Time Series Forecasting

Overview:

This project demonstrates an end-to-end time series forecasting workflow using Python's `StatsForecast` and `UtilsForecast` libraries. It predicts daily bakery item sales (like Baguette and Croissant) by applying models such as Naive, Historic Average, Window Average, Seasonal Naive, ARIMA, and SARIMA. The process includes data preparation, visualization, model training, evaluation, and model comparison using performance metrics like MAE and RMSE.

Key Terms and Concepts:

Time Series: A sequence of data points recorded at successive time intervals (daily, weekly, etc.). In this project, it represents bakery item sales over time.

Forecasting: Predicting future values based on historical data patterns. Here, the goal is to predict future bakery sales.

Naive Forecast: A simple method assuming the next value is the same as the last observed value. Used as a performance baseline.

Historic Average: A model that predicts future values as the average of all past observations.

Window Average: Takes the average of a fixed window of previous values (e.g., last 7 days) to forecast the next period

Seasonal Naive: Assumes that data repeats every season. For example, sales on Monday are expected to be similar to the previous Monday.

ARIMA (AutoRegressive Integrated Moving Average): A statistical model that combines autoregression (using past values), differencing (to remove trends), and moving average (to account for past forecast errors). It is used here to model bakery sales trends and patterns.

SARIMA (Seasonal ARIMA): An extension of ARIMA that includes seasonal components to capture weekly or monthly patterns. Used here to model the repeating weekly behavior in sales.

Exogenous Variables (Exog): External factors that influence the target variable but are not part of the time series itself, such as `unit_price`. SARIMA_exog uses these to improve predictions.

Cross-Validation: A method to test model stability by training and testing on multiple rolling time windows instead of one static train/test split.

MAE (Mean Absolute Error): A metric that measures the average absolute difference between predicted and actual values. Lower MAE means better accuracy.

RMSE (Root Mean Squared Error): A metric that emphasizes larger errors by squaring them before averaging, making it sensitive to big deviations.

MAPE (Mean Absolute Percentage Error): Measures accuracy as a percentage, showing how far predictions are from actual values in relative terms.

SMAPE (Symmetric MAPE): A variation of MAPE that normalizes the difference, making it less sensitive to small values.

CRPS (Continuous Ranked Probability Score): Used in probabilistic forecasting to measure how well predicted distributions match observed values.

Fourier Features: Mathematical transformations used to represent seasonal patterns in time series data. These are especially useful in models like SARIMA with time-based patterns.

Step-by-Step Implementation:

• 1. Setup and Installation: Install required packages using `pip install statsforecast utilsforecast`.

• 2. Data Import: Load the bakery sales dataset (`daily_sales_french_bakery.csv`). Filter products with enough historical data.

• 3. Data Cleaning: Remove irrelevant columns such as `unit_price` initially for basic modeling.

• 4. Visualization: Use `plot_series` to explore trends in daily sales of items like Baguette and Croissant.

• 5. Model Selection: Initialize models such as Naive, Historic Average, Window Average, and Seasonal Naive with a 7-day forecast horizon.

• 6. Model Training: Fit the models using `StatsForecast` and generate predictions for future days.

• 7. Evaluation: Compare models using the MAE metric to identify which performs best.

• 8. Advanced Modeling: Implement ARIMA and SARIMA models to capture trends and seasonality.

• 9. Cross-Validation: Apply rolling window validation to ensure model robustness over time.

• 10. Exogenous Variables: Include features like `unit_price` (SARIMA_exog) to improve forecasts.

• 11. Feature Engineering: Add Fourier and time-based features (day, week, month) to better capture seasonal trends.

• 12. Probabilistic Forecasting: Generate prediction intervals with 80% confidence to quantify forecast uncertainty.

• 13. Final Comparison: Evaluate all models using metrics like MAE, RMSE, MAPE, SMAPE, MASE, and CRPS. SARIMA_exog performed best in accuracy.

Conclusion:

This project walks through a complete time series forecasting pipeline, from simple statistical methods to advanced ARIMA-based models with external factors. By including feature engineering and cross-validation, it provides a reliable way to forecast bakery sales and evaluate model performance. The SARIMA model with exogenous variables achieved the highest accuracy, demonstrating the power of incorporating contextual features into forecasting models.