bar chart was plotted to visualize the difference in sales revenue during holidays vs. non-holidays.

SQL Query for Sales Based on Holiday Effect:

```
SELECT Holiday_Effect, AVG(Sales_Revenue_USD) AS Avg_Sales
FROM SalesData
GROUP BY Holiday_Effect;
```

5. **Product Performance**:
    - The performance of different product categories was evaluated by grouping the data by Product_Category and calculating the average sales revenue for each category. A bar chart was generated to show which categories performed the best in terms of revenue generation.

SQL Query for Product Category Sales:

```
SELECT Product_Category, AVG(Sales_Revenue_USD) AS Avg_Sales_Revenue
FROM SalesData
GROUP BY Product_Category;
```

Insights and Recommendations

Based on the EDA, several key insights emerged:

1. **Seasonality**: Sales were significantly higher on weekends, with Saturdays and Sundays showing the highest revenue. This trend is important for inventory management and staffing optimization.
2. **Discounts and Sales**: A strong correlation was found between discounts and sales revenue, with higher discounts leading to increased sales. However, the effect varied by product category, and not all products responded equally to discounts.
3. **Store Performance**: Locations in urban areas outperformed rural locations in terms of total sales revenue. Targeted marketing and promotional strategies may help boost sales in less performing locations.

Conclusion

The project successfully leveraged SQL and Python to conduct a thorough analysis of retail sales data. By exploring daily trends, discount effects, store location performance, and holiday impacts, we uncovered actionable insights that can be utilized by the retail business to optimize sales strategies, enhance inventory management, and improve marketing efforts. The combination of SQL for data extraction and Python for visualization made this analysis efficient and effective, providing clear and understandable insights for decision-making.