Predictive Sales Analysis and Optimization

Task Overview:

You are tasked with developing a **Predictive Sales Analysis and Optimization System** for a retail company. The project involves **data cleaning**, **statistical modeling**, **forecasting**, and **reporting** using **Excel**, **SQL**, **NumPy**, **Pandas**, and **Python**.

Part 1: Data Cleaning and Transformation (Python with Pandas and NumPy)

Subtasks:

1. **Data Cleaning:**
    - Detect and handle missing values using different strategies like interpolation, mean imputation, or forward-fill.
    - Identify and remove duplicate records.
    - Detect and remove outliers using **Z-Score** and **IQR (Interquartile Range)**.
2. **Feature Engineering:**
    - Create new columns for:
        - **Profit Margin:** Assume a dynamic profit margin based on product categories (Electronics - 40%, Clothing - 50%, etc.).
        - **Month and Weekday:** Extract from 'OrderDate'.
        - **Cumulative Sales:** Calculate cumulative sales by category using Pandas.
3. **Trend Segmentation:**
    - Identify high-performing products by calculating sales growth percentages using **NumPy**.
    - Tag products as **Growing**, **Stable**, or **Declining** based on trends.

Part 2: SQL Database Analysis (PostgreSQL)

Subtasks:

1. Import the cleaned dataset into PostgreSQL.
2. Create normalized tables: Customers, Orders, Products, and Sales.
3. Write complex SQL queries to:
    - Identify seasonal trends by month.
    - Calculate customer lifetime value (CLV) based on repeat purchases and spending.
    - Find products with declining sales in specific regions over time.
    - Generate sales forecasts using SQL Window Functions for moving averages.
    - Rank customers based on revenue contribution using **RANK()** and **DENSE_RANK()**.
    - Determine sales anomalies by calculating deviations from mean values.

Part 3: Statistical Analysis and Forecasting (Python with NumPy and Pandas)

Subtasks:

1. **Descriptive Statistics:**
    - Calculate **mean**, **median**, **variance**, and **standard deviation** of sales data.
    - Use correlation coefficients to analyze relationships between **UnitPrice** and **Quantity Sold**.
2. **Predictive Modeling:**
    - Build a **Linear Regression** model to forecast future sales using Python.
    - Evaluate model performance using $\mathbf{R^2}$, **RMSE**, and **MAE**.
    - Use **ARIMA** for time-series forecasting of monthly sales.
3. **Anomaly Detection:**
    - Detect outliers in sales data using Z-Scores and visualize anomalies with **Matplotlib/Seaborn**.
    - Use clustering algorithms like **K-Means** to group customers by spending behavior.

4. **Hypothesis Testing:**
    - Perform hypothesis tests to determine if sales performance varies significantly across regions.
    - Use **t-tests** and **chi-square tests** to validate hypotheses.

Part 4: Reporting and Visualization (Excel and Python)

Subtasks:

1. Export SQL query results to Excel.
2. Create **Pivot Tables** and **Slicers** in Excel for interactive analysis:
    - Region-wise performance dashboards.
    - Customer segmentation and performance charts.
3. Use Python libraries (**Matplotlib**, **Seaborn**, **Plotly**) to generate visualizations:
    - Heatmaps showing sales patterns across time.
    - Bar charts and line graphs for trends and anomalies.

Additional Challenges:

1. Automate report generation and email it as a PDF using **Python**.
2. Implement machine learning models (e.g., Decision Trees) to classify products into **High Demand** or **Low Demand** categories.
3. Optimize pricing strategies by analyzing elasticities using regression techniques.
4. Use SQL triggers and stored procedures to automate data updates.

Expected Deliverables:

1. Cleaned and preprocessed dataset in SQL.
2. SQL queries with insights and optimizations.
3. Python scripts for analysis, forecasting, and anomaly detection.
4. Excel dashboards with pivot tables and charts.
5. Documentation with conclusions and improvement strategies.