

Sales Forecasting using Machine Learning

Overstocking ties up capital and understocking loses sales — inventory mistakes cost retailers billions every year.

01 Title

Sales Forecasting using Machine Learning

02 Introduction / Background

Overstocking ties up capital and understocking loses sales — inventory mistakes cost retailers billions every year. Demand forecasting sits at the center of that decision, and it is an ideal final-year topic: real data, honest error metrics, a clear model comparison story (classical ARIMA vs modern XGBoost) and recommendations a business would actually use.

03 Problem Statement

Naive forecasts (last-period carry-forward, simple averages) ignore seasonality, promotions and holidays, while single-model demos skip the comparison examiners ask for. This project engineers time-series features, trains XGBoost against ARIMA and Prophet baselines with walk-forward validation, and converts forecasts into inventory recommendations — reorder points, safety stock and days of cover.

04 Objectives

- Time-series feature engineering: lags, rolling stats, calendar, holidays, promos.
- XGBoost demand model with ARIMA and Prophet baselines.
- Walk-forward validation with no future-data leakage.
- MAPE/RMSE model comparison, per model and per SKU.
- Seasonality and promotion-lift analysis.
- Inventory recommendations and a what-if promo simulator.

05 Proposed System / Working

Implemented in Python 3.9+ with Pandas, XGBoost, Prophet/statsmodels and Streamlit. Sales CSVs are cleaned and enriched with lag features (7/14/28 days), rolling means, day-of-week/month indicators, holiday flags and promo/price fields. An expanding-window walk-forward split trains on the past and tests on genuinely future windows. ARIMA, Prophet and XGBoost are trained on identical splits; MAPE and RMSE select the production model per SKU. Forecasts carry confidence

bands, and inventory logic derives reorder points and safety stock from forecast mean and error. The SaleSight Streamlit dashboard visualizes forecast-vs-actual charts, model-comparison bars, seasonality heatmaps and a what-if simulator, with CSV export.

XGBoost typically achieves 8–15% MAPE on the included retail-style data, clearly beating the ARIMA and Prophet baselines on promo-heavy SKUs — documented per model and per SKU with walk-forward honesty. The live SaleSight demo uploads the sample CSV, trains, and renders forecast charts, comparison bars and inventory recommendations in real time.

06 Technologies / Components Used

Python · scikit-learn · XGBoost · Prophet/ARIMA · Streamlit dashboard

07 Key Features

- XGBoost vs ARIMA vs Prophet comparison on identical splits.
- 7/30/90-day per-SKU forecasts with confidence intervals.
- Seasonality heatmaps and promotion-lift charts.
- Inventory outputs: reorder point, safety stock, days of cover.
- What-if simulator for promo intensity and price.
- SaleSight Streamlit dashboard with CSV upload and export.

08 Applications / Use Cases

- Retail demand planning
- e-commerce inventory
- FMCG distribution
- restaurant/hospitality ordering
- any SKU-based business with sales history

10 Conclusion

SaleSight is a complete, examinable forecasting project — from feature engineering to inventory decisions — with the walk-forward discipline and model comparison that make a demand-forecasting viva strong.