Time Series Analysis Using ARIMA Model from Scratch

Lauvindra Raj

September 2021

I. Introduction (What is Time Series?)

- Sequence of observations recorded at regular time intervals.
- Main reason to analyse a time series:
  - Preparatory step before we develop a forecast of a time series.

II. Steps to conduct a Time Series Analysis

1. Importing time series in Python

   - read_csv()
   - We can import it as a pandas series, with **date as index.**

2. Visualizing a time series

   - Use matplotlib to visualize the series.
   - We can plot season plot on a time series.
   - Box plot of Month-Wise (Seasonal) & Year-Wise (Trend) distribution

3. Patterns in Time Series

- Base Level + Trend + Seasonality + Error
- Cyclic behaviour may occur
- Different between ‘Cyclic’ & ‘Seasonal’ pattern.
    - If pattern is not fixed calendar-based frequencies, then it is cyclic
    - Cyclic depends on business & other socio-economic factors
    - Seasonality will be along with Trend

4. Addictive & Multiplicative Time Series

- Additive Time Series:
    - Value = Base Level + Trend + Seasonality + Error

- Multiplicative Time Series:
    - Value = Base Level + Trend + Seasonality + Error

5. Time Series Decomposition

- Statsmodels library provide an implementation with a decomposition method called **seasonal_decompose()**
- **seasonal_decompose()** function returns a results object.
- The result object contains arrays 4 pieces of data decomposition
    - Trend
    - Seasonal
    - Residual
    - Observed

6. Stationary & Non-stationary Time Series

- Stationary is a property of a time series
- Stationary series is where the values of the series is not a function of time
- Stationary properties of the Series:
    - Mean, Variance & Autocorrelation are constant over time
- Autocorrelation of the series is the correlation of the Series with its previous values.
- Technically stationary properties will have a constant graph whereas non-stationary will have an uptrend or downtrend.

7. Make Non-stationary to stationary

- Forecasting a stationary series is easier, and the forecast itself as products.
- **Autoregressive Forecasting models** are essentially **Linear Regression models** that utilize the lag(s) of the series itself as products
- **Linear Regression** works best if the prediction(X-variables) is not correlated against each other.
- Stationarizing the Series solves this problem since it will remove as persistent autocorrelation

8. How to make a time series stationary

- Differencing the Series
- Take a log of the series
- Take the nth root of the Series
- Combination of above

9. Making the Time Series Stationary

- There are 2 Factors:
  - **Trend**
  - **Seasonality** (Variation at specific time frames)
- Model the trend & seasonality in this Series.
- Remove it & make the Series Stationary.
- Apply statistical forecasting to the stationary series.
- Convert the forecasted values into original by applying the trend& seasonality constrains back to those that we previously separated.

Trend

- Make larger values than smaller, using log, sqrt, cube root etc.
- 2 Methods to model trends & remove them from series
  - Smoothing: Rolling / Moving Average
  - Aggression; Taking the mean for a certain time period(year/month)

- **Smoothing**:
  - We usually take the past few instances (rolling estimates)
  - 2 methods under smoothing:
    - Moving Average
    - Exponentially Weight moving average
  - Moving Average
    - Take X consecutive values e.g., frequency 1 year, we take 12 as the values.

Forecasting a Time Series

- After Time Series stationary, making models on time series is easy because it is easy to add the error, trend & seasonality back into predicted values.
- Method called **ARIMA** (Auto Regressive Integrated Moving Average)
- Similar to Linear Regression e.g., depends on parameter
  - p (AR) Auto Regressive'
  - q (MA) Moving Average
  - d (Amount of differences /Number of Non-Seasonal differences).
- RSS value (Residual sum of squares)
- Arima works best with a single dependant variable
- Linear regression works best with more than one dependant variable.

III. References

- https://www.machinelearningplus.com/time-series/time-series-analysis-python/
- https://machinelearningmastery.com/decompose-time-series-data-trend-seasonality/
- https://medium.com/@stallonejacob/time-series-forecast-a-basic-introduction-using-python-414fcb963000
- https://towardsdatascience.com/time-series-analysis-resampling-shifting-and-rolling-f5664ddef77e
- https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/