Défis en Intelligence Artificielle

Défi 3 : L'IA pour l'analyse et la prévision de séries temporelles (I/III)

Souhaib Ben Taieb

University of Mons

December 8, 2022

UMONS
Université de Mons

Faculté
des Sciences

Overview

Course organization

Introduction to time series forecasting

Concepts and tools for time series data

- Time series patterns
- Time series plots
- The autocorrelation function
- Stationarity
- Transformations
- Time series decomposition

Methods and tools for time series forecasting

- Overview of forecasting methods
- Challenges in training AI models for forecasting
- Some simple forecasting methods
- Residual diagnostics
- Evaluating forecast accuracy

Outline

Course organization

Introduction to time series forecasting

Concepts and tools for time series data

Methods and tools for time series forecasting

Teaching assistants

Tanguy BOSSER
PhD candidate
Big Data and Machine Learning Lab
Department of Computer Science
`tanguy.bosser@umons.ac.be`

Victor DHEUR
PhD candidate
Big Data and Machine Learning Lab
Department of Computer Science
`victor.dheur@umons.ac.be`

Schedule

- Week 1: 8 December 2022, 6pm-9pm
- Week 2: 15 December 2022, 6pm-9pm
- Week 3: 22 December 2022, 6pm-9pm

Assessment

- Kaggle competition on Time Series Forecasting (More details soon)
- Google Colab or Kaggle Notebooks

Task	**Due Date**	**Value**
Project		100%
$\rightarrow$ Kaggle submission	22 January 2023, 11:55pm	35%
$\rightarrow$ Report	25 January 2023, 11:55pm	65%

Communication

- Moodle
  - https://moodle.umons.ac.be/course/view.php?id=2666#section-4
  - Forum for asking questions
  - Project submissions

- Course webpage (GitHub)
  - https://github.com/bsouhaib/Hands-On-AI-2022-Challenge3

- **No email please — use the forum**

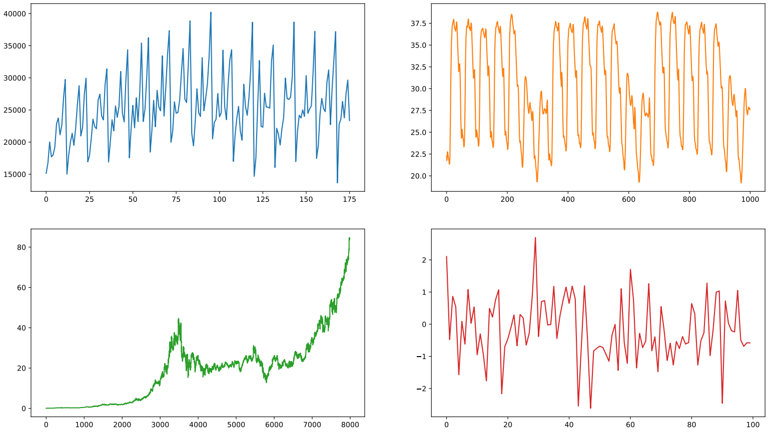
Outline

What is Forecasting?

"Forecasting is a sub-discipline of prediction in which we are making predictions **about the future** *(on the basis of* **time-series data***). Thus, the only difference between prediction and forecasting is that we consider the* **temporal dimension***."*

Time series data

A (regular) time series is a **series of data points** observed at **successive** (equally spaced) points in time.

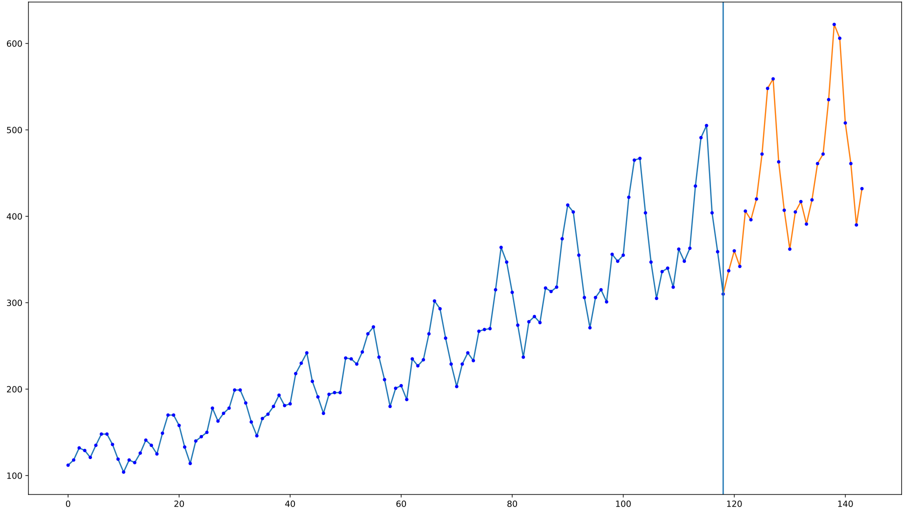


Different types of data

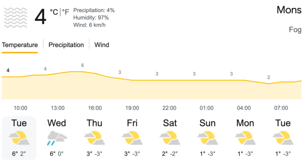
- Image (Défi 1)
- Tabular data
- Graphs
- Sequential data
  - Text (Défi 2)
  - Video
  - Dynamic graphs
  - ...
- **Time series/signals**
  - Quantity measured over time (Défi 3)
  - Audio
  - Sensor measurements
  - ...
- ...

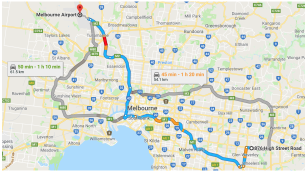
Time series forecasting

Forecasting is estimating how the sequence of observations will continue into the future.



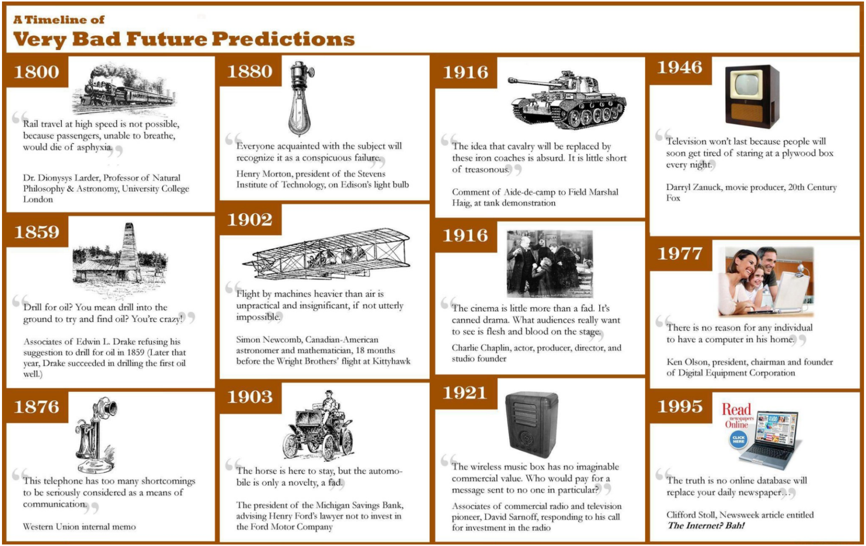
Forecasts are crucial for decision making





Note the different **forecastability** of these phenomena/applications.

Forecasting is difficult



Forecasting is difficult

Forecasting for COVID-19 has failed

John P.A. Ioannidis [a,*], Sally Cripps [b], Martin A. Tanner [c]

[a] *Stanford Prevention Research Center, Department of Medicine, and Departments of Epidemiology and Population Health, of Biomedical Data Science, and of Statistics, Stanford University, and Meta-Research Innovation Center at Stanford (METRICS), Stanford, CA, USA*
[b] *School of Mathematics and Statistics, The University of Sydney and Data Analytics for Resources and Environments (DARE) Australian Research Council, Sydney, Australia*
[c] *Department of Statistics, Northwestern University, Evanston, IL, USA*

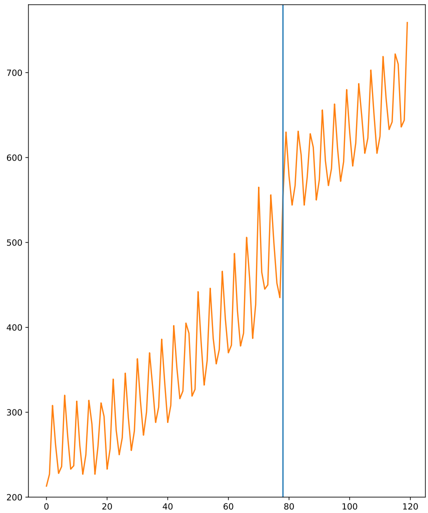
A R T I C L E   I N F O

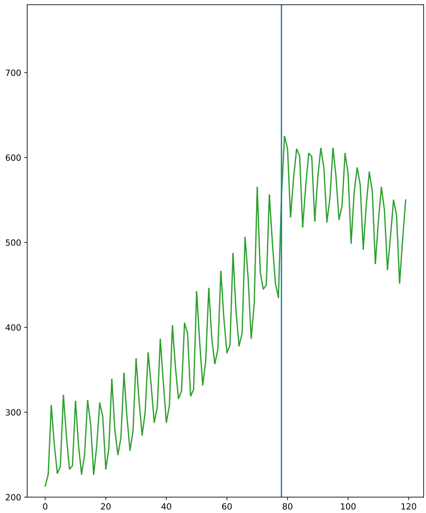
Keywords:
Forecasting
COVID-19
Mortality
Hospital bed utilization
Bayesian models
SIR models
Bias
Validation

A B S T R A C T

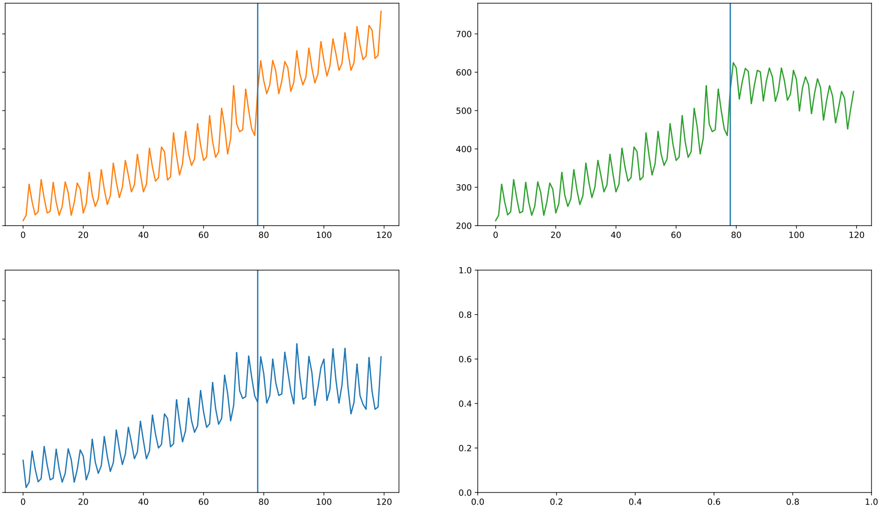
Epidemic forecasting has a dubious track-record, and its failures became more prominent with COVID-19. Poor data input, wrong modeling assumptions, high sensitivity of estimates, lack of incorporation of epidemiological features, poor past evidence on effects of available interventions, lack of transparency, errors, lack of determinacy, consideration of only one or a few dimensions of the problem at hand, lack of expertise in crucial disciplines, groupthink and bandwagon effects, and selective reporting are some of the causes of these failures. Nevertheless, epidemic forecasting is unlikely to be abandoned. Some (but not all) of these problems can be fixed. Careful modeling of predictive distributions rather than focusing on point estimates, considering multiple dimensions of impact, and continuously reappraising models based on their validated performance may help. If extreme values are considered, extremes should be considered for the consequences of multiple dimensions of impact so as to continuously calibrate predictive insights and decision-making. When major decisions (e.g. draconian lockdowns) are based on forecasts, the harms (in terms of health, economy, and society at large) and the asymmetry of risks need to be approached in a holistic fashion, considering the totality of the evidence.

Time series forecasting





Time series forecasting



Factors affecting forecastability

A quantity is easier to forecast if:

- We have a **good understanding** of the underlying phenomenon and the factors that contribute to it
- There is **lots of data** available
- The future is somewhat **similar to the past**
- There is relatively **low** natural/unexplainable **random variation**
- The forecasts cannot **affect** the quantity we are trying to forecast

The A.I./statistical forecasting perspective

- The quantity we want to forecast, y_t, is a **random variable**.
- We are interested in the random variable y_{T+h} **given what we know** in $\mathcal{I} = \{y_1, y_2, \ldots, y_T\}$, i.e.

$$y_{T+h} | \mathcal{I},$$

for the **forecast horizons** $h = 1, 2, \ldots, H$.

The A.I./statistical forecasting perspective

- A "**point forecast**" is an estimate of the mean, median or quantile of $y_{T+h}|\mathcal{I}$.

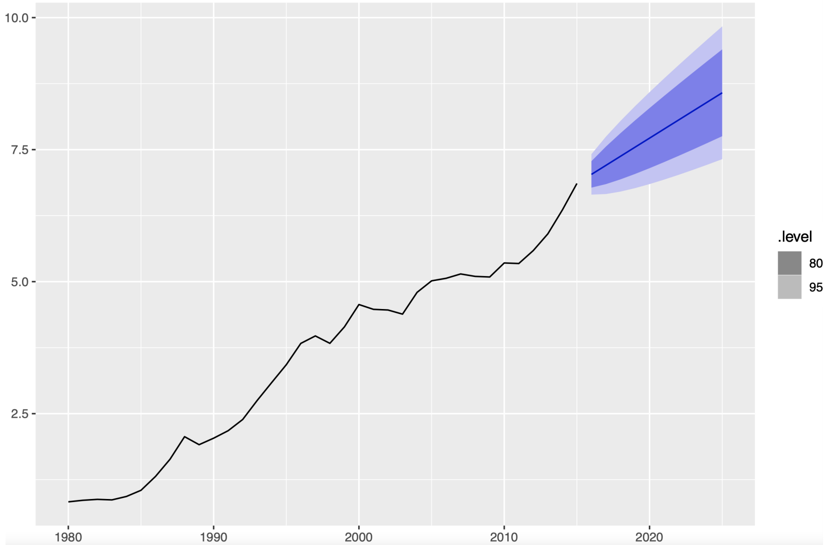
- A "**variance forecast**" is an estimate of the variance of $y_{T+h}|\mathcal{I}$.

- A "**prediction/forecast region**" with coverage $\alpha \in [0, 1]$ is a set of values of y_{T+h} with probability coverage α. If the region is an interval, we call it a "**predition/forecast interval**".

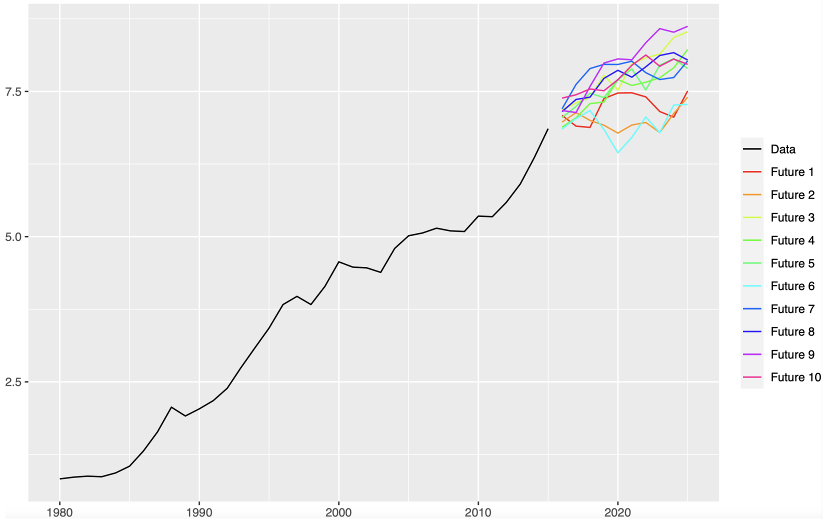
- More generally, a "**probabilistic forecast**" is an estimate of the distribution of $y_{T+h}|\mathcal{I}$.

$\rightarrow$ We will focus on **"point forecasts"**.

The A.I./statistical forecasting perspective



The A.I./statistical forecasting perspective



Outline

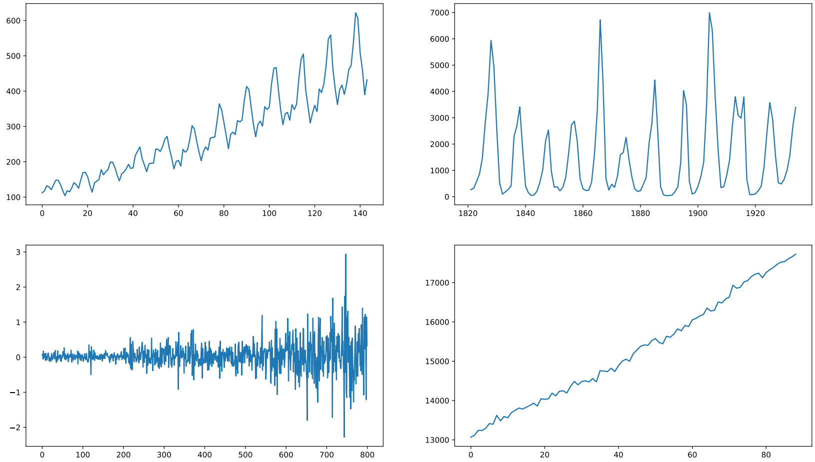
Outline

Time series patterns

Many time series include **trend**, **cycles** and **seasonality**.

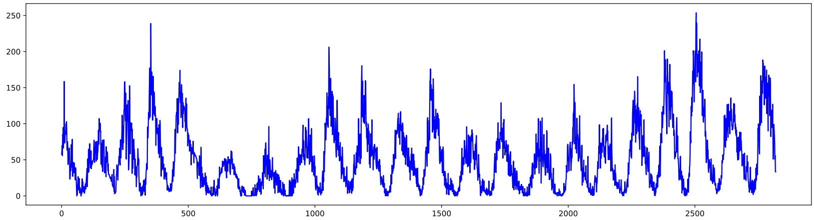
- **Trend**: a long-term increase or decrease in the data.
- **Cyclical**: data exhibits rises and falls that are not of fixed period (duration usually of at least 2 years)
- **Seasonal**: occurs when a time series is affected by seasonal factors such as the time of the year or the day of the week. It has a fixed and known frequency.

Examples of time series patterns

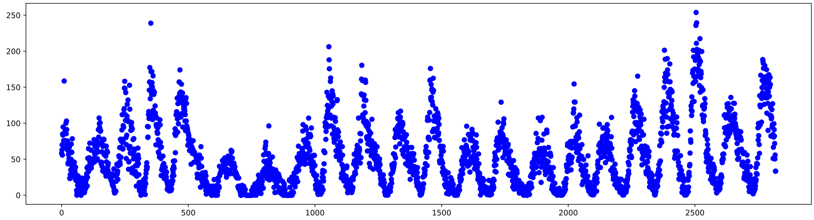


Outline

Time plots

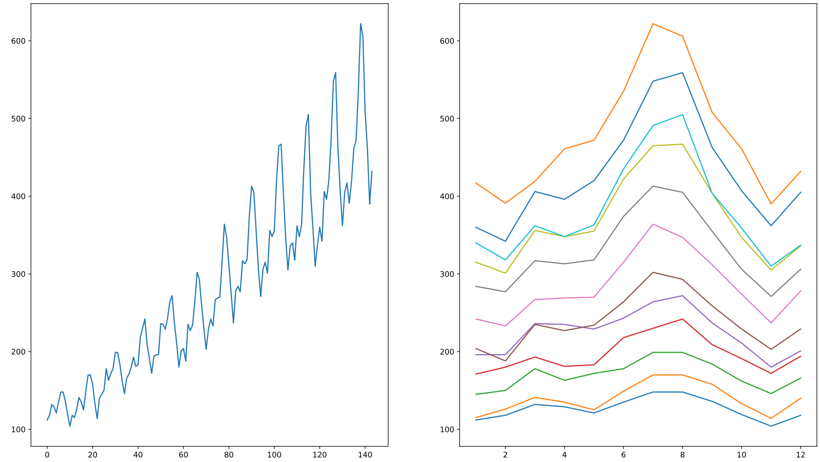


250
200
150
100
50
0
0
500
1000
1500
2000
2500

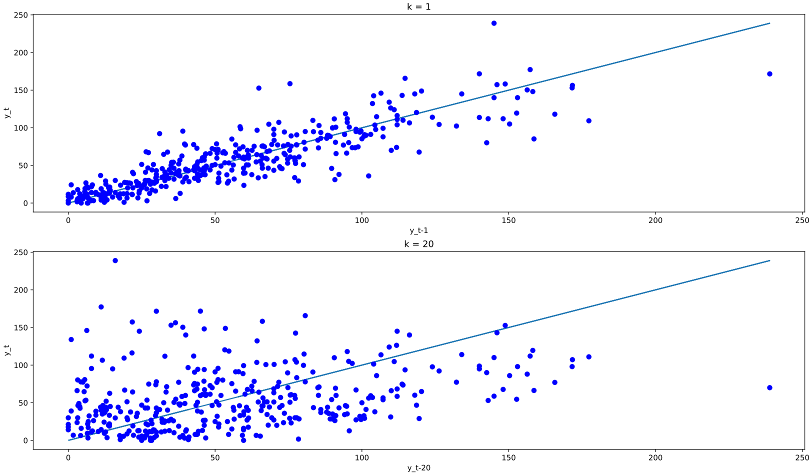


250
200
150
100
50
0
0
500
1000
1500
2000
2500

Seasonal plots



Lagged scatterplots



Each graph shows y_t plotted against y_{t-k} for different values of k.

Outline

Autocorrelation

- The autocorrelations are the correlations associated with the lagged scatterplots.
- Covariance and correlation: measure **linear** relationship between two **variables** x and y.
- Autocovariance and autocorrelation: measure **linear** relationship between two **lagged variables** of a time series y_t:
  - y_t and y_{t-1}
  - y_t and y_{t-2}
  - ...
  - y_t and y_{t-k}

Autocorrelation function

The sample **autocovariance** at lag k is given by

$$c_k = \sum_{t=k+1}^{T} (y_t - \bar{y})(y_{t-k} - \bar{y}), \quad \bar{y} = \frac{1}{T}\sum_{t=1}^{T} y_t.$$

The sample **autocorrelation** at lag k is given by

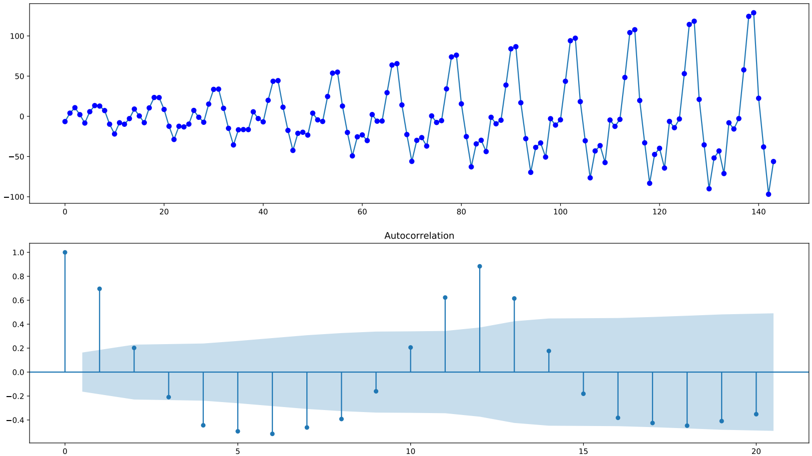
$$r_k = c_k / c_0.$$

We compute the **autocorrelation function** (**ACF**) at different lags:

- $r_1 = \text{Correlation}(y_t, y_{t-1})$
- $r_2 = \text{Correlation}(y_t, y_{t-2})$
- $\ldots$
- $r_k = \text{Correlation}(y_t, y_{t-k})$

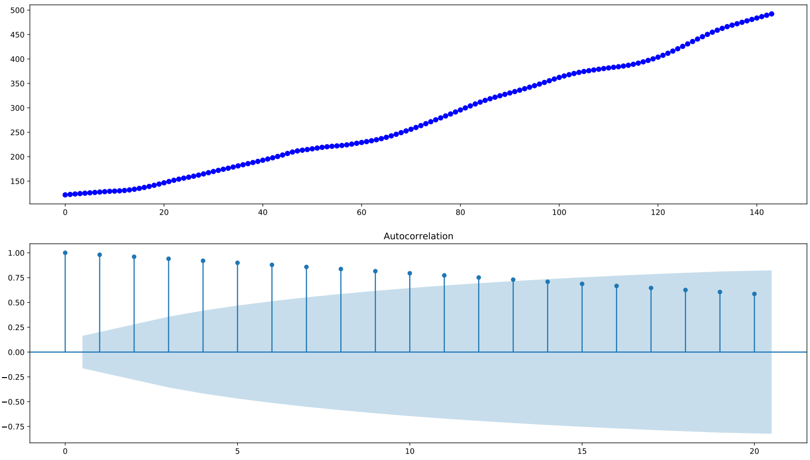
The plot of the ACF at different lags is known as the **correlogram**.

Autocorrelation function (seasonality)



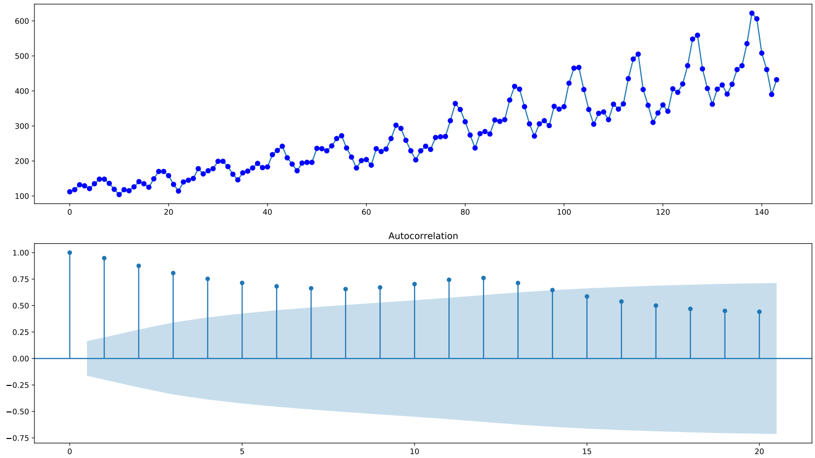
When data are seasonal, the autocorrelations will be larger at the seasonal lags (i.e., at multiples of the seasonal frequency).

Autocorrelation function (trend)



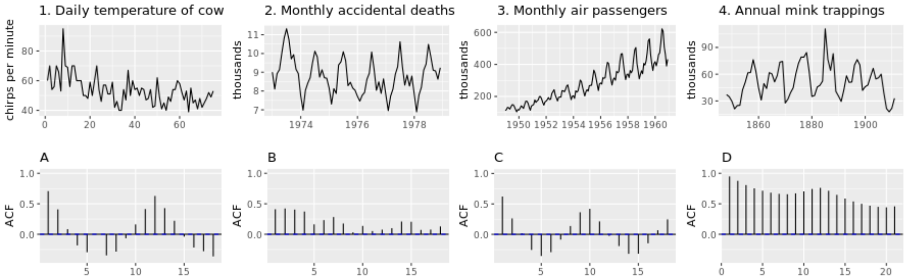
When data have a trend, the autocorrelations for small lags tend to be large and positive.

Autocorrelation function (trend and seasonality)



When data are trended and seasonal, you see a combination of these effects.

Match the ACF to the time series



Match the ACF plots shown (A-D) to their corresponding time plots (1-4).

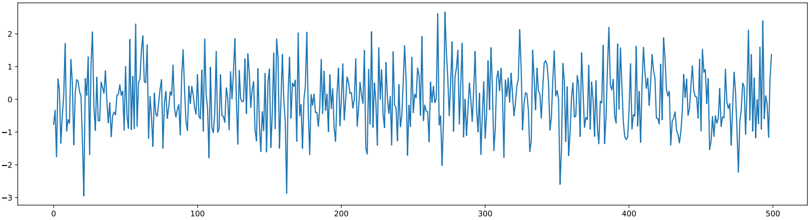
- Use the process of elimination - pair the easy ones first, then see what is left.
- Trends induce positive correlations in the early lags.
- Seasonality will induce peaks at the seasonal lags.
- Cyclicity induces peaks at the average cycle length.

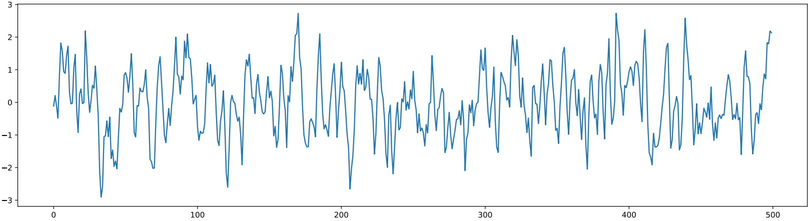
White noise

White noise data is uncorrelated across time with zero mean and constant variance (Technically, we require independence as well.).

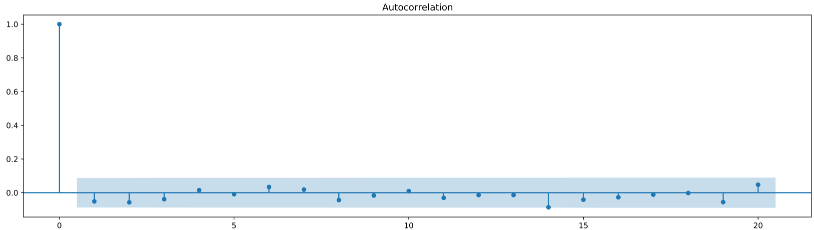
- We expect each autocorrelation to be close to zero. However, with a finite sample, they will not be exactly equal to zero due to random variation.
- Sampling distribution of r_k for white noise data is asymptotically $N(0, \frac{1}{T})$.
- For white noise data, we expect 95% of all r_k to lie within $+/-1.96/\sqrt{T}$. These are the critical values.
- If this is not the case, the series is probably not white noise.

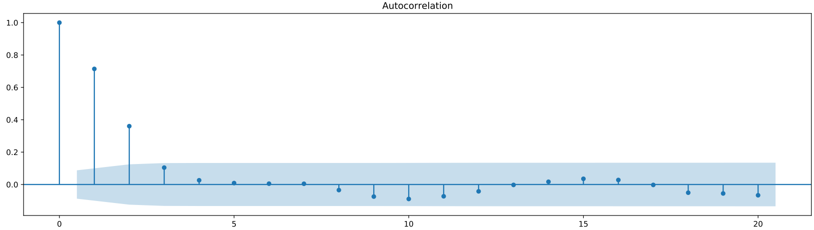
White noise





White noise


Autocorrelation
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20


Autocorrelation
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20

Serial correlation tests

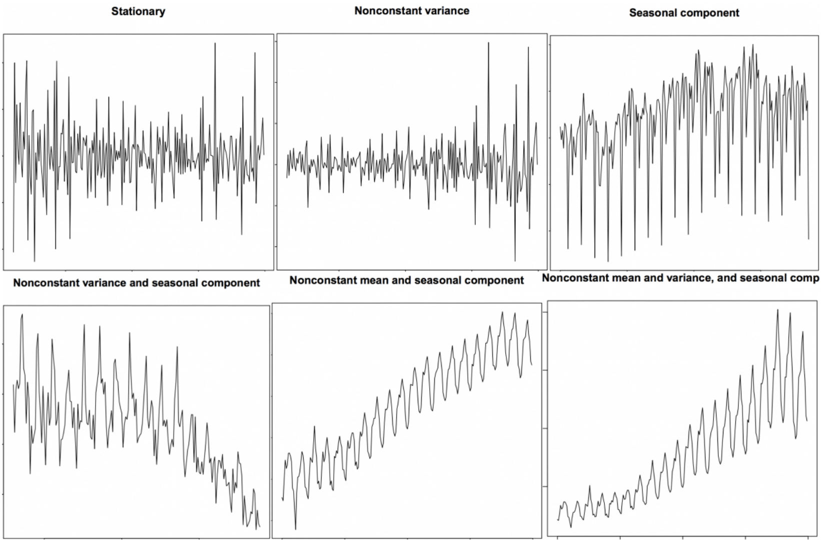
- **Ljung-Box test** considers the first h autocorrelation values together.
- A significant test (small p-value) indicates the data are probably not white noise.
- See *statsmodels.stats.diagnostic.acorr_ljungbox*

Outline

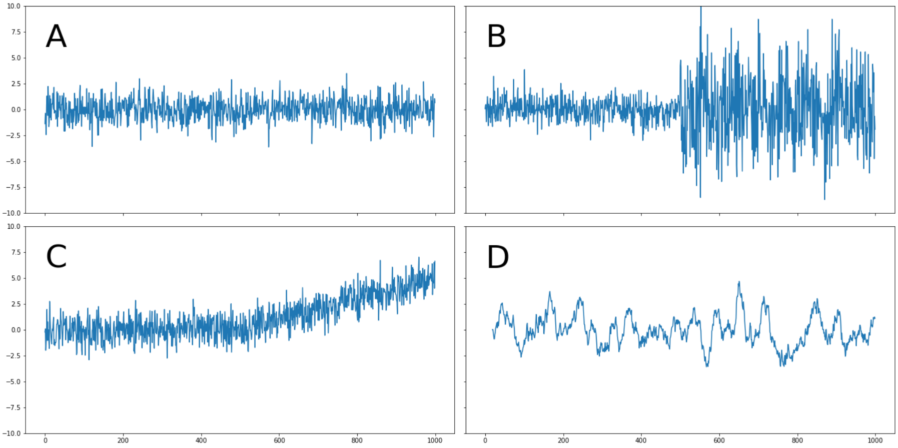
Stationarity

If $\{y_t\}$ is a **stationary** time series, then for all s, the distribution of $(y_t, y_{t+1}, \ldots, y_{t+s})$ does not depend on t.

Stationarity



Stationarity



Which of the series are non-stationary?

Outline

Transformations to stabilize the variance

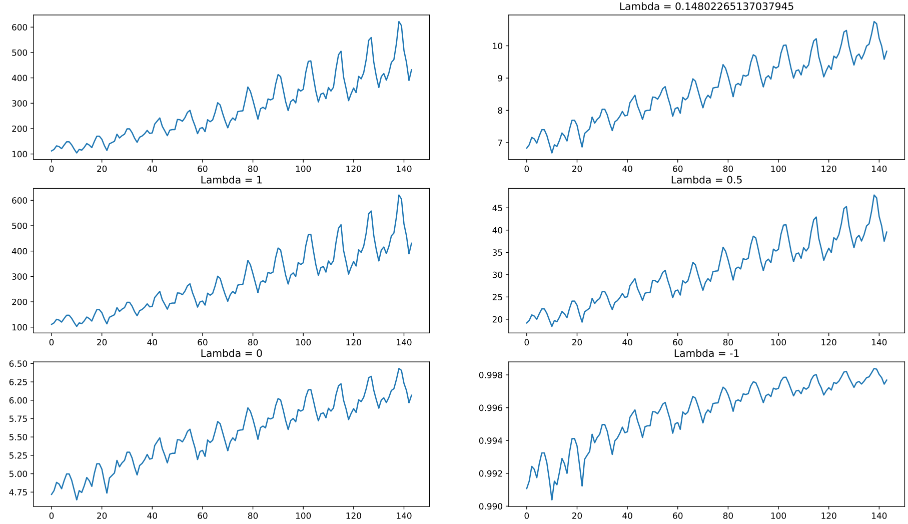
- If the data show different variation at different levels of the series, then a mathematical transformation for stabilizing variation can be useful.
- Transformations must be reversed to obtain forecasts on the original scale.
- If the data contains zeros, then don't take logs. The transformation $\log(y_t + 1)$ can be useful for data with zeros. See np.logp1.
- Choosing logs is a simple way to force forecasts to be positive.

Box-Cox transformations

$$z_t = \begin{cases} \log(y_t), & \lambda = 0; \\ (y_t^\lambda - 1)/\lambda & \lambda \neq 0. \end{cases}$$

- $\lambda = 1$: No substantive transformation.
- $\lambda = \frac{1}{2}$: Square root plus linear transformation.
- $\lambda = 0$: Natural logarithm.
- $\lambda = -1$: Inverse plus one.

Box-Cox transformations



Transformations to stabilize the mean

- **Differencing** helps to stabilize the mean.
- The **first difference** is the change between each observation in y_t:

$$z_t = y_t - y_{t-1}.$$

- Only $T - 1$ values since z_1 is undefined.
- Occasionally, a **second-order difference** may be necessary:

$$z_t^{'} = z_t - z_{t-1}.$$

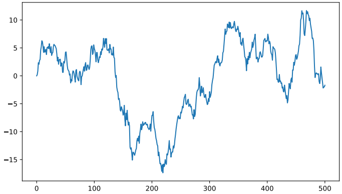
- In practice, it is rare to go beyond second-order differences.
- A **seasonal difference** is the difference between an observation and the corresponding observation from the previous year:

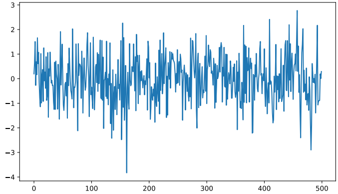
$$z_t = y_t - y_{t-m},$$

where m is the number of seasons (e.g. $m = 12$ for monthly data).

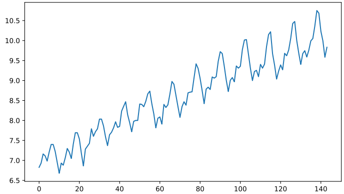
Examples (mean stabilization)

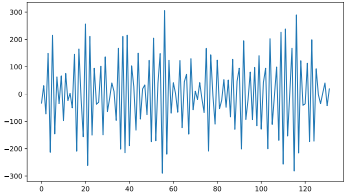
- First difference: $z_t = y_t - y_{t-1}$



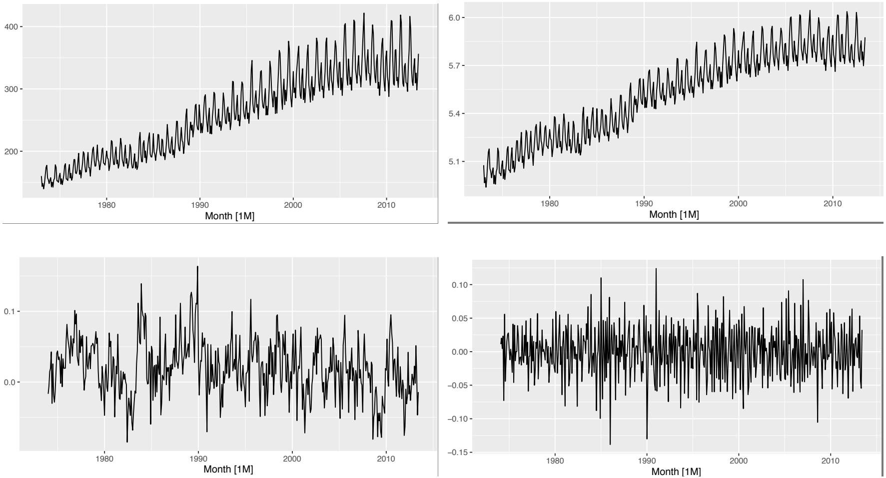


- Seasonal difference (period = 12): $z_t = y_t - y_{t-12}$





Examples (mean and variance stabilization)



(1) Initial series; (2) Log transformation; (3) Seasonal difference; (4) first difference

Unit root tests (stationarity tests)

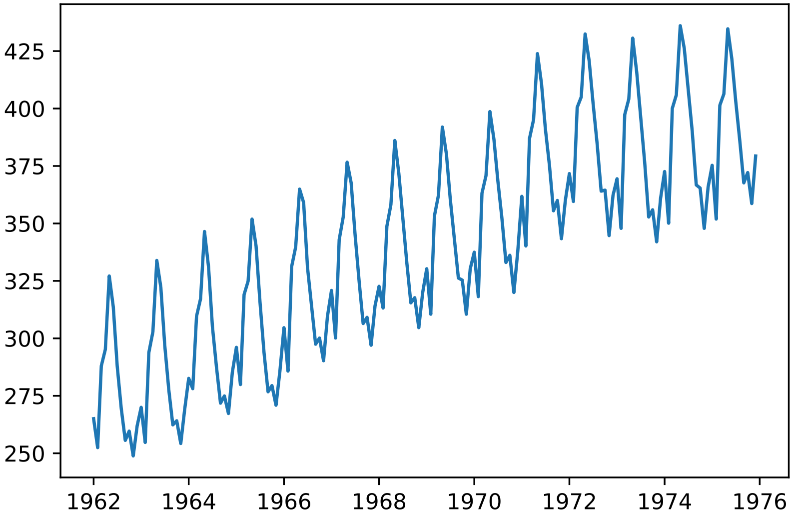
- Statistical tests to determine the required order of differencing
- **Augmented Dickey–Fuller** test: null hypothesis is that the data are non-stationary and non-seasonal.
- A significant test (small p-value) indicates the data are probably stationary.
- They do not detect nonstationarity of the seasonal kind. Other tests available for seasonal data.
- See *statsmodels.tsa.stattools.adfuller*

Outline

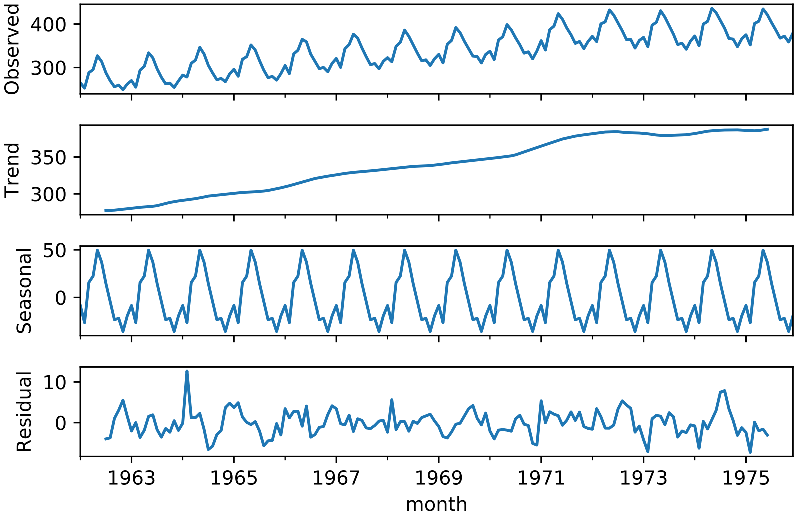
Time series decomposition

- Three types of time series patterns: **trend**, **seasonality** and **cycles**.
- When we decompose a time series into components, we usually combine the trend and cycle into a single **trend-cycle** component (sometimes called the trend for simplicity).
- Thus we think of a time series as comprising *three components*:
  - a **trend-cycle** component
  - a **seasonal** component
  - a **remainder** component (containing anything else in the time series)

Time series decomposition - example



Time series decomposition - example



Time series decomposition

If we assume an **additive decomposition**, then we can write

$$y_t = S_t + T_t + R_t,$$

where y_t is the data, S_t is the seasonal component, T_t is the trend-cycle component, and R_t is the remainder component, all at period t.

Alternatively, a **multiplicative decomposition** would be written as

$$y_t = S_t \times T_t \times R_t.$$

Note that

$$y_t = S_t \times T_t \times R_t \quad \text{is equivalent to} \quad \log y_t = \log S_t + \log T_t + \log R_t.$$

In other words, when a **log transformation** has been used, an additive decomposition is equivalent to using a multiplicative decomposition.

Outline

Outline

Overview of forecasting methods

- Simple forecasting methods
  - Often used as (naive) baselines
- Statistical/time series models
  - Good performance for many real-world applications
  - Often used as (stronger) baselines for AI models
- AI/Machine learning methods
  - Time series forecasting can be reduced to a regression problem
  - Use any AI learning algorithm for regression
  - Specific AI architectures for sequential data

Outline

Challenges in training AI models for forecasting

- (Statistically) dependent data
  - (naive) shuffling can destroy the temporal dependence structure
  - Time-series training/test split needed
- Non-stationarity
  - Validity of the training/test split?
  - Transformations to stabilize mean and variance
- Specific patterns: seasonality, trend, cycle, etc
  - Can the model capture these patterns?
- Arrow of time: use past observations to predict future values
- Multi-step ahead forecasting, i.e. sequential predictions

Outline

Some simple forecasting methods

Time series data: $y_1, y_2, \dots, y_T$
Target: $\hat{y}_{T+h|T}$ (a forecast of y_{T+h} based on $y_1, \dots, y_T$)

- **Naïve method**:

$$\hat{y}_{T+h|T} = y_T.$$

  Because a naïve forecast is optimal when data follow a random walk, these are also called random walk forecasts.

- **Average method**:

$$\hat{y}_{T+h|T} = \bar{y} = (y_1 + \cdots + y_T)/T.$$

Some simple forecasting methods

- **Seasonal naïve method**:

$$\hat{y}_{T+h|T} = y_{T+h-m(k+1)},$$

where m is the seasonal period, and k is the integer part of $(h-1)/m$ (i.e., the number of complete years in the forecast period prior to time $T+h$).

- **Drift method (a variation on the naïve method)**:

$$\hat{y}_{T+h|T} = y_T + \frac{h}{T-1}\sum_{t=2}^{T}(y_t - y_{t-1}) = y_T + h\left(\frac{y_T - y_1}{T-1}\right).$$

We allow the forecasts to increase or decrease over time, where the amount of change over time (called the drift) is set to be the average change seen in the historical data. This is equivalent to drawing a line between the first and last observations, and extrapolating it into the future.

Some simple forecasting methods

- **Forecasting with decomposition (seasonal series)**:
  - Apply time series decomposition
  - Forecast the trend-cycle component
  - Forecast the seasonal component by repeating the last season
  - (Optional) Forecast the remainder term
  - Combine forecasts to obtain forecasts of original data

Outline

Course organization

Introduction to time series forecasting

Concepts and tools for time series data

- Time series patterns
- Time series plots
- The autocorrelation function
- Stationarity
- Transformations
- Time series decomposition

Methods and tools for time series forecasting

- Overview of forecasting methods
- Challenges in training AI models for forecasting
- Some simple forecasting methods
- Residual diagnostics
- Evaluating forecast accuracy

Residual diagnostics

- $\hat{y}_{t|t-1}$ is the **forecast** of y_t based on observations $y_1, \dots, y_{t-1}$.
- We call these **fitted values**', also denoted $\hat{y}_t$.
- **Not true forecasts** since model parameters are often estimated on all data.

What are the **fitted values** for the average method and the drift method?

- **Average method** $\hat{y}_{t|t-1} = (y_1 + \cdots + y_{t-1})/(t-1)$
- **Drift method**: $\hat{y}_{t|t-1} = y_{t-1} + h\left(\frac{y_{t-1} - y_1}{t-2}\right)$

Residual diagnostics

- $\hat{y}_{t|t-1}$ is the **forecast** of y_t based on observations $y_1, \dots, y_{t-1}$.
- We call these **fitted values**', also denoted $\hat{y}_t$.
- **Not true forecasts** since model parameters are often estimated on all data.

What are the **fitted values** for the average method and the drift method?

- **Average method** $\hat{y}_{t|t-1} = (y_1 + \cdots + y_{t-1})/(t-1)$
- **Drift method**: $\hat{y}_{t|t-1} = y_{t-1} + h\left(\frac{y_{t-1}-y_1}{t-2}\right)$

Residual diagnostics

The **residuals** are the difference between observed value and fitted values:

$$e_t = y_t - \hat{y}_{t|t-1}.$$

Two common assumptions:

- $\{e_t\}$ are **uncorrelated**.

  If they aren't, then information left in residuals should be exploited.

  A standard residual diagnostic is to check the ACF of the residuals of a forecasting method. We expect these to look like white noise.

- $\{e_t\}$ have **mean zero**.

  If they don't, then forecasts are biased.

Other useful properties (for distribution forecasts): $\{e_t\}$ have **constant variance** and are **normally distributed**.

Outline

Evaluating forecast accuracy

- A model which fits the time series well will not necessarily forecast well.
- A perfect fit can always be obtained by using a complex model.

  Tradeoff between overfitting and underfitting.
  Split the time series into a **training set** and a **test set**.
- The test set must not be used for any aspect of model development or calculation of forecasts.
- Forecast accuracy is based only on the test set.

Measures of forecast accuracy

Forecast "errors" are given by the difference between an observed value and its forecast:

$$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T}.$$

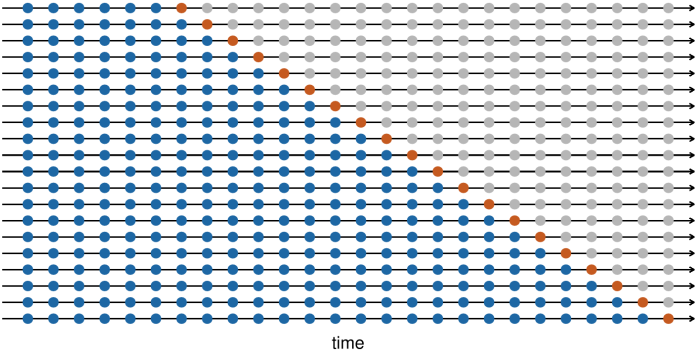
Examples:

- Mean Squared Error (MSE): $\text{Mean}(e^2_{T+h})$
- Mean Absolute Error (MAE): $\text{Mean}(|e_{T+h}|)$
- Root Mean Squared Erros (RMSE): $\sqrt{\text{Mean}(|e_{T+h}|)}$
- Mean Absolute Percentage Error (MAPE): $100 \times \text{Mean}(|e_{T+h}|/|y_{T+h}|)$
- ...

Evaluating forecast accuracy

Hold-out method:

Cross-validation method:



Evaluating forecast accuracy

Cross-validation method (h-step ahead forecasting):