Sales Forecasting and Optimization

Project's Objectives:

- Develop a machine learning model to accurately forecast sales trends, helping businesses optimize inventory and reduce financial losses.
- Build a user-friendly web application that provides real-time sales predictions and visual insights for better decision-making.
- Enable companies to improve resource management, marketing strategies, and operational efficiency through AI-driven forecasting.

Future Plans:

- **Enhancing Model Accuracy:** Incorporate external factors such as weather conditions, economic trends, and competitor data to refine predictions.
- **Automated Model Updates:** Implement an AutoML pipeline to continuously retrain the model based on new data.
- **Expanding Functionality**: Extend the system to include customer behavior analysis and predictive marketing recommendations.
- **Cloud Deployment:** Host the model on a scalable cloud platform (AWS/Heroku) to ensure accessibility and performance.

Tasks:

Abdelrahman Hatem:

- Collect and validate historical sales data.
- Handle missing values, duplicates, and inconsistencies.
- Feature engineering (time-based features, category grouping, etc.).
- Deliverables: Cleaned dataset, EDA report.

Maryam Galal:

- Perform statistical analysis to understand trends and seasonality.
- Generate insights using visualizations (line charts, bar graphs, heatmaps).
- Identify key factors affecting sales.
- Deliverables: EDA Notebook, Data Insights Report.

Mariam Ibrahim:

- Research and implement forecasting models (ARIMA, Prophet, LSTM, etc.).
- Train and evaluate different models using RMSE, MAE, and MAPE.
- Optimize models through hyperparameter tuning.
- Deliverables: Model Training Code, Performance Evaluation Report.

Youssef Morsy:

- Perform back testing to validate model accuracy.
- Ensure generalization by testing on unseen data.
- Compare model predictions against actual sales.
- Deliverables: Model Validation Report, Test Results.

Amr Shereen & Hend Elhout:

- Develop an API (Flask/FastAPI) for model predictions.
- Deploy the model on a cloud service (AWS, GCP, or Heroku).
- Ensure scalability and response efficiency.
- Deliverables: Deployed API, Endpoint Documentation.
- Implement real-time monitoring for model performance.
- Track drift in sales predictions over time.
- Set up alerts for accuracy drops or unusual patterns.
- Deliverables: Monitoring Dashboard, Performance Report.