Advanced topics in Time Series Forecasting

Zeham Management Technologies BootCamp by SDAIA
August 21th, 2024

SDAIA
الهيئة السعودية للبيانات والذكاء الاصطناعي
Saudi Data & AI Authority

Advanced topics in Time Series Forecasting

Let’s start together…

Agenda

- Exponential Smoothing
- SARIM
- XGBoost

Exponential Smoothing

Exponential Smoothing Forecasting

What is Exponential Smoothing?

Time series methods follow the assumption that a forecast is a linear sum of all past observations or delays. Exponential smoothing gives more weight to the most recent observations and reduces exponentially as the distance from the observations rises, with the premise that the future will be similar to the recent past. The word “exponential smoothing” refers to the fact that each demand observation is assigned an exponentially diminishing weight.

Exponential Smoothing Forecasting

What is Exponential Smoothing?

1. This technique captures the general pattern and can be expanded to include trends and seasonal variations, allowing for precise time series forecasts using past data.
2. This method gives a bit of erroneous long-term forecasts.
3. It works well with the technique of smoothing when the parameters of the time series change gradually over time.

Exponential Smoothing.

What is Exponential Smoothing?

Exponential Smoothing is a time series forecasting method for univariate data that can be extended to support data with a systematic trend or seasonal component. The method assigns exponentially decreasing weights to past observations, which allows the model to give more importance to recent observations while still considering older data.

Exponential Smoothing

Types of Exponential Smoothing

1 - Simple Exponential Smoothing (SES):

Simple smoothing is a method of forecasting time series using univariate data without a trend or seasonality. One must have a single parameter, which is also referred to as alpha (α) or smoothing factor so as to check how much the impact of past observations should be minimized.the weight to be given to the current data as well as the mean estimate of the past depends on the smoothing parameter (α). A smaller value of a implies more weight on past prediction and vice-versa. The range of this parameter is typically 0 to 1.

Exponential Smoothing

Types of Exponential Smoothing

1 - Simple Exponential Smoothing (SES):

- Used for time series data without trend or seasonality.
- The forecast is a weighted average of past observations, with the weights decreasing exponentially.

Formula:

$$\hat{y}_t = \alpha y_{t-1} + (1 - \alpha)\hat{y}_{t-1}$$

where $\hat{y}_t$ is the forecast at time t, y_{t-1} is the actual value at time t−1 , and α (0 < \alpha < 1) is the smoothing parameter.

Exponential Smoothing

Types of Exponential Smoothing

2 - Double Exponential Smoothing :

- Extends simple exponential smoothing to capture linear trends in the data.
- Uses two equations: one for the level and one for the trend.

Formula:

$$l_t = \alpha y_t + (1-\alpha)(l_{t-1} + b_{t-1})$$
$$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1}$$
$$\hat{y}_{t+k} = l_t + kb_t$$

where l_t is the level at time t, b_t is the trend, α(alpha) is the smoothing parameter for the level, and β(beta) is the smoothing parameter for the trend.

Exponential Smoothing

Types of Exponential Smoothing

2 - Double Exponential Smoothing :

Double exponential smoothing, also known as the **Holt’s trend model**, or second-order smoothing, or Holt’s Linear Smoothing is a smoothing method used to predict the trend of a time series when the data does not have a linear trend but does not have a seasonal pattern. The fundamental idea behind double exponential smoothing is to use a term that can take into account the possibility that the series will show a trend.

Double exponential smoothing requires more than just an alpha parameter. It also requires a beta (b) factor to control the decay of the effect of change in the trend. The smoothing method supports both additive and multiplicative trends.

Exponential Smoothing

Types of Exponential Smoothing

3 - Holt-Winters' exponential smoothing:

Triple exponential smoothing (also known as Holt-Winters smoothing) is a smoothing method used to predict time series data with both a trend and seasonal component.

This is the most advanced variation of smoothing. It is used for forecasting time series when the data contains linear trends and seasonality.

Exponential Smoothing

Types of Exponential Smoothing

3 - Holt-Winters' exponential smoothing:

The technique uses exponential smoothing applied three times:

Level smoothing

Trend smoothing

Seasonal smoothing

New smoothing parameter, gamma (γ), is used to control the effect of seasonal component.

Exponential smoothing can be divided into two categories, depending on the seasonality. The Holt-Winter's Additive Method (HWIM) is used for addictive seasonality. The Holts-Winters Multiplicative method (MWM) is used for multiplicative seasonality.

Exponential Smoothing

Types of Exponential Smoothing

3 - Holt-Winters' exponential smoothing:

- Extends Holt's linear trend model to capture seasonality.
- Suitable for time series with both trend and seasonal components.

There are two variations: additive and multiplicative seasonality.

Additive Model Formulas:

$$
\begin{aligned}
l_t &= \alpha(y_t - s_{t-m}) + (1-\alpha)(l_{t-1} + b_{t-1}) \\
b_t &= \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1} \\
s_t &= \gamma(y_t - l_t) + (1-\gamma)s_{t-m} \\
\hat{y}_{t+k} &= l_t + kb_t + s_{t-m+k}
\end{aligned}
$$

Where s_t is the seasonal component, m is the season length, γ is the smoothing parameter for seasonality, and k is the forecast horizon.

Exponential Smoothing

When to use ES?

Exponential smoothing is particularly effective for time series data with a consistent trend, seasonality, and random fluctuations.

It is especially valuable for short to medium-term forecasting of business metrics like sales, revenue, and customer traffic. Additionally, it is beneficial for monitoring and predicting seasonal variations in industries such as tourism, agriculture, and energy.

Exponential smoothing in Python

Python has several exponential smoothing libraries, such as Pandas, Statsmodels, Prophet, etc. These libraries offer different functions and methods to implement different types of smoothing methods.

Let’s practice

Tutorial :

5- Time Series Forecasting/4- Advanced topics in Time Series Forecasting

/LAB/Exponential smoothing_Tutorial.ipynb

Datasets :

5- Time Series Forecasting/4- Advanced topics in Time Series Forecasting

/LAB/Datasets/airline-passengers.csv

Let’s practice

Exercise:

5- Time Series Forecasting/4- Advanced topics in Time Series Forecasting

/LAB/Exponential smoothing_Exercise.ipynb

Datasets :

5- Time Series Forecasting/4- Advanced topics in Time Series Forecasting

/LAB/Datasets/traffic.csv

When to use Exponential Smoothing

The selection of an exponential smoothing method is dependent on the properties of the time series and the forecasting needs.

Simple Exponential Smoothing (SES):

SES best suits time series data with no trend and no seasonality. It is basic, which can be applied when there is no overall systematics in trends or anomalies and straightforward forecasting based on the last observation and the preceding forecast. Because SES is based on computing and is simple to set up, it's ideal for forecasting in real time or where there's a lack of data.

When to use Exponential Smoothing

Holt's Linear Smoothing:

Holt's Linear Smoothing is used for time series data with a trend. A trend is a systematic change in a time series value over time. It's an extension of Simple Exponential Smoothing that includes a trend component along with the level component. This allows for trend patterns to be captured in the data.

Holt's Linear Smoothing is used when the data has a consistent upward or down trend. A forecast that takes into account both current level and trend is required.

It is also used when the data does not have seasonality but has a trend.

When to use Exponential Smoothing

Holt-Winter's Seasonal Smoothing:

Holt-Winter's seasonal smoothing is used in cases of trend and seasonality time series data. It is an extension of the DES, adding a seasonal component to the level and trend components such that seasonal patterns in data could be captured.

Holt-Winter's Seasonal Smoothing method can be applied where the data exhibits both the trend and the repeating pattern over time, like monthly seasonality or quarterly seasonality. In this regard, Holt-Winter's Seasonal Smoothing is able to produce forecasts that can keep up with the current level, trend, and seasonality, thus being very suitable for forecasting situations where the trend and seasonality coexist.

When to use Exponential Smoothing

The choice of an exponential smoothing method may also depend on your specific needs in the forecasting: desired horizon of forecast, level of accuracy, availability of historic data, etc. You would then have to test a number of different methods and adjust smoothing parameters—like alpha and beta—since it is necessary to find the best approach for your particular time series data and forecasting goals.

It is always a good practice to test the accuracy of your forecast with performance metrics and validate the forecast using out-of-sample data before finalizing the decisions.

Benefits of Exponential Smoothing

Analysts can modify the rate at which older observations become less significant in the computations by varying the values of these parameters. As a result, analysts can adjust the weighting of recent observations in relation to previous observations to suit the needs of their field.

On the other hand, the moving average approach assigns 0 weight to observations outside of the moving average window and assigns equal weight to all historical observations when they occur within its frame. Because exponential smoothing models error, trend, and seasonality in time series data, statisticians refer to it as an ETS model, just like they do with the Box-Jenkins ARIMA methodology.

Limitations of Exponential Smoothing

However, there are some drawbacks to exponential smoothing.

For example, it may not work well for time series with complex patterns or anomalies, such as sudden level or trend changes, outliers or sudden seasonality.

In these cases, other sophisticated forecasting techniques may be more suitable.

Also, the selection of smoothing parameters, such as alpha, beta and γ, can affect the precision of the forecasts. Finding the best values for these parameters might require some trial and error or model selection techniques.

SARIMA

SARIMA

SARIMA (Seasonal Autoregressive Integrated Moving Average)

Is an extension of the ARIMA (Autoregressive Integrated Moving Average) model specifically designed to handle univariate time series data with seasonal components. SARIMA integrates both non-seasonal and seasonal elements, making it ideal for modelling more complex time series patterns.

The SARIMA model can be expressed as:

$$(1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p)(1 - \Phi_1 B^m - \Phi_2 B^{2m} - \ldots - \Phi_P B^{Pm})$$

where B is the backshift operator, ϕ and Φ are the coefficients for the non-seasonal and seasonal autoregressive terms, θ and Θ are the coefficients for the non-seasonal and seasonal moving average terms, and ϵ_t is the error term.

SARIMA

SARIMA (Seasonal Autoregressive Integrated Moving Average)

Let's break down the meaning of SARIMA:

S - Seasonal

SARIMA explicitly accounts for seasonality in time series data. Seasonality refers to patterns that repeat at regular intervals, such as monthly or yearly.

A - Autoregressive (AR)

The autoregressive part of the model (denoted by p) uses the dependency between an observation and a number of lagged observations (previous values). It tries to predict future values based on past values of the series.

R - Integrated (I)

The integrated part (denoted by d) represents the differencing of raw observations to make the time series stationary, meaning it has a constant mean and variance over time. Differencing helps remove trends and seasonality.

SARIMA

SARIMA (Seasonal Autoregressive Integrated Moving Average)

Let's break down the meaning of SARIMA:

M - Moving Average (MA)

The moving average part (denoted by q) uses the dependency between an observation and a residual error from a moving average model applied to lagged observations. It helps in smoothing out the noise.

SARIMA

Components of SARIMA :

The SARIMA model is represented as SARIMA(p, d, q)(P, D, Q)m, where:

- p: Order of the non-seasonal autoregressive component.
- d: Degree of non-seasonal differencing.
- q:Order of the non-seasonal moving average component.
- P: Order of the seasonal autoregressive component.
- D:Degree of seasonal differencing.
- Q:Order of the seasonal moving average component.
- m: Number of periods in each season (e.g., m = 12 for monthly data with yearly seasonality).

SARIMA

Steps to Build a SARIMA Model :

1 - **Identification:**

- Plot the time series data to check for seasonality, trends, and stationarity.
- Use ACF (Autocorrelation Function) and PACF (Partial Autocorrelation Function) plots to identify the order of non-seasonal and seasonal components.

2 - **Differencing:**

- Apply differencing to remove trends and seasonality. Non-seasonal differencing is applied d times, and seasonal differencing is applied D times with a lag of m.

SARIMA

Steps to Build a SARIMA Model :

3 - **Parameter Estimation:**

- Estimate the parameters (p, d, q, P, D, Q, m) using techniques such as Maximum Likelihood Estimation (MLE).

4 - **Model Fitting:**

- Fit the SARIMA model to the data using statistical software like R, Python (statsmodels library), etc.

SARIMA

Steps to Build a SARIMA Model :

5 - **Diagnostic Checking:**

- Check the residuals of the fitted model to ensure they resemble white noise, indicating a good fit.
- Use statistical tests and plots (e.g., Ljung-Box test) to validate the model.

6 - **Forecasting :**

- Generate forecasts using the fitted SARIMA model.

SARIMA

Application of SARIMA :

SARIMA models are widely used in various fields for forecasting purposes, including:

- **Business and Economics:** Forecasting sales, revenue, and other financial metrics.
- **Weather and Climate:** Predicting temperature, precipitation, and other weather-related variables.
- **Energy:** Forecasting electricity demand, gas consumption, and other energy-related metrics.
- **Healthcare:** Predicting disease outbreaks, hospital admissions, and other healthcare-related statistics.

XGBoost

XGBoost

How XGBoost works for time series?

Feature Engineering: Time series data needs to be transformed into a format suitable for XGBoost. This typically involves creating lagged features (e.g., previous day's values, weekly averages) and potentially incorporating other external features that might influence the target variable.

Model Training: The XGBoost model is trained on this engineered dataset, using gradient boosting to iteratively build an ensemble of decision trees .

Forecasting: The trained model is used to generate predictions for future time steps, taking into account the engineered features.

XGBoost

Advantages of using XGBoost for time series:

Handles non-linear relationships: XGBoost can capture complex non-linear patterns in time series data, unlike linear models like ARIMA.

Feature importance analysis: It provides insights into which features are most important for forecasting, helping you understand the underlying drivers of the time series .

Regularization: XGBoost includes regularization techniques to prevent overfitting, which is crucial for time series models.

Handling Missing Values and Outliers: XGBoost is capable of managing missing values and outliers in the data, minimizing the need for extensive data preprocessing.

XGBoost

Challenges and considerations:

Handling long-term dependencies: XGBoost might struggle to capture very long-term dependencies in time series data due to the nature of decision tree splits.

Data preparation: Feature engineering requires careful consideration and domain knowledge to create relevant features for forecasting .

Evaluation: Time series models should be evaluated using techniques like walk-forward validation to avoid overestimating performance due to data leakage.

XGBoost

Best practices for using XGBoost in time series:

Experiment with feature engineering: Try different combinations of lagged features, rolling statistics, and external variables.

Tune hyperparameters: Optimize parameters like learning rate, tree depth, and regularization to find the best model configuration .

Consider alternative models: If XGBoost doesn't perform well, explore other time series models like ARIMA, ETS, or Prophet.

Thank you!

SDAIA
الهيئة السعودية للبيانات
والذكاء الاصطناعي
Saudi Data & AI Authority