# PYTHON-BASED SALES FORECASTING FOR A MULTI-LOCATION RETAIL CHAIN IN THE UNITED STATES

## 1. Background

A U.S.-based retail chain operating across 12 states approached us to forecast monthly sales at both the national and store levels. The client had historical sales data, but their internal teams lacked expertise in forecasting models and automation using Python. Inventory imbalances and reactive staffing were common, especially during seasonal fluctuations.

We were brought in to build time series forecasting models in Python using ARIMA and Facebook Prophet to deliver store-level and aggregate forecasts. The goal was to generate actionable, automated outputs usable by operations, finance, and merchandising teams.

## 2. Objective

- To forecast monthly sales for each store and for the entire retail chain using Python
- To compare forecasting models (ARIMA vs. Prophet) and select the most accurate for production
- To create visual, structured forecasts for 12 months ahead, with clear confidence intervals
- To deliver a Python-based script for monthly updates, along with an executive-facing PDF report

## 3. Data Used

**Source**: Internal POS system (January 2018 – December 2022)

**Dataset**:

- Monthly total sales for 78 stores
- Fields: Store_ID, Month, Sales, Store_Location, Store_Type
- Supplemented with metadata on regional promotions and seasonal events

**Frequency**: Monthly (aggregated sales revenue)

# 4. Methodology

### 4.1 Data Preprocessing

- Checked for missing months and corrected reporting errors
- Deseasonalized trend with STL decomposition
- Conducted stationarity tests (ADF) and differenced non-stationary series
- Normalized sales for chain-level aggregation

### 4.2 Models Used

- **ARIMA (AutoRegressive Integrated Moving Average)**: Model selection via AIC minimization and PACF/ACF plots
- **Prophet (Facebook)**: Used with holiday effects and seasonality controls
- Forecast horizon: 12 months (Jan 2023 – Dec 2023)
- Evaluation Metrics (on last 12 months of actuals):
    - MAE (Mean Absolute Error)
    - RMSE (Root Mean Squared Error)
    - MAPE (Mean Absolute Percentage Error)

# 5. Forecasting Results

| Model | MAE | RMSE | MAPE |
|---|---|---|---|
| ARIMA | $5,200 | $6,850 | 6.1% |
| Prophet | $4,740 | $6,190 | 5.4% |

- Prophet performed slightly better, especially in handling holidays and store-specific promotions
- Both models showed strong performance in high-volume urban stores, with wider variance in rural locations
- Visual inspection showed Prophet captured **seasonal spikes** more clearly

# 6. Insights and Applications

- **Top 10 stores** showed clear **Q4 sales peaks**, now forecasted 3 months in advance for 2023

- Identified **underperforming stores** with flat or declining trend lines requiring targeted marketing
- Quantified **stockout risk windows** by comparing forecasts to existing safety stock thresholds
- Supported **labor scheduling adjustments** based on expected monthly volume forecasts

## 7. Reporting Output

- **Python Forecasting Script (.ipynb)**:
  - ARIMA and Prophet models wrapped in reusable functions
  - Configurable forecast horizon and store filtering
  - Integrated monthly output to .csv and .xlsx reports
- **PDF Forecast Report (16 pages)**:
  - Overview of methodology
  - Store-wise 12-month forecast charts with 80% and 95% intervals
  - Summary table of projected sales and YOY growth by region
  - Recommendations for Q2 inventory planning and resource reallocation
- **Excel Output**:
  - Tabs by store with forecast values, error margins, and color-coded alerts
  - Aggregated national forecast and category breakdown

## 8. Strategic Impact

- Enabled **data-driven inventory orders** ahead of Q1 and Q4 sales peaks
- Client reduced overstock by 9% in the first post-project quarter
- Staffing forecasts shared with HR team to optimize labor during high/low sales months
- Python script integrated into a monthly automated reporting process used by all regional managers