How to Run the Sales Forecasting ML Application

Quick Start

1. **Navigate to the application directory:**
   ```
   bash
   cd /home/ubuntu/streamlit_sales_app
   ```

2. **Install dependencies (if not already installed):**
   ```
   bash
   pip install -r requirements.txt
   ```

3. **Run the application:**
   ```
   bash
   streamlit run app.py
   ```

4. **Open your browser and go to:**
   `http://localhost:8501`

Using the Application

Step 1: Prepare Your Data

Ensure your CSV file has these columns:
- `Product Name` - Name of the product
- `Total Sales` - Sales amount
- `Date` - Transaction date (YYYY-MM-DD format)
- `Temperature` - Temperature on that day
- `Holiday(0/1)` - Holiday indicator (0=No, 1=Yes)
- `Day` - Day of week (1=Monday, 7=Sunday)

Step 2: Upload Your File

1. Click "Browse files" or drag and drop your CSV file
2. The app will automatically load and display data overview

Step 3: Configure Model Training

1. Adjust "Minimum samples required per product" (default: 5)
2. Set "Maximum products to model" (default: 20)

Step 4: Train Models

1. Click " Train Models" button
2. Wait for the training process to complete
3. Review model performance metrics

Step 5: Generate Visualizations

1. Click " Generate Visualizations" button
2. View the comprehensive dashboard with:
3. Top 10 products by sales
4. Temperature vs sales relationship
5. Holiday vs non-holiday sales comparison
6. Sales by day of week

Step 6: Download Results

1. Click " Download Summary Statistics" to get a CSV with key metrics

Features

Automatic ML Model Training: Trains Random Forest, Gradient Boosting, and Linear Regression models
Smart Feature Engineering: Creates lag features, rolling averages, and cyclical encodings
Interactive Visualizations: Generates charts similar to your reference images
Performance Metrics: Shows MSE, MAE, and R^2 scores for each model
Feature Importance: Displays which factors most influence sales
Export Capabilities: Download summary statistics and results

Sample Data

A sample CSV file (`sample_data.csv`) is included in the application directory for testing purposes.

Troubleshooting

Issue: Application won't start
Solution: Make sure all dependencies are installed with `pip install -r requirements.txt`

Issue: File upload fails
Solution: Ensure your CSV has all required columns with correct names

Issue: No models trained
Solution: Reduce the "Minimum samples required per product" setting

Issue: Visualizations don't appear
Solution: Make sure your data has the required columns and sufficient data points

Technical Notes

- The application uses time series cross-validation to prevent data leakage
- Products with insufficient data are automatically skipped
- The best performing model is automatically selected for each product
- All visualizations are generated using matplotlib and seaborn
- The app is responsive and works on both desktop and mobile devices