## **Dự Đoán Doanh Số Nhãn Hàng**

### **I. Công Cụ Sử Dụng**

1. **Python & Jupyter Notebooks/VScode**: Nền tảng chính cho việc phân tích.

* Python: Python có cú pháp đơn giản, dễ đọc và khả năng hỗ trợ mạnh mẽ trong phân tích dữ liệu cũng như học máy. Với tính linh hoạt cao, Python cho phép tích hợp các thư viện chuyên dụng để xử lý dữ liệu doanh số, xây dựng mô hình dự đoán và trực quan hóa kết quả. Đây là lựa chọn lý tưởng để xử lý các tác vụ phức tạp trong dự án này mà không cần viết lại mã từ đầu.
* Jupyter Notebook: sử dụng Jupyter Notebook để viết, kiểm tra và trình bày mã nguồn một cách trực quan. Công cụ này cho phép kết hợp mã Python, văn bản giải thích, và hình ảnh trực quan hóa dữ liệu trong cùng một tài liệu. Trong dự đoán doanh số, Jupyter Notebook giúp dễ dàng thử nghiệm các thuật toán, chạy code từng bước, và trình bày kết quả phân tích (ví dụ: biểu đồ doanh số theo thời gian) một cách rõ ràng và dễ hiểu.
* Visual Studio Code (VS Code): là một trình soạn thảo mã nguồn miễn phí, đa nền tảng, được phát triển bởi Microsoft. Đây là một công cụ mạnh mẽ hỗ trợ nhiều ngôn ngữ lập trình khác nhau, trong đó có Python – ngôn ngữ phổ biến trong phân tích dữ liệu và dự đoán doanh số nhãn hàng.

1. **Thư viện chính**:

* **Pandas, NumPy:** Xử lý dữ liệu.
* Pandas đóng vai trò quan trọng trong việc quản lý và xử lý bộ dữ liệu doanh số. Với hai cấu trúc dữ liệu chính là **Series** (dùng cho dữ liệu một chiều như doanh số theo tháng) và **DataFrame** (dùng cho dữ liệu đa chiều như bảng doanh số theo nhãn hàng và thời gian), Pandas giúp dễ dàng lọc, nhóm, và tính toán thống kê trên dữ liệu. Ví dụ, có thể sử dụng Pandas để tính doanh số trung bình của từng nhãn hàng hoặc xác định xu hướng bán hàng theo thời gian, tạo nền tảng cho mô hình dự đoán.
* NumPy là thư viện cốt lõi hỗ trợ các phép toán trên dữ liệu số, đặc biệt là các mảng và ma trận – yếu tố quan trọng khi làm việc với dữ liệu doanh số. Với NumPy, có thể thực hiện các phép tính nhanh chóng như tổng hợp doanh số, tính trung bình, hoặc chuẩn hóa dữ liệu để đưa vào mô hình dự đoán. Tính năng **broadcasting** và hiệu suất cao của NumPy giúp xử lý các tập dữ liệu lớn một cách hiệu quả.
* **Matplotlib, Seaborn, Plotly:** Trực quan hóa.
* Matplotlib là thư viện được sử dụng để tạo các biểu đồ cơ bản nhằm trực quan hóa dữ liệu doanh số. Ví dụ: có thể vẽ biểu đồ đường để thể hiện xu hướng doanh số theo thời gian hoặc biểu đồ cột để so sánh doanh số giữa các nhãn hàng. Với tính linh hoạt cao, Matplotlib cho phép tùy chỉnh màu sắc, nhãn, và kiểu biểu đồ, giúp tôi trình bày kết quả một cách rõ ràng và dễ hiểu trong báo cáo.
* Seaborn: được xây dựng dựa trên Matplotlib, giúp tạo ra các biểu đồ đẹp mắt và phức tạp hơn với ít mã hơn. Ví dụ, sử dụng Seaborn để vẽ biểu đồ nhiệt (heatmap) nhằm phân tích mối quan hệ giữa doanh số và các yếu tố khác, hoặc biểu đồ phân tán (scatter plot) để phát hiện xu hướng và ngoại lệ trong dữ liệu. Seaborn nâng cao tính thẩm mỹ và chuyên nghiệp của các hình ảnh trong báo cáo.
* Plotly: mang lại khả năng tạo biểu đồ tương tác, rất hữu ích khi muốn trình bày dữ liệu doanh số một cách sinh động. Ví dụ, có thể tạo biểu đồ đường tương tác cho phép người xem phóng to, thu nhỏ, hoặc di chuột để xem chi tiết doanh số từng tháng. Điều này đặc biệt hữu ích khi báo cáo được chia sẻ dưới dạng kỹ thuật số, giúp người đọc tương tác trực tiếp với dữ liệu.
* **Scipy.stats, Statsmodels:** Phân tích thống kê.
* Scipy.stats cung cấp các công cụ thống kê để kiểm tra và hiểu rõ hơn về dữ liệu doanh số. Sử dụng thư viện này để thực hiện các bài kiểm tra thống kê như kiểm định t (t-test) để so sánh doanh số giữa hai nhãn hàng, hoặc tính hệ số tương quan (correlation) để xác định mối liên hệ giữa doanh số và các yếu tố như giá cả hoặc chi phí quảng cáo. Kết quả từ Scipy.stats giúp củng cố cơ sở khoa học cho các phân tích trong báo cáo.
* Statsmodels hỗ trợ xây dựng các mô hình thống kê phức tạp hơn, chẳng hạn như hồi quy tuyến tính để dự đoán doanh số dựa trên các biến đầu vào. Thư viện này cung cấp các chỉ số thống kê chi tiết như p-value và R-squared, giúp đánh giá độ tin cậy của mô hình. Trong báo cáo, Statsmodels giúp tôi trình bày các kết quả dự đoán có cơ sở thống kê vững chắc.
* **Scikit-learn:** Dự đoán bằng machine learning.
* Scikit-learn là công cụ chính để tôi xây dựng các mô hình học máy nhằm dự đoán doanh số nhãn hàng. Ví dụ, có thể sử dụng thuật toán hồi quy (regression) để dự đoán doanh số tương lai dựa trên dữ liệu lịch sử, hoặc thuật toán phân cụm (clustering) để phân nhóm nhãn hàng theo đặc điểm doanh số. Scikit-learn cung cấp quy trình đơn giản để huấn luyện, kiểm tra và đánh giá mô hình, giúp đưa ra dự đoán chính xác hơn và trình bày kết quả trong báo cáo với độ tin cậy cao.

### **II. Phương Pháp Phân Tích**

* **Xử lý dữ liệu**: Đọc dữ liệu, làm sạch, và tính toán các số liệu cơ bản (tổng, trung bình).
* **Trực quan hóa**: Vẽ biểu đồ (histogram, đường, cột) để xem xu hướng và mối quan hệ.
* **Phân tích thống kê**: Kiểm tra tương quan và giả thuyết về dữ liệu.
* **Dự đoán**: Xây dựng mô hình hồi quy để dự báo doanh số.

**III. Xây dựng mô hình dự đoán**

1. **Lựa chọn phương pháp dự đoán**

* Mô hình chính: Kết hợp Mô hình chuỗi thời gian (như Prophet) và Học máy (như Gradient Boosting hoặc Random Forest).
* Lí do:
* Mô hình chuỗi thời gian tận dụng xu hướng và mùa vụ từ dữ liệu lịch sử.
* Học máy xử lý tốt các biến số phức tạp (giá, chiết khấu, giá đối thủ) và cải thiện độ chính xác.
* Phân tích nhân quả bổ sung: Sử dụng Difference-in-Differences (DiD) để đánh giá tác động của các sự kiện (như giảm giá hoặc ngày lễ) lên doanh số, sau đó tích hợp kết quả vào mô hình dự đoán

#### **Tiền xử lý Dữ liệu**

* Làm sạch dữ liệu: Xóa hoặc điền giá trị thiếu (ví dụ: dùng trung bình cho Price, loại bỏ hàng thiếu Sales\_Quantity).
* Chuyển đổi biến:
* Chuyển Date thành các đặc trưng như tháng, năm, hoặc ngày lễ (Holiday\_Indicator).
* Mã hóa biến phân loại (Category, Brand, Day\_of\_Week) bằng one-hot encoding hoặc target encoding.
* Tạo đặc trưng mới: Tính chênh lệch giá (Price - Competitor\_Price), biến trễ (Sales\_Quantity của ngày trước), hoặc các chỉ số mùa vụ.

#### **Xây dựng Mô hình**

* Mô hình chuỗi thời gian:
* Sử dụng Prophet (dễ sử dụng, xử lý tốt xu hướng và mùa vụ) để dự đoán doanh số dựa trên dữ liệu lịch sử.
* Ví dụ: Nhập dữ liệu theo định dạng ngày và doanh số, sau đó thêm các biến ngoại sinh (như giá).
* Mô hình học máy:
* Sử dụng Gradient Boosting (như XGBoost hoặc LightGBM) vì nó xử lý tốt dữ liệu lớn, biến phân loại, và tương tác giữa các biến.
* Các biến đầu vào: Price, Discount, Competitor\_Price,Holiday\_Indicator, Past\_Purchase\_Trends, và các đặc trưng thời gian.
* Biến mục tiêu: Sales\_Quantity.
* Kết hợp: Sử dụng kết quả từ Prophet làm đầu vào cho Gradient Boosting để cải thiện dự đoán.

#### **Đánh giá và Điều chỉnh Mô hình**

* Chia dữ liệu: Dùng 80% dữ liệu cũ làm tập huấn luyện, 20% gần nhất làm tập kiểm tra, đảm bảo thứ tự thời gian.
* Chỉ số đánh giá: Sử dụng Mean Squared Error (MSE), Mean Absolute Error (MAE), hoặc R-squared để đo độ chính xác.
* Điều chỉnh: Thêm đặc trưng (như biến trễ), thử nghiệm các siêu tham số (hyperparameters) để tối ưu hóa.

#### **Tích hợp Phân tích Nhân quả (DiD)**

* Ứng dụng DiD: Nếu có dữ liệu trước/sau một sự kiện, so sánh sự thay đổi doanh số giữa nhóm áp dụng và nhóm không áp dụng để xác định tác động nhân quả.
* Tích hợp: Sử dụng kết quả DiD (tác động định lượng) làm biến đầu vào hoặc điều chỉnh mô hình học máy.