Time Series Forecasting: Model Selection and Evaluation

Jan Robert Eljasiak 909837

March 4, 2025

Introduction

This report presents a comprehensive analysis of forecasting the time series variable from the dataset *ts2024.csv*. The dataset contains hourly observations from January 1, 2015, to November 30, 2016, with a total of 16,800 data points. The objective is to predict the missing values for December 2016 (744 time points) using three distinct models: Seasonal ARIMA (SARIMA), Unobserved Components Model (UCM), and XGBoost. The report details the data preprocessing steps, exploratory analysis, model selection, hyperparameter tuning, evaluation metrics, and final forecasting results.

Data Preprocessing and Exploration

Data Loading and Transformation

The dataset was loaded, and the *DateTime* column was set as the index. Additional temporal features such as Year, Month, Day, Hour, Day of the Week, and Quarter were derived to assist with model performance.

Exploratory Data Analysis

A visual examination of the time series showed a strong seasonal pattern, which was confirmed through seasonal decomposition (using an additive model with a period of 24 hours). This decomposition identified a clear seasonal component, a trend, and residuals.

The autocorrelation function (ACF) plot revealed strong periodic autocorrelations, with significant spikes at intervals of 24 lags, reinforcing the presence of daily seasonality.

Train-Test Split Justification

To evaluate the accuracy of forecasting models, we split the dataset, which contains 16,800 known values of the "X" variable, into training and testing sets. The first 16,080 observations, covering the period from January 1, 2015, at 0:00 to October 31, 2016, at 23:00 were used for training. The last 720 observations, corresponding to the period from November 1, 2016, at 00:00 to November 30, 2016, at 23:00, were reserved for testing. This approach was chosen because the primary forecasting task is to predict values for December 2016. By using November 2016 as the test set, we can assess how well the models perform on recent data before making final predictions for December.

Model 1: SARIMA

The first model selected for forecasting was SARIMA (Seasonal ARIMA). The optimal parameters were determined using the auto-ARIMA method, which selects the best configuration based on AIC/BIC criteria. The SARIMA model was then fitted to the training data and used to generate forecasts. However, it is important to note that extracted features, such as additional temporal variables, were not included in the training process, as their inclusion led to reduced model accuracy.

The chosen SARIMA parameters were $p = 1, d = 0, q = 2$ for the ARIMA part, and $P = 2, D = 0, Q = 1, s = 24$ for the seasonal component, which corresponds to daily seasonality with hourly data.

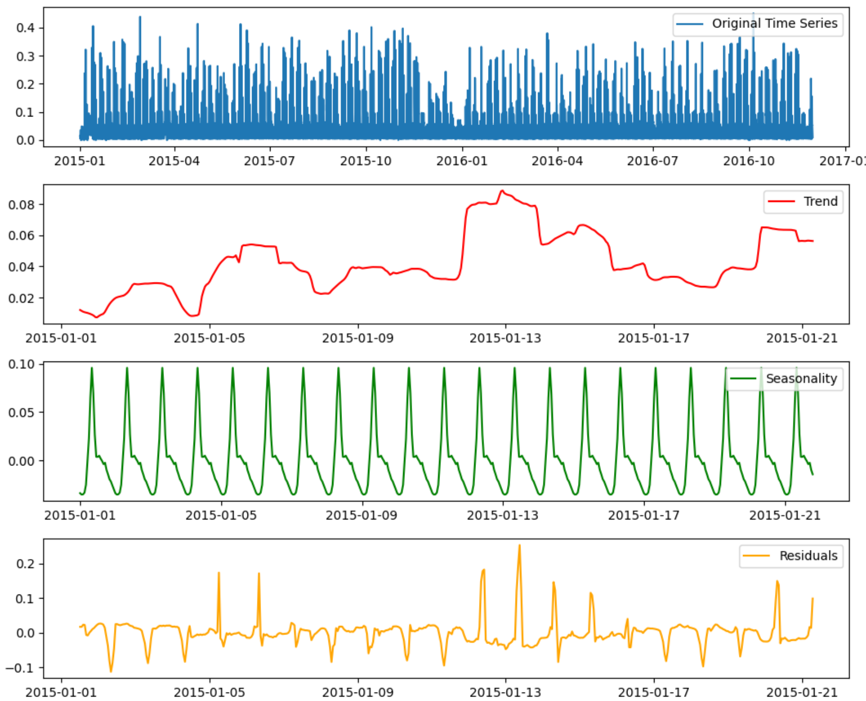

Figure 1: Seasonal Decomposition of the Time Series

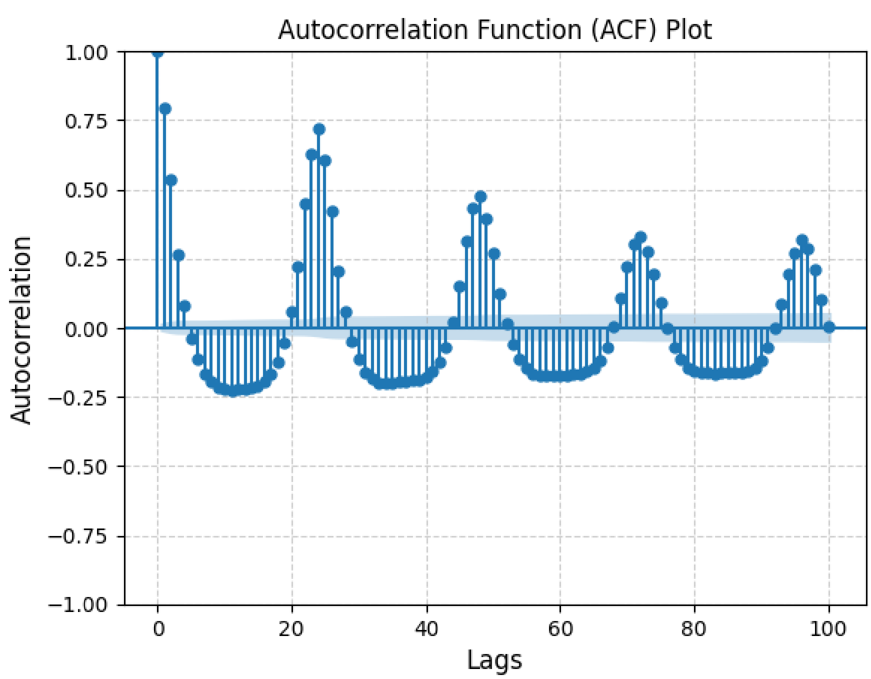

Figure 2: Autocorrelation Function (ACF) Plot

The performance of the SARIMA model was evaluated using Mean Absolute Error (MAE) and Mean Squared Error (MSE), and the results showed a reasonable fit to the data.

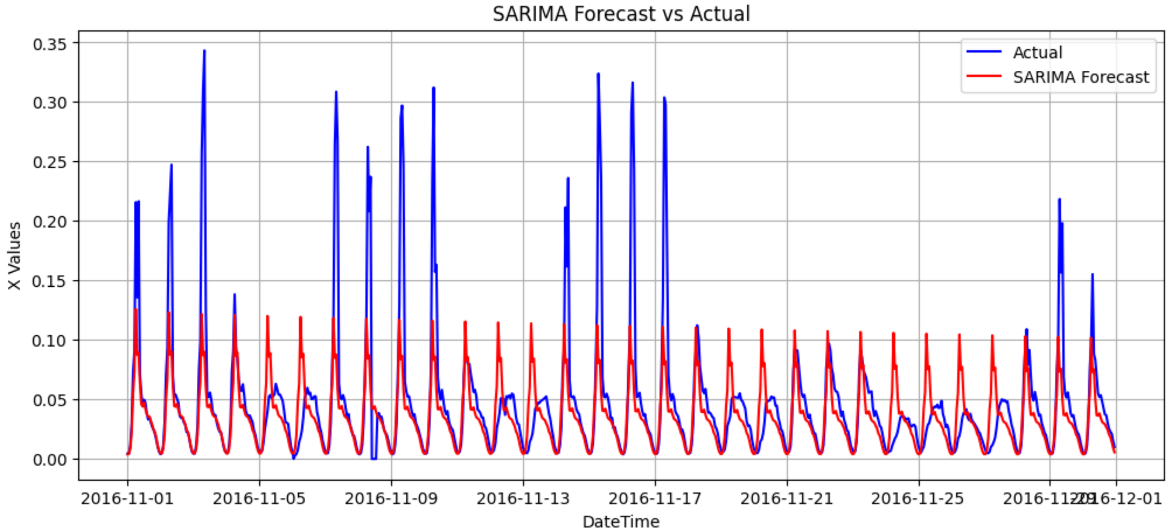

Figure 3: SARIMA Forecast vs Actual

Model 2: Unobserved Components Model (UCM)

The second model selected was the Unobserved Components Model (UCM). This model was chosen due to its ability to capture complex seasonal and trend components, making it suitable for time series data with such characteristics. The UCM was fitted to the training data with exogenous features such as Hour, Year, Month, Day, DayOfWeek, and Quarter. The model was then used to generate forecasts for the test period.

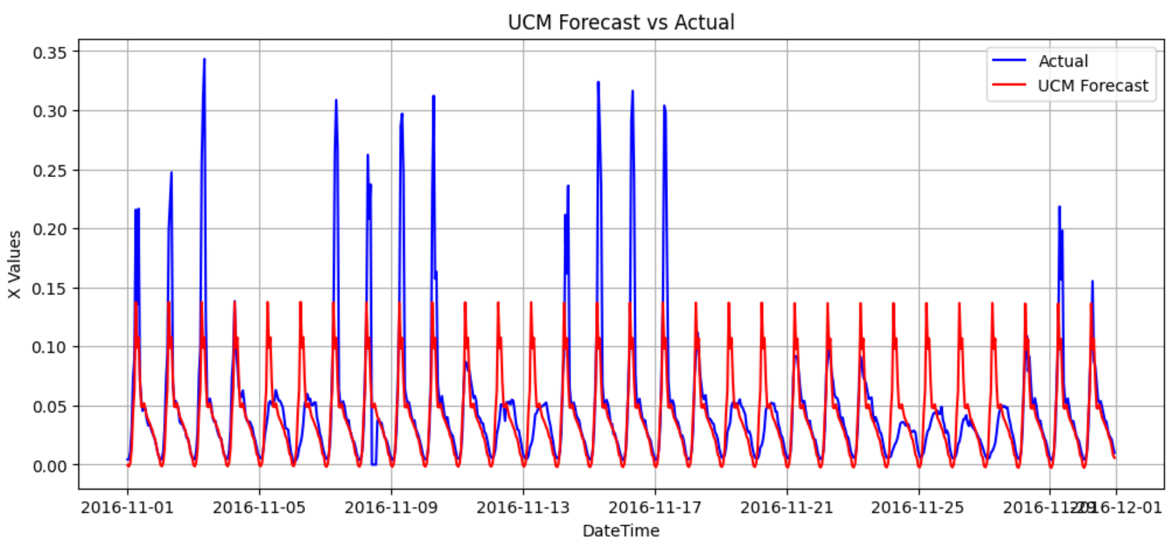

Figure 4: UCM Forecast vs Actual

Similar to SARIMA, the performance of the UCM was evaluated using MAE and MSE, and the model showed good predictive performance.

Model 3: XGBoost

The third model selected for forecasting was XGBoost, a gradient boosting algorithm known for its flexibility and ability to handle non-linear relationships. XGBoost was chosen after evaluating other machine learning models, including LSTM and Prophet, where XGBoost demonstrated superior performance.

A hyperparameter search using RandomizedSearchCV was performed to tune the model's parameters, such as the number of estimators, learning rate, and tree depth.Best hyperparameters selected for this modele were:

- n_estimators = 500
- max_depth = 7
- learning_rate = 0.2

XGBoost was trained using the features from the training set and then used to generate forecasts for the test period.

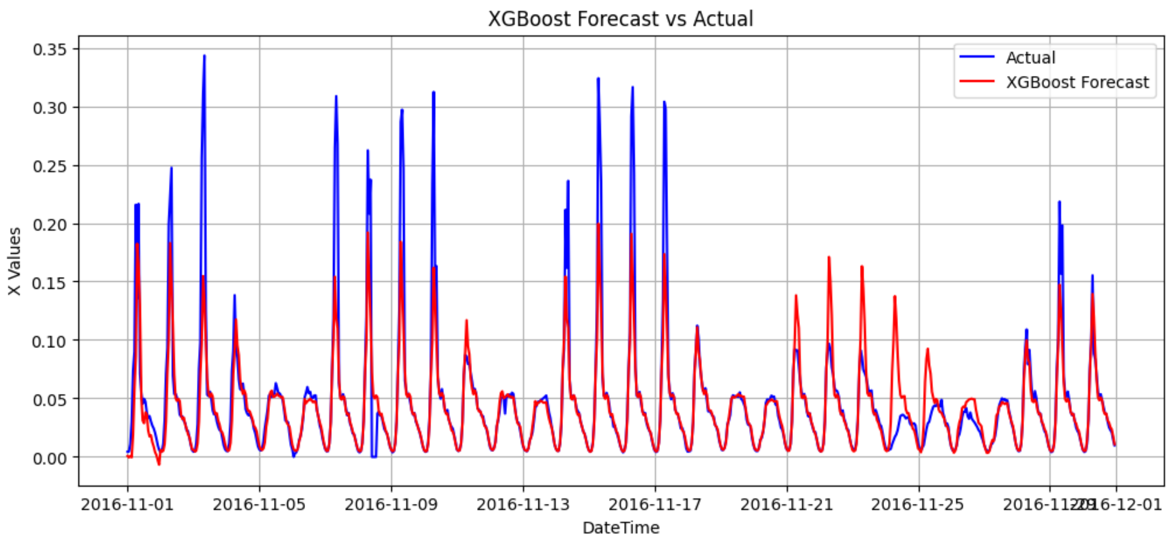

Figure 5: XGBoost Forecast vs Actual

Evaluation of Models

The three models (SARIMA, UCM, and XGBoost) were evaluated using the following metrics:

- **Mean Absolute Error (MAE)**: Measures the average absolute errors between forecasted and actual values.
- **Mean Squared Error (MSE)**: Measures the average squared difference between forecasted and actual values.

The results of the evaluation are as follows:

Model	MAE	MSE
SARIMA	0.0177	0.0015
UCM	0.0181	0.0015
XGBoost	0.0101	0.0009

Table 1: Model Evaluation Results

From the results, we can conclude that XGBoost performed the best in terms of both MAE and MSE.

Forecasting Future Values for December 2016

Using the three selected models, which were validated on November 2016 data, we forecasted the unknown values of variable X for December 2016. The same model parameters used during training were applied to a dataset of 16,080 data points—comparable in size to the full dataset of 16,800 values.

The final forecast results have been saved in the file *909837_20250216.csv*.

Conclusion and Future Work

This report presented the forecasting of time series data using SARIMA, UCM, and XGBoost. The evaluation results indicated that XGBoost outperformed the other models in terms of predictive accuracy, capturing the underlying patterns in the data more effectively.

Although computational limitations restricted certain enhancements, the models demonstrated reliable forecasting performance, and predictions for December 2016 were successfully generated.

Future improvements could address some of the constraints by incorporating:

- Further hyperparameter tuning for SARIMA and UCM to enhance model accuracy.
- Integration of external factors such as holidays or weather data to improve predictive performance.
- Implementation of time series cross-validation techniques (e.g., rolling window validation) for more robust model evaluation.

While computational power was a limiting factor, the applied models provided meaningful insights, and further refinements could lead to even more accurate forecasts.