Time Series Regression Assignment

Ryoji Takahashi

December 5, 2024

1 Description of the Assignment and Objectives

The objectives of this time series regression assignment are:

1. To choose an open multivariate time series dataset (not financial) where the target variable does not exhibit seasonality (reference source).
2. To confirm the dataset's non-seasonality.
3. To select, train, and test machine learning/time series models that can predict 100 future time periods, assuming no regressor variable values are available beyond the prediction point.
4. To graphically present the predictions and evaluate the model performance using relevant KPIs.
5. To draw conclusions from the conducted analysis.
6. To deliver all necessary files for reproducing the analysis and deploying the trained models (DevOps integration).

1.1 Dataset Selection and Description

The Population Time Series Data was selected for this study. The dataset contains 816 rows and 3 columns of population records from 01/01/1952 to 01/12/2019. The target variable is the "value," representing the population size over time.

Due to the dataset's small size, I resampled it to daily intervals and interpolated the missing values. This preprocessing increased the sample size to 24,807, providing sufficient data for building prediction models.

1.2 Non-Seasonality Confirmation

To confirm non-seasonality, the following steps were undertaken:

- Seasonal decomposition: A decomposition plot was analyzed to ensure the seasonal component is flat or negligible.
- Augmented Dickey-Fuller (ADF) test: This test was applied to check for stationarity. A p-value > 0.05 indicates that the series is non-stationary, but this result does not confirm or refute the presence of seasonality.

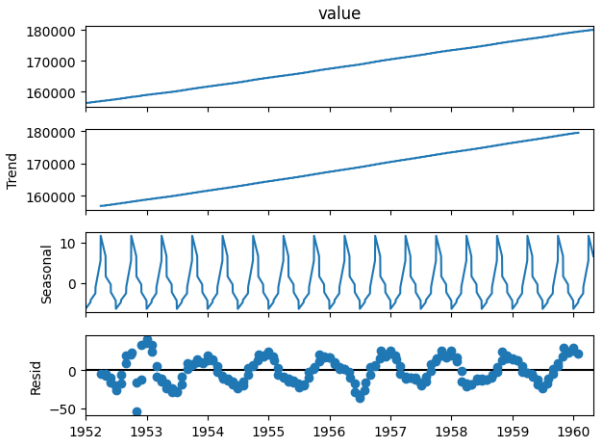

Figure 1: Seasonal Decomposition

Figure 1 shows the decomposition plot. The seasonal component has weak seasonal feature. However, this may be artifacts. Therefore, I further looked at ACF (Autocorrelation Function), PACF (Partial Autocorrelation Function). For seasonal data, ACF sees significant spikes at regular intervals (lags) corresponding to the seasonal period, and PACF may show significant spikes at seasonal lags. For non-seasonal data, ACF will decay rapidly, indicating no seasonality, and spikes are typically confined to the first few lags.

Figure 2, ACF plot shows a very slow decay, indicating non-stationarity.

Figure 3, PACF indicates significant lags at very early points, which is typical for stationary data. However, the lack of a periodic or repeating pattern in PACF also leans toward non-seasonality.

Additionally, the ADF test results (ADF Statistic $= -1.441$, P-value $= 0.562$) confirm that the dataset is non-stationary and non-seasonal.

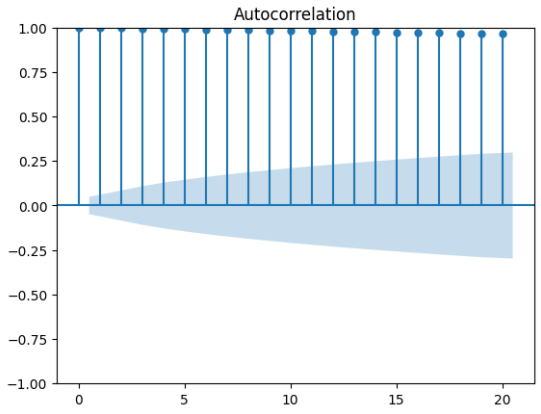

Figure 2: ACF (Autocorrelation Function) plot

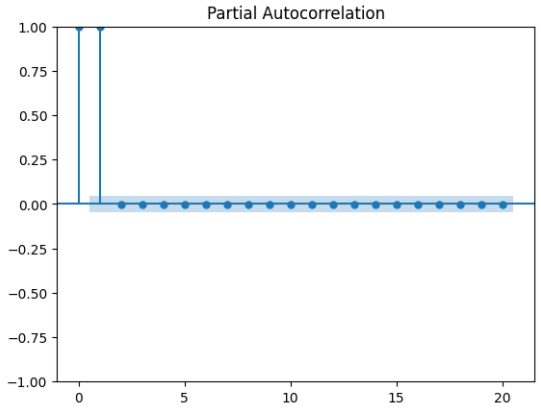

Figure 3: PACF (Partial Autocorrelation Function)

1.3 Trained Models and Methodology

The dataset was split into training (80%) and testing (20%) subsets. The following models were trained and evaluated:

- **SARIMAX (Seasonal ARIMA with Exogenous Variables):** A time series forecasting technique extending ARIMA to include seasonal effects and external regressors. It is well-suited for smaller datasets.
- **XGBoost with Optuna:** A gradient boosting framework optimized using the Optuna library for hyperparameter tuning. XGBoost is effective for structured data and handles missing values efficiently. While XGBoost has shown strong results in seasonal datasets, it struggled in this non-seasonal dataset. This reason could be that XGBoost is not designed to capture long-term dependencies, which are often present in non-seasonal time series data.
- **Long Short-Term Memory (LSTM):** A recurrent neural network architecture tailored for sequence predictions. Its ability to learn long-term dependencies makes it powerful for time series forecasting.

Figure 4 illustrates the predictions for 100 future time periods (6 days each).

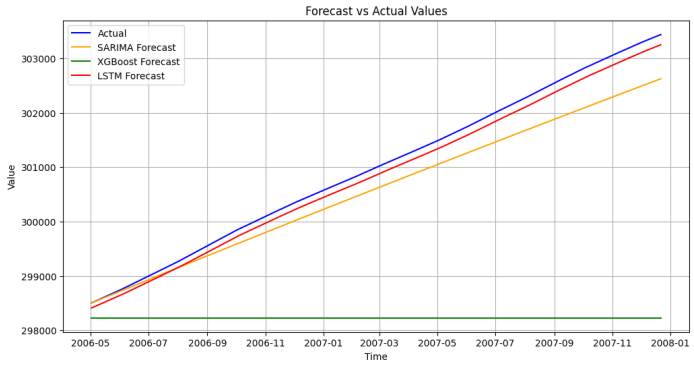

Figure 4: Prediction Results for 100 Future Periods (6 Days Each)

Model	MSE(NMSE)	MAE(NMAS)	R^2
SARIMAX	218267.59 (0.107)	404.41 (0.001)	0.892
XGBoost	9682372.40(4.797)	2766.88 (0.0092)	-3.777
LSTM	20921.49 (0.005)	141.99 (0.0003)	0.990

Table 1: KPIs (Metrics) Performance of Trained Models

Table 1 shows KPIs (metrics), Mean Squared Error (MSE), Mean Absolute Error (MAE), and R^2, NMSE, NMAS stand for normalized MSE and MAS, respectively. As shown in Table 1, LSTM demonstrated higher accuracy compared to SARIMAX and XGBoost. SARIMAX displayed consistent performance, while I am seeing the XGBoost limitations for non-seasonal datasets. Notice that the low NMAE of XGBoost indicates that errors are small relative to the mean of the target variable. However, NMSE and R2 show the model's predictions deviate significantly from the true values as in Figure 4.

2 Conclusion and Future Work

The analysis highlighted that SARIMAX and LSTM are better suited for non-seasonal datasets. SARIMAX is computationally efficient and LSTM excels in accuracy for larger datasets. Tree-based models may fail to capture long-term dependencies, making them less effective for non-seasonal time series forecasting.

Future improvements could involve:

- Addressing overfitting in LSTM models by incorporating dropout layers and early stopping.
- Exploring real-time forecasting methods and generative AI for time series analysis.

Time series forecasting remains an evolving field, with exciting opportunities in automated model selection and real-time prediction for large-scale applications.