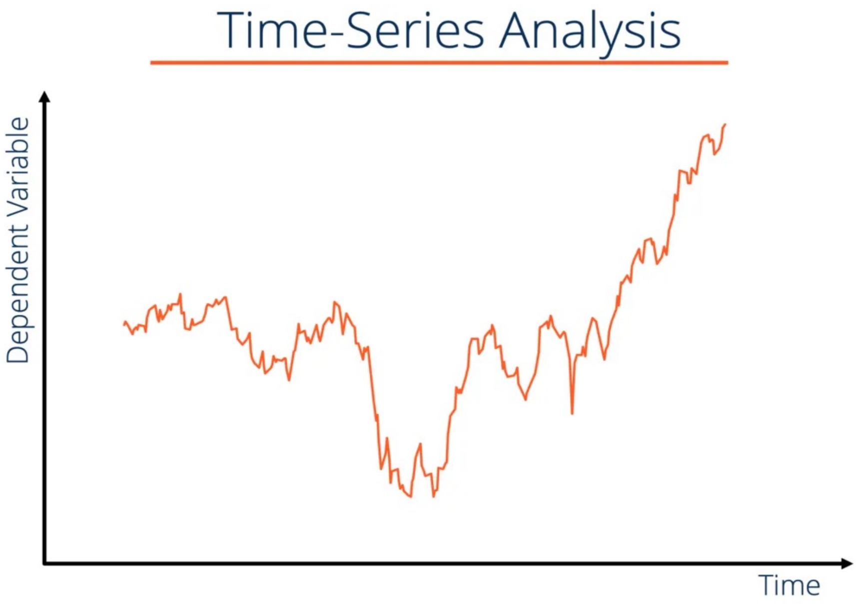
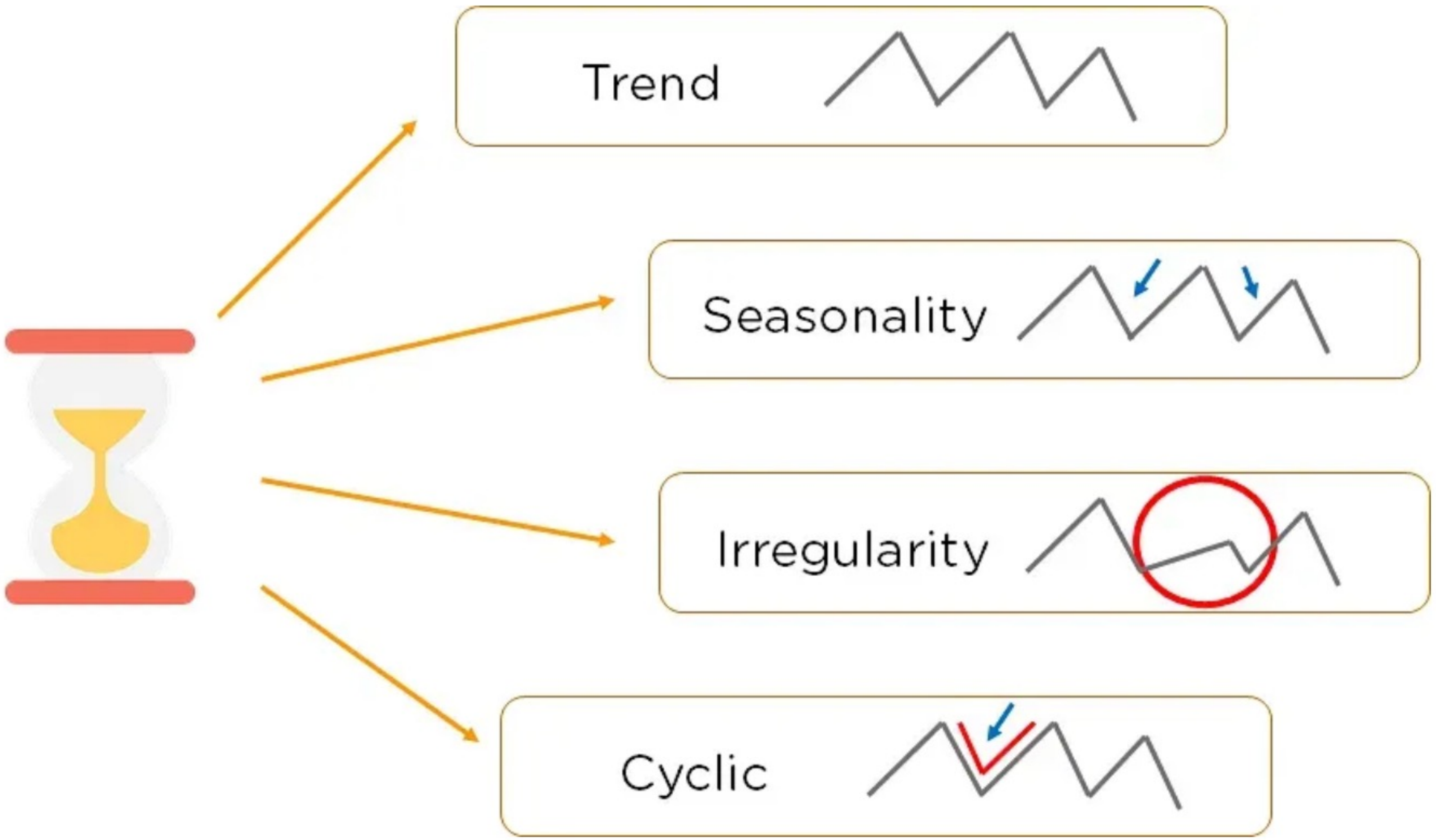
TIME SERIES ANALYSIS

Introduction to Time Series Analysis

- **Definition**: Time series analysis is a statistical technique to comprehend and evaluate data gathered over a period of time.
- One aspect of time series data is temporal dependence, where each observation is influenced by its predecessors.



Components of Time Series Data



Introduction to Time Series Analysis

Components of Time Series Data:

- **Trend**: Trends are long-term patterns or movements in time series data. Rising, falling, or stagnant trends indicate growth, decline, or no trend. Understanding dynamics and creating forecasts requires trend identification and analysis.

Introduction to Time Series Analysis: Trend

- **Trend**: Trends are long-term patterns or movements in time series data. Rising, falling, or stagnant trends indicate growth, decline, or no trend. Understanding dynamics and creating forecasts requires trend identification and analysis.

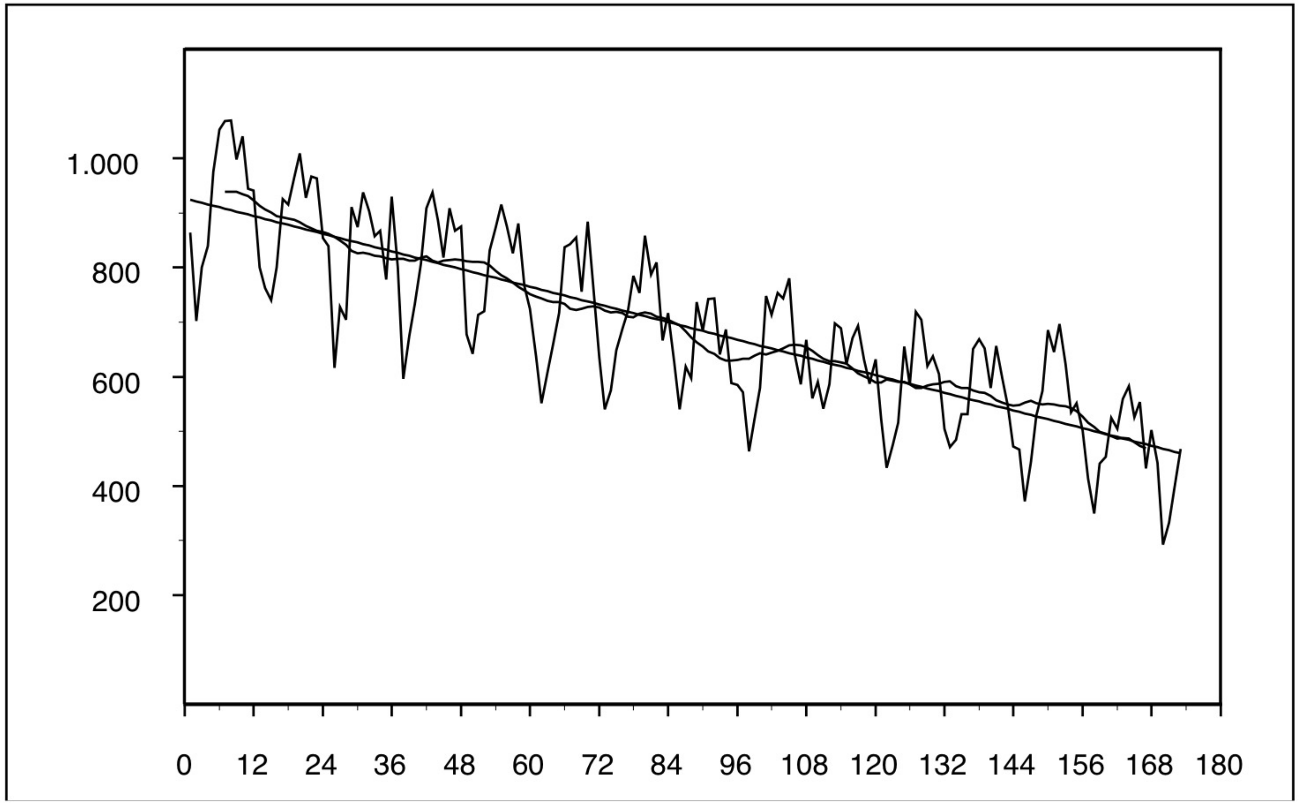
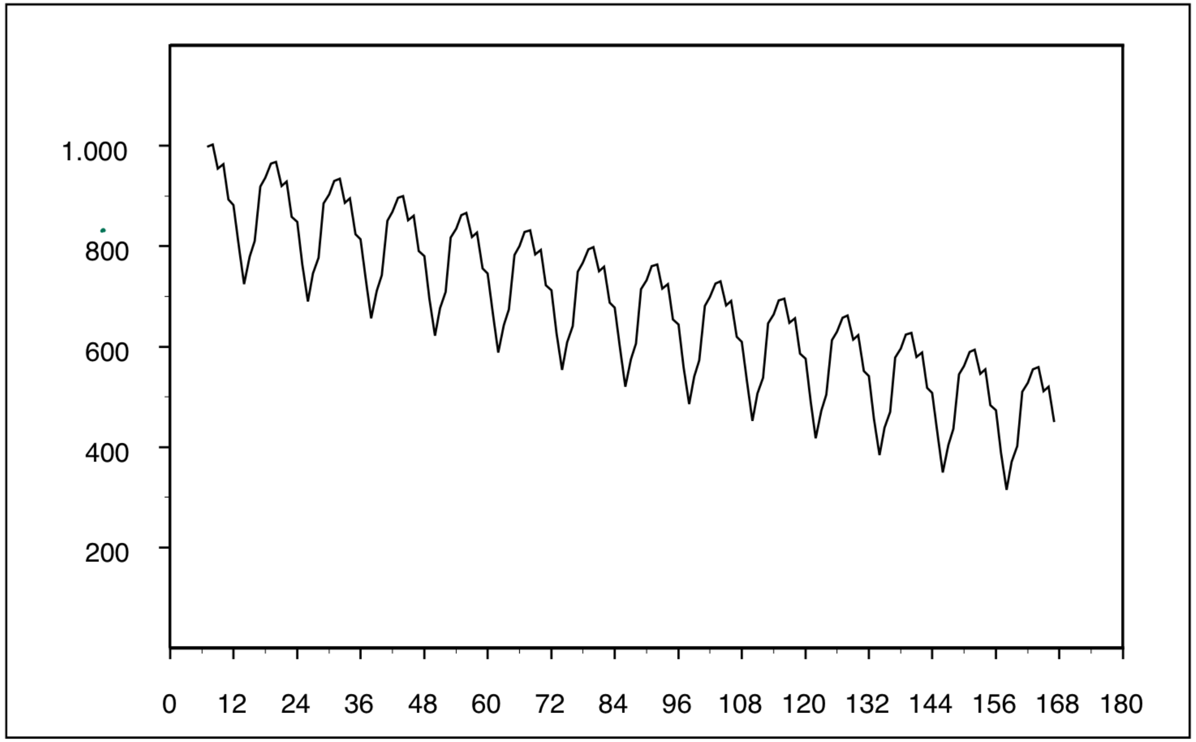


Figure: Monthly traffic-related deaths in Germany from January 1991 - May 2005 together with estimated linear trend (straight line) and an estimated trend based on a moving average over 12 months.

Time Series Components

- **Seasonality**: Seasonality refers to regular fluctuations or patterns in a time series. Holidays may boost store sales, or summer and winter temperatures may vary predictably. Seasonality affects predicting accuracy and must be considered in time series analysis.



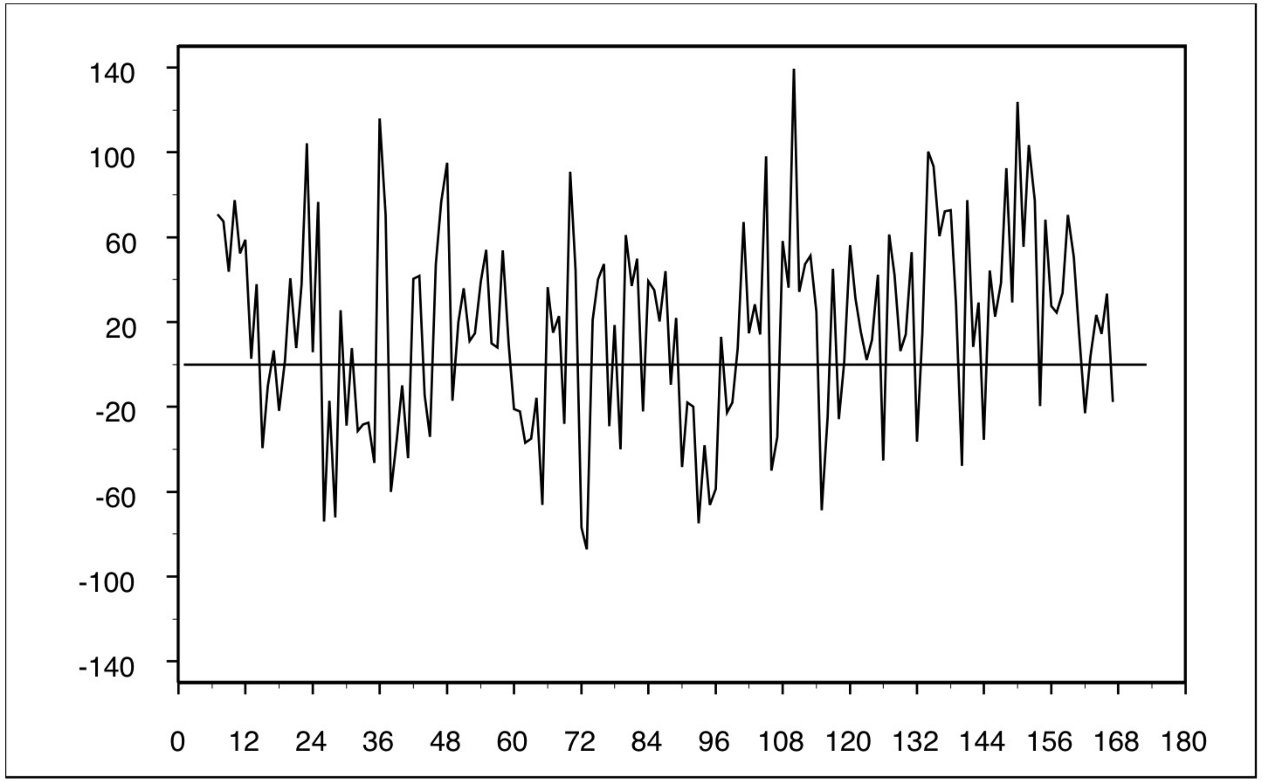
Estimated seasonal component of monthly traffic-related deaths in Germany plus an estimated linear trend (after seasonal adjustment).

Time Series Components

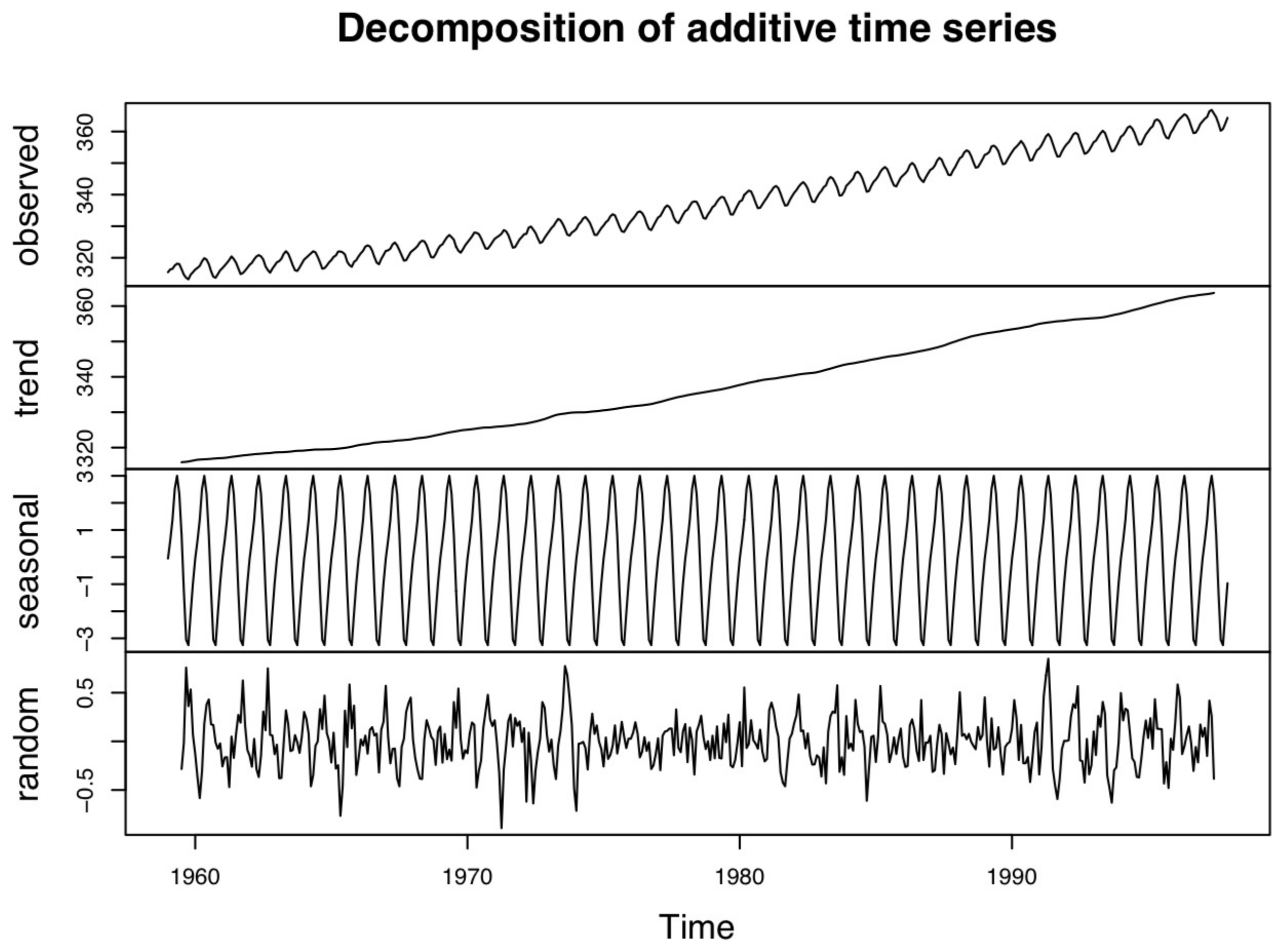
- **Randomness**: Time series data may include fluctuations or noise in addition to predictable patterns. These abnormalities may be caused by measurement errors, external influences, or random processes. Building robust models and making accurate predictions requires distinguishing signals from noise.
- **Non-stationarity**: Stationarity is the statistical stability of a time series over time. In many real-world situations, time series data has non-stationarity when mean, variance, or other statistical features fluctuate. Time series analysis methods must detect and address non-stationarity to be valid and successful.

Time Series Components

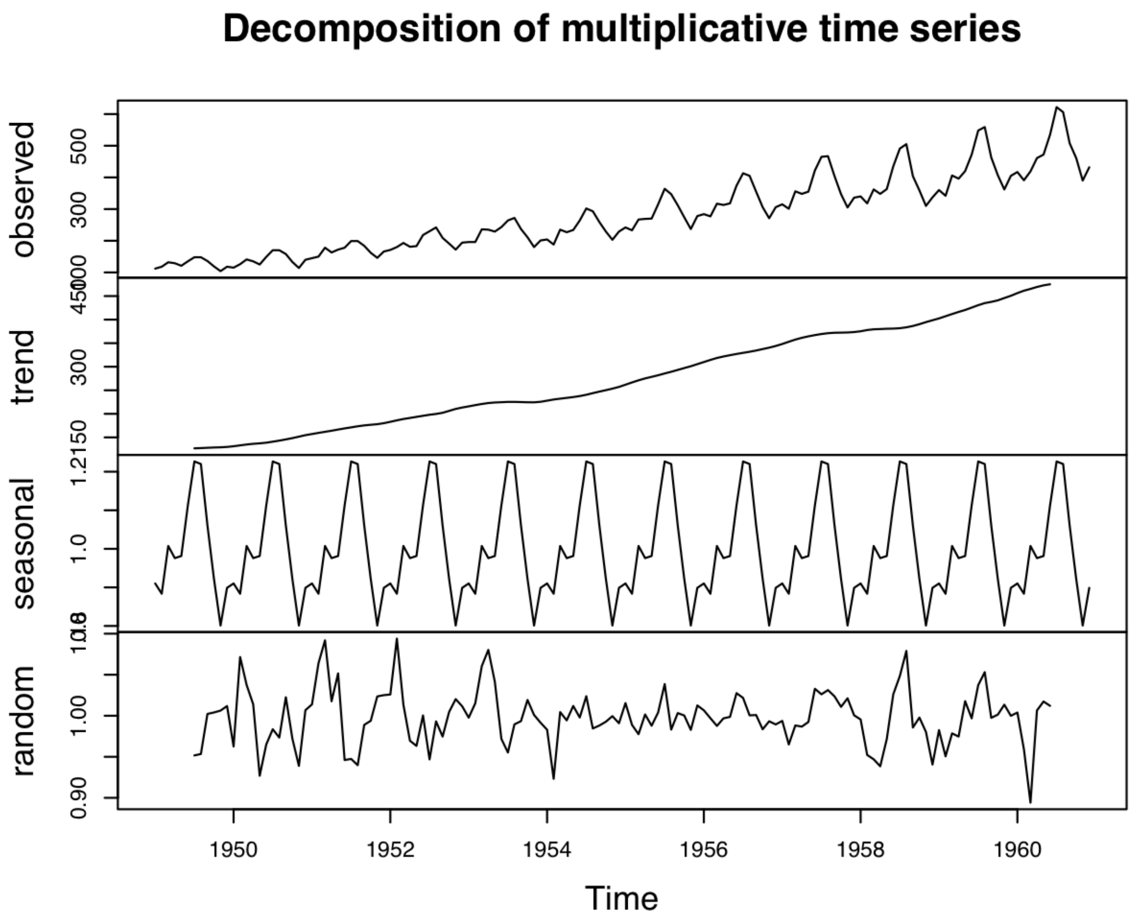
- Detrended and seasonally adjusted monthly traffic-related deaths in Germany



Time Series Components



Time Series Components



Application of Time Series

- Understand the Data
  - Identify Patterns and Trends
  - Detect Seasonal Variations
  - Identify Cyclic Patterns
- Make forecasts

Forecasting: Naive Forecasting

- The **Naive Forecasting** is the simplest form of forecasting where the forecast for the next period is equal to the last observed value.

- **Advantages**:
  - Easy to Understand and Implement
  - Useful as a Benchmark

- **Disadvantages**:
  - Ignores Trends and Seasonality
  - Often Not Accurate for Complex Datasets

Forecasting: Moving Average (MA)

- The Moving Average is a technique to smooth out short-term fluctuations and highlight longer-term trends or cycles in the data.
- **Calculation**:
  - Choose the number of past periods
  - Calculate the average over them

- **Advantages:**
  - Smoothing Effect
  - Flexibility

- **Disadvantages:**
  - Lag Effect
  - Doesn't Handle Seasonality Well

Forecasting: Exponential Smoothing

- Exponential Smoothing is a time series forecasting method that uses weighted averages of past observations to forecast future values.
- The weights decrease exponentially as observations get older.

  - **Simple Exponential Smoothing**: Suitable for data with no clear trend or seasonality.
  - **Double Exponential Smoothing** (Holt's Linear Trend Model): Extends simple exponential smoothing to capture trends in the data.

Simple Exponential Smoothing

- **Simple Exponential Smoothing** is used for time series data without a trend or seasonal pattern.
- It applies a single smoothing constant to the entire series.

S_t: Forecast for time point t
y_t : Actual value at time t
α: Smoothing factor for the level

$$S_t = \alpha y_t + (1 - \alpha) S_{t-1}$$

- **Smoothing Constant (α):** Determines the weight given to the most recent observation.
  - Higher α gives more weight to recent observations.
  - Lower α gives more weight to past observations.

Double Exponential Smoothing (Holt's Method)

- **Holt's Linear Exponential Smoothing**, also known as double exponential smoothing, is used to forecast time series data that has a linear trend but no seasonal pattern
- **Components**:
  - Level - smoothed value at time

$$L_t = \alpha Y_t + (1-\alpha)(L_{t-1} + T_{t-1})$$

$$T_t = \beta(L_t - L_{t-1}) + (1-\beta)T_{t-1}$$

  - Trend - estimated trend at time t.

$$F_{t+k} = L_t + kT_t$$

- **Smoothing Constants**:
  - α for the level: controls the speed at which the level component reacts to changes.
  - β for the trend: controls the speed at which the trend component reacts to changes.

Forecasting: Autoregressive models (AR)

- **Autoregressive (AR) models** are used in time series analysis to predict future values based on past values of the same variable.
- The current value of the series is modeled as a linear combination of its previous values plus a random error term.
- AR models assume the time series is **stationary**, meaning its statistical properties do not change over time.

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \ldots + \phi_p Y_{t-p} + \epsilon_t$$

Y_t: The value of the time series at time tt.

$\phi_1, \phi_2, \ldots$ The coefficients (parameters) of the model that need to be estimated. These coefficients determine the impact of past values on the current value.

p: The order of the AR model, indicating the number of lagged observations included

ϵ_t: The error term (also known as white noise) at time tt.

Forecasting: Autoregressive Moving Average Model (ARMA)

- The ARMA model combines both AR (autoregressive) and MA (moving average) components to predict future values.

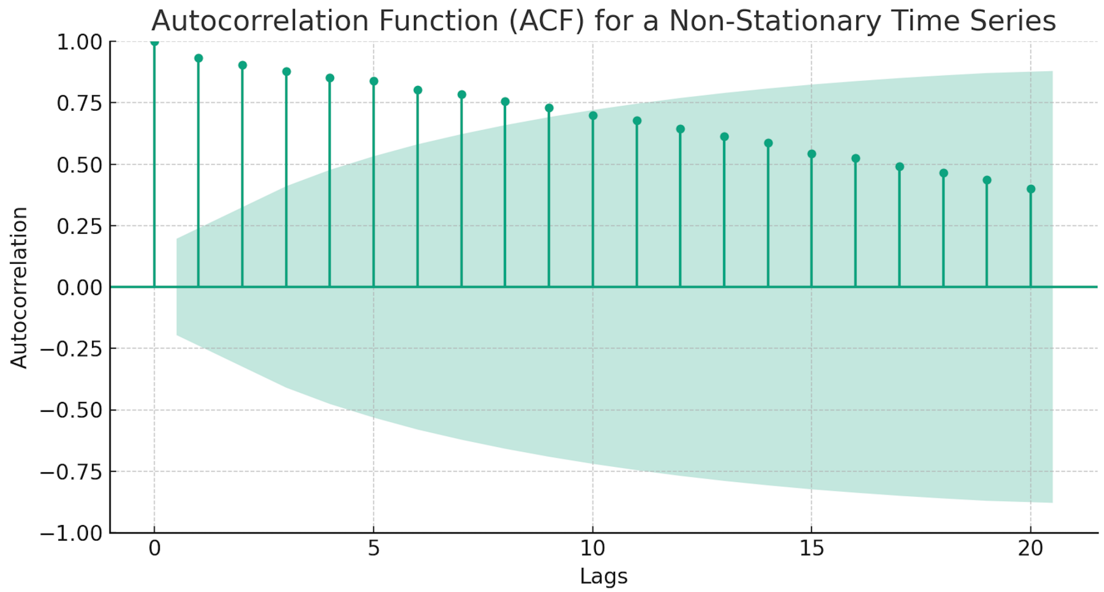
$$Y_t = \phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p} + \theta_1 \epsilon_{t-1} + \ldots + \theta_q \epsilon_{t-q} + \epsilon_t$$

- Where ϕ are AR parameters, θ are MA parameters, and ϵ is the error term.

- Order (p, q):
  - p is the number of lagged observations (AR part).
  - q is the number of lagged forecast errors (MA part).

Autocorrelation Function (ACF)

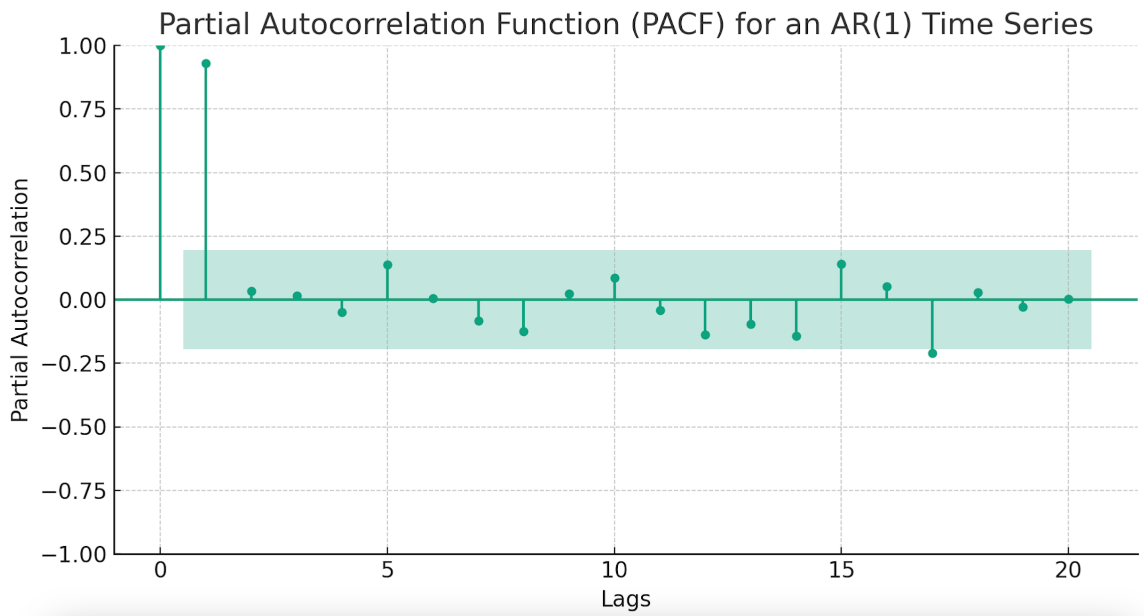
- Measures the correlation between a time series and lagged versions of itself over intervals of increasing length.
- Helps to identify the appropriate **MA** (Moving Average) terms (q) in an ARIMA model.



$$\mathrm{ACF}(k) = \frac{\sum_{t=k+1}^{T}(Y_t - \overline{Y})(Y_{t-k} - \overline{Y})}{\sum_{t=1}^{T}(Y_t - \overline{Y})^2}$$

Partial Autocorrelation Function (PACF)

- Measures the extent of correlation between the series and its lag, after eliminating the variations explained by their in-between lags.
- Helps to identify the appropriate **AR** (Autoregressive) terms (p) in an ARIMA model.



$$\mathrm{PACF}(k) = \mathrm{Corr}(Y_t, Y_{t-k} | Y_{t-1}, Y_{t-2}, \ldots, Y_{t-k+1})$$

ARIMA models for time series data

1. **AutoRegressive (AR) Component**: This component models the relationship between the current observation and its previous observations, also known as lagged values. It captures the linear relationship between the variable and its own lagged values.

2. **Moving Average (MA) Component:** This component models the relationship between the current observation and a residual error from a moving average model applied to lagged observations. It captures the short-term, random fluctuations in the data series.

3. **Integrated (I) Component:** This component involves differencing the time series data to make it stationary. Stationarity implies that the statistical properties of a process do not change over time. By differencing, we aim to remove trends and seasonality.

Forecasting: Autoregressive Integrated Moving Average Model

- ARIMA models extend ARMA models to include differencing to make the time series stationary.

$$\Delta^d Y_t = \phi_1 \Delta^d Y_{t-1} + \ldots + \phi_p \Delta^d Y_{t-p} + \theta_1 \epsilon_{t-1} + \ldots + \theta_q \epsilon_{t-q} + \epsilon_t$$

Δ^d indicates differencing d times to achieve stationarity (subtracting the current value of the series from the previous one)

Order (p, d, q):

- p: Number of lagged observations (AR part).
- d: Number of differences needed to make the series stationary (I part).
- q: Number of lagged forecast errors (MA part).

Forecasting: Seasonal ARIMA with Exogenous Variables (SARIMAX)

- SARIMAX models extend ARIMA models to include seasonal components and external (exogenous) variables.

$$Y_t = \phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p} + \theta_1 \epsilon_{t-1} + \ldots + \theta_q \epsilon_{t-q} + \gamma_1 X_{1,t} + \ldots + \gamma_k X_{k,t} + \epsilon_t$$

- **Order (p, d, q) and Seasonal Order (P, D, Q, s):**
  - p,d,q: Non-seasonal ARIMA orders.
  - P,D,Q: Seasonal ARIMA orders.
  - s: Length of the seasonal cycle.

- fg

Model Choice Using Information Criteria

- Bayesian Information Criterion: $$BIC = \log(m)p - 2\log\left(\hat{L}\right)$$
- Akaike Information Criterion: $$AIC = 2p - 2\log\left(\hat{L}\right)$$

  - m is the number of data points,
  - p is the number of parameters learned by the model,
  - L is the maximized value of the likelihood function of the model (the probability of the observed data given the model).

Model Choice: AIC

- It helps in model selection by balancing the trade-off between the goodness of fit and model complexity.
- Lower AIC values indicate a better model.
- AIC is used to compare multiple models: the model with the lowest AIC is preferred.
- It penalizes models with more parameters to prevent overfitting.
-

- **Limitations**:
  - Assumes the model is correctly specified.
  - May not always select the true model, especially with small sample sizes.

Model Choice: BIC

- It introduces a stronger penalty for models with more parameters compared to AIC
- Lower BIC values indicate a better model.
- Like AIC, BIC is used to compare multiple models: the model with the lowest BIC is preferred.
- BIC includes the sample size (n) in its calculation, making it more stringent for larger datasets.

- **Limitations**:
  - Can be overly strict on small sample sizes.
  - Like AIC, assumes the model is correctly specified.