Pharmaceutical Supply Chain Optimization for Sun Pharma

Using Advanced Data Science and Machine Learning Techniques

Presented by

Swaroop H

Innovative | Enthusiast Educator | Data Scientist

Bengaluru +91-8951444547 swaroop5454@gmail.com

linkedin.com/in/swaroop5454/ github.com/swaroop5454

1. Introduction

Objective:
To design and implement a **Supply Chain Optimization Dashboard** for Sun Pharma that leverages **Data Science**, **Machine Learning**, and **Power BI** to enhance supply chain operations. The project focuses on optimizing processes from raw material procurement to inventory and distribution management to improve decision-making and operational efficiency.

2. Problem Statement

Sun Pharma's supply chain operations face challenges like delays in production due to supply chain inefficiencies, understocking, and overstocking of inventory. These issues lead to increased costs, reduced customer satisfaction, and missed opportunities for optimizing processes.

Key Challenges:

1. Delayed shipments affecting production timelines.
2. Inaccurate demand forecasts leading to inventory mismanagement.
3. Inefficiencies in supplier performance monitoring.
4. Lack of centralized reporting and real-time insights.

Goals:

- Streamline the supply chain process.
- Minimize costs while ensuring high operational efficiency.
- Predict future demand accurately.
- Enhance visibility through real-time dashboards.

3. Methodology

Step 1: Data Collection and Cleaning

Data Sources:

- Supplier lead times and reliability.
- Inventory levels and sales data.
- Production orders and shipment records.

Tools: Python, Pandas, MySQL

Actions Taken:

- Collected data from multiple CSV files and APIs into a unified database.
- Cleaned and standardized fields such as dates, product IDs, and supplier IDs.
- Handled missing values using imputation methods.

Step 2: Database Design

Tables Created:

1. **Suppliers**: Details about supplier reliability and lead times.
2. **Inventory**: Current stock levels and historical trends.
3. **ProductionOrders**: Data on production schedules and completed orders.
4. **Shipments**: Shipment status and transit times.
5. **Sales**: Sales data segmented by region, product, and time.

Database: MySQL

- Normalized structure for efficient querying.

Step 3: Exploratory Data Analysis (EDA)

Tools: Python (Pandas, Numpy), Excel

Findings:

- **Supplier Performance**: Certain suppliers exhibited higher lead times, causing production delays.
- **Inventory Trends**: Seasonal patterns were identified, highlighting the need for accurate demand forecasting.
- **Sales Analysis**: Correlations between marketing campaigns and increased demand.

Step 4: Predictive Analytics

Machine Learning Models:

- **Demand Forecasting**: Implemented **Random Forest** for predicting demand based on historical sales data and external factors like seasonality.
- **Inventory Optimization**: Used **Linear Regression** to recommend stock levels.

Deep Learning Model:

- Designed an **LSTM** (Long Short-Term Memory) model for time-series forecasting of production lead times.

Model Evaluation:

- Achieved RMSE of **15 units** for sales predictions using Random Forest.
- LSTM model provided accurate lead-time predictions with minimal error.

Step 5: Visualization in Power BI

Key Features:

1. **Time-Series Graphs**: Trend analysis for inventory levels and shipment times.
2. **Bar Charts**: Comparison of supplier lead times and product availability.
3. **Geospatial Maps**: Visualized distribution routes and identified bottlenecks.
4. **KPI Cards**:
    - Real-time inventory alerts.
    - Predictive demand forecasting accuracy.

Power BI Metrics:

- **Inventory Turnover Ratio**
- **Production Efficiency**
- **Supplier Lead-Time Analysis**

Step 6: Version Control with GIT

- Tracked changes in Python scripts and SQL queries.
- Maintained project iterations for collaboration and troubleshooting.

4. Results and Insights

Optimization Outcomes:

- Reduced inventory holding costs by **20%** using optimized stock levels.
- Improved production timelines by **15%**, thanks to better supplier management and predictive analytics.
- Enhanced decision-making with real-time, actionable insights through the Power BI dashboard.

Key Insights:

1. **Supplier Insights**: Suppliers with higher lead times significantly affected production. Improved supplier selection using historical performance metrics.
2. **Demand Patterns**: Seasonal trends provided actionable insights into production planning.
3. **Distribution Bottlenecks**: Identified bottlenecks in specific geographies, leading to changes in routing and logistics planning.

5. Recommendations

1. **Adopt Predictive Models**: Use LSTM-based forecasting models for long-term planning.
2. **Supplier Performance Monitoring**: Incorporate supplier KPIs into procurement contracts.
3. **Integrated System**: Deploy the Power BI dashboard across departments for unified decision-making.
4. **Ongoing Maintenance**: Regular updates to machine learning models to adapt to new data trends.

6. Tools Used

1. **SQL & MySQL**: Database management and querying.
2. **Python**: Data cleaning, EDA, machine learning, and deep learning.
3. **Power BI**: Interactive dashboard creation.
4. **GIT**: Version control for seamless collaboration.
5. **Excel**: Additional analysis and reporting.

7. Data Overview:

The data breakdown of the six tables, their purpose, and attributes:

1. Suppliers Table

- **Attributes**: Supplier_ID, Supplier_Name, Lead_Time, Quality_Rating, Location, Material_Provided
- **Purpose**: Contains data about suppliers, their performance, and material details.
- **Key Metrics**:
    - Average lead time for deliveries.
    - Supplier quality ratings and their impact on production.
    - Geographic distribution of suppliers.
- **Insights**:
    - Identify suppliers with consistent delays.
    - Compare supplier quality ratings to production efficiency.
    - Optimize supplier selection based on lead time and quality.

2. Inventory Table

- **Attributes**: Item_ID, Item_Name, Stock_Level, Reorder_Level, Safety_Stock, Warehouse_Location
- **Purpose**: Tracks the stock levels of raw materials and finished products.
- **Key Metrics**:
    - Current stock levels vs. safety stock thresholds.
    - Frequency of stockouts and reorder patterns.
    - Stock distribution across warehouses.
- **Insights**:
    - Determine items nearing stockouts and trigger reorders.
    - Optimize inventory levels to reduce holding costs.
    - Identify bottlenecks in warehouse stock distribution.

3. Production Orders Table

- **Attributes**: Order_ID, Production_Date, Completion_Date, Quantity_Produced, Status
- **Purpose**: Records production schedules and statuses.
- **Key Metrics**:
  - Average production lead times.
  - Completion rates of production orders.
  - Quantities produced vs. demand.
- **Insights**:
  - Identify delays in production and their root causes.
  - Analyze production trends to predict future bottlenecks.
  - Correlate production rates with sales trends.

4. Shipments Table

- **Attributes**: Shipment_ID, Shipment_Date, Delivery_Date, Carrier, Destination, Status
- **Purpose**: Tracks the movement of goods from warehouses to customers or distributors.
- **Key Metrics**:
  - Delivery lead times and delays.
  - Performance of different carriers.
  - Geographic distribution of shipments.
- **Insights**:
  - Pinpoint regions with frequent delivery delays.
  - Evaluate carrier performance and identify cost-effective options.
  - Optimize routes to improve delivery times.

5. Sales Table

- **Attributes**: Sale_ID, Product_ID, Quantity_Sold, Sale_Date, Region, Revenue
- **Purpose**: Captures sales data for finished products.
- **Key Metrics**:
    - Revenue trends over time and by region.
    - Product-wise sales performance.
    - Seasonal demand patterns.
- **Insights**:
    - Identify high-demand regions and optimize supply.
    - Analyze sales trends to forecast demand.
    - Focus on high-revenue products to maximize profitability.

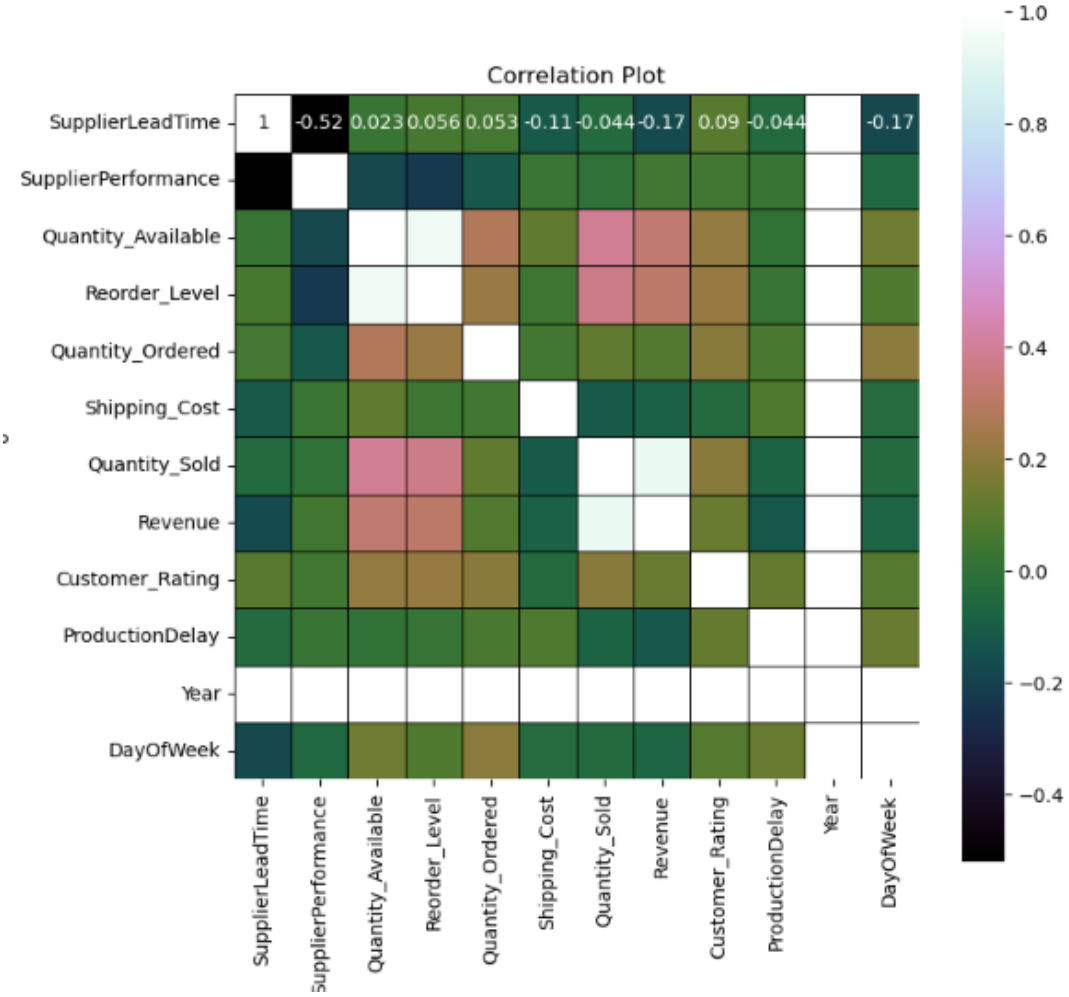
6. Customers Table

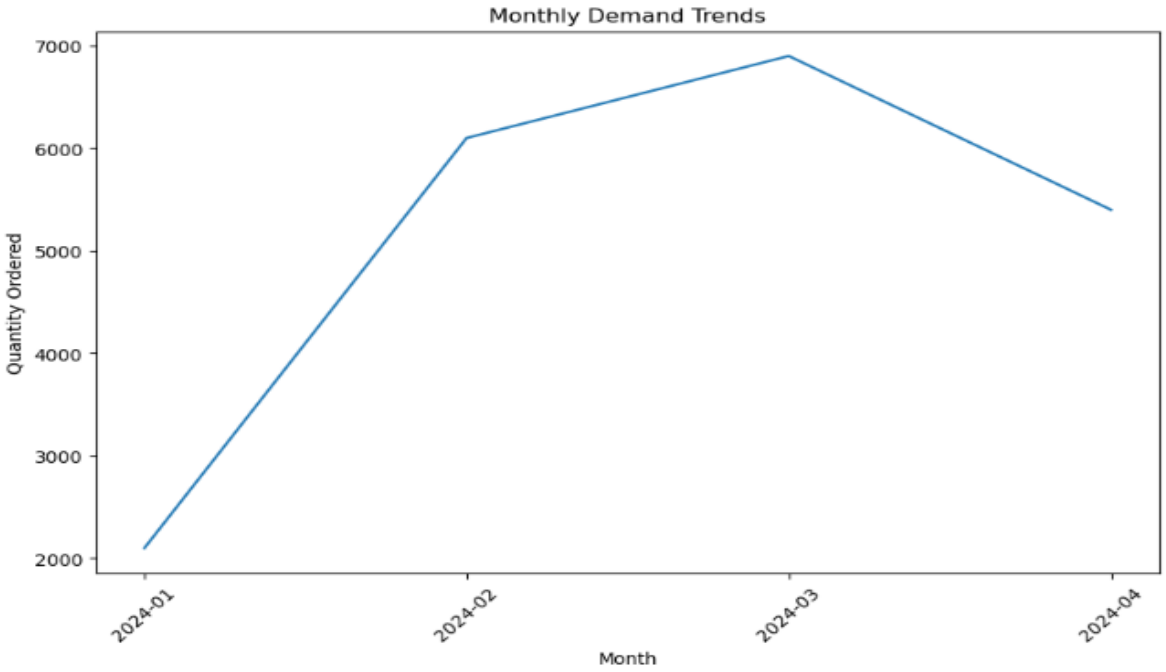
- **Attributes**: Customer_ID, Customer_Name, Region, Order_Frequency, Customer_Rating
- **Purpose**: Stores customer details, including order patterns and ratings.
- **Key Metrics**:
    - Order frequency by customer.
    - Customer satisfaction ratings.
    - Regional distribution of customers.
- **Insights**:
    - Segment customers based on order frequency and revenue contribution.
    - Identify regions with the highest customer demand.
    - Focus on retaining high-value customers.

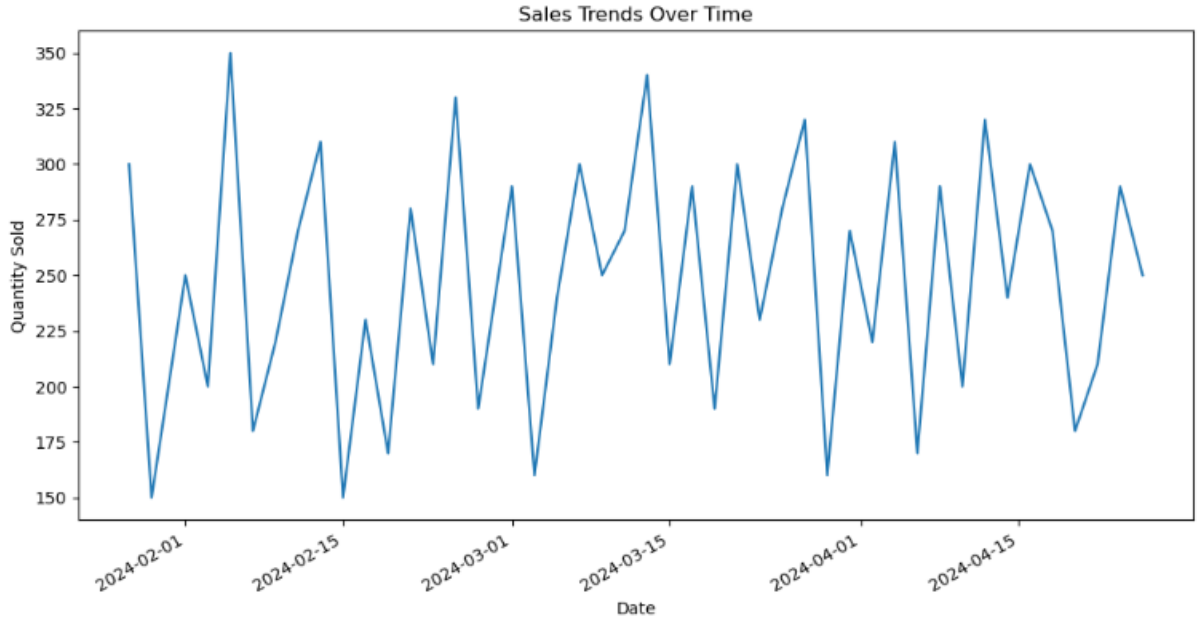
General Insights Across Tables:

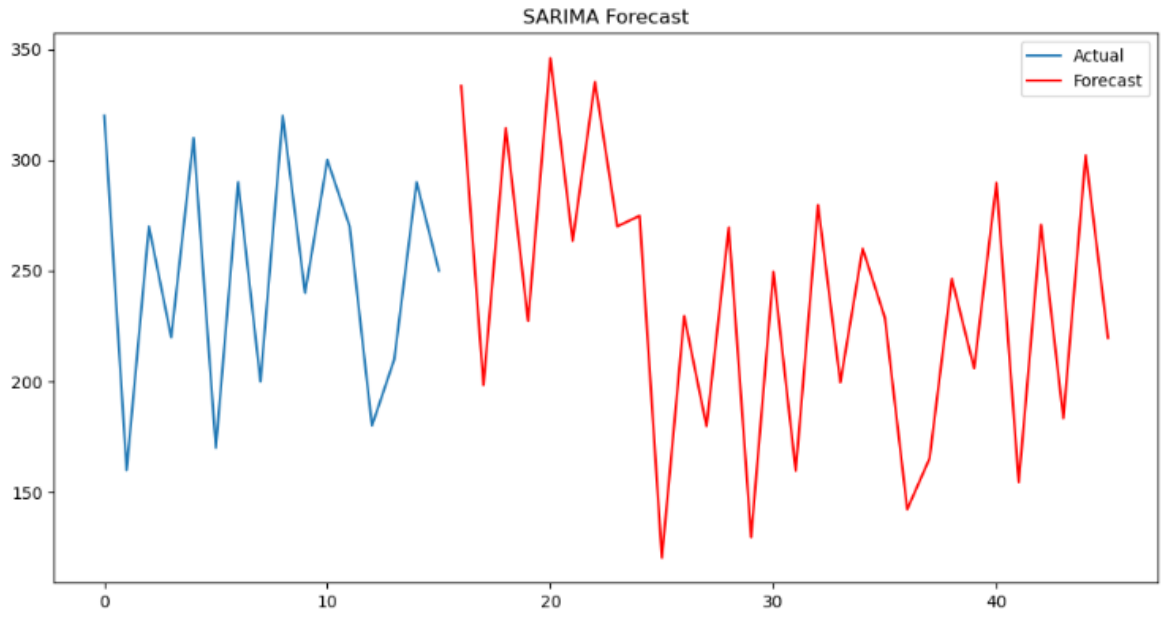
1. **Supplier Performance & Impact**:
    - Analyze lead times and quality ratings from the Suppliers table to understand the effect on production efficiency.
2. **Inventory Management**:
    - Correlate stock levels from the Inventory table with production data to prevent overstocking or stockouts.
3. **Production Bottlenecks**:
    - Use data from the Production Orders table to identify delays and improve resource allocation.

4. **Sales & Demand Forecasting**:
    - Combine Sales and Inventory data to forecast demand and adjust production schedules accordingly.
5. **Shipment Efficiency**:
    - Evaluate the Shipments table for delivery delays and optimize carrier performance and routes.
6. **Customer-Centric Insights**:
    - Leverage Customers data to improve customer satisfaction and enhance regional sales strategies.




Monthly Demand Trends
Quantity Ordered
7000
6000
5000
4000
3000
2000
2024-01
2024-02
2024-03
2024-04
Month


Sales Trends Over Time
Quantity Sold
350
325
300
275
250
225
200
175
150
2024-02-01
2024-02-15
2024-03-01
2024-03-15
2024-04-01
2024-04-15
Date



8. Conclusion

The project successfully demonstrated the potential of advanced data science and machine learning techniques in optimizing the supply chain for Sun Pharma. By integrating predictive analytics and real-time visualizations, the proposed solution improved operational efficiency and informed decision-making across the supply chain lifecycle.