UNIVERSITÉ DE NEUCHÂTEL

Master of Cognitive Science

Data Science Course

Time-Series analysis

Professor: Moran Steven

Lecturer: Maiolini Marco

Lecture 7: 6/April/2022

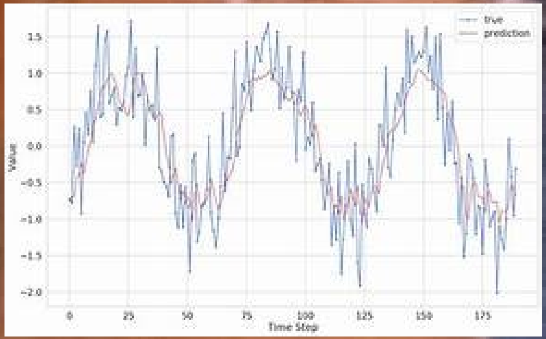
Outline

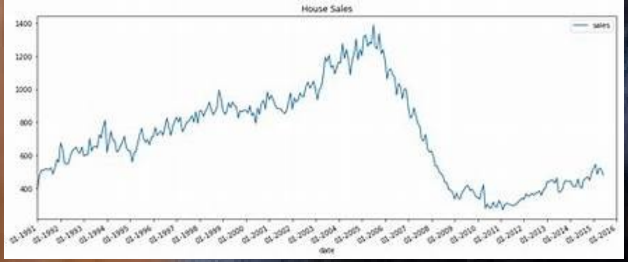
- Part 1: Book report & discussion (15 minutes)
- Part 2: Understand Time-series analysis (45 minutes)

Break (15 minutes)

- Part 3: Practical
- Part 4: Start the report

What is a Time-Series ?



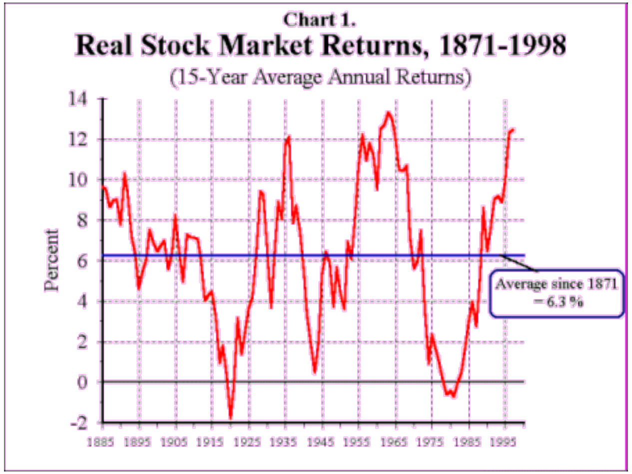


Time-Series

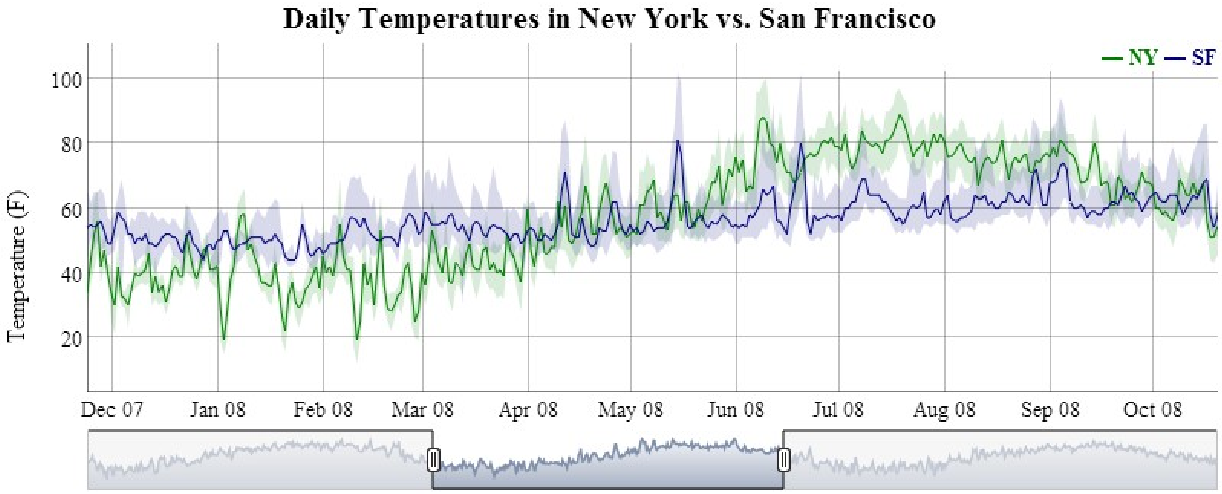
"Collection of observation made sequentially through time"

Chris Chatfield, 2019 "The analysis of time series and introduction with R"

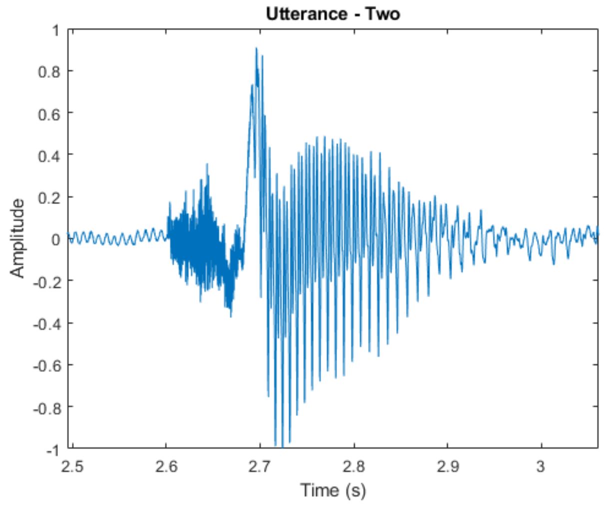
Economic & Finance



Physic

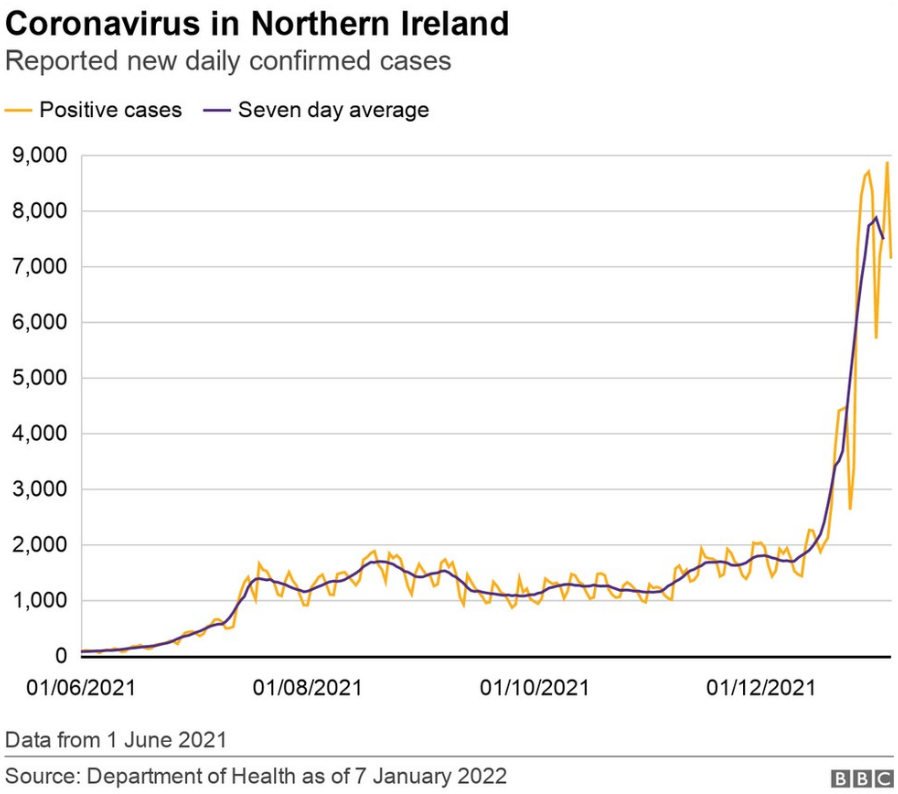


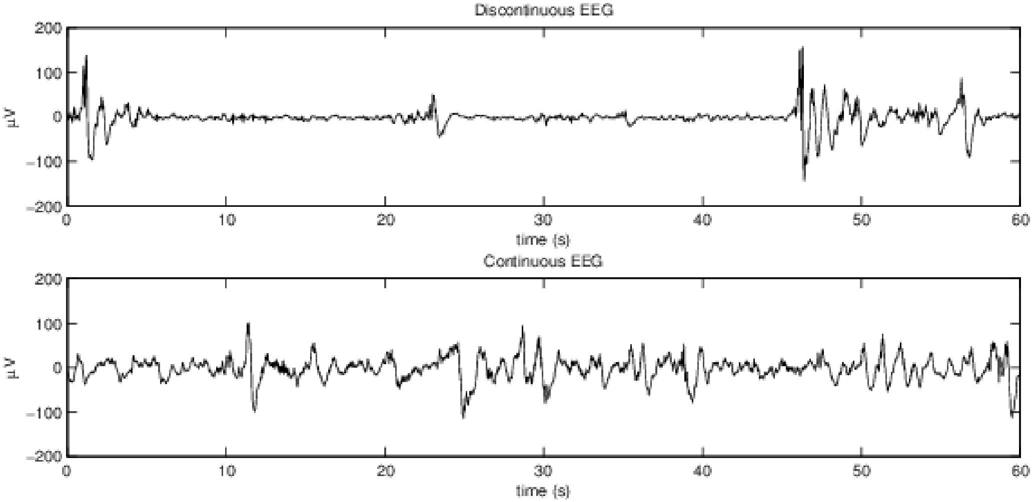
Daily Temperatures in New York vs. San Francisco
NY
SF
Temperature (F)
100
80
60
40
20
Dec 07
Jan 08
Feb 08
Mar 08
Apr 08
May 08
Jun 08
Jul 08
Aug 08
Sep 08
Oct 08



Utterance - Two
Amplitude
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
2.5
2.6
2.7
2.8
2.9
3
Time (s)

Biological


Coronavirus in Northern Ireland
Reported new daily confirmed cases
Positive cases
Seven day average
9,000
8,000
7,000
6,000
5,000
4,000
3,000
2,000
1,000
0
01/06/2021
01/08/2021
01/10/2021
01/12/2021
Data from 1 June 2021
Source: Department of Health as of 7 January 2022
BBC


Discontinuous EEG
200
100
0
−100
−200
μV
0
10
20
30
40
50
60
time (s)
Continuous EEG
200
100
0
−100
−200
μV
0
10
20
30
40
50
60
time (s)

Types of time-series

- *Discrete:* Data taken only in specific fixed time
- *Continuous:* Data taken continuously through time
- *Deterministic:* Predicted exactly by previous values
- *Stochastic:* Only partially predicted by past values

What are we use to?

Describe

What are we use to?

Describe

Explain

What are we use to?

Describe

Explain

Predict

What are we use to?

Describe

Explain

Predict

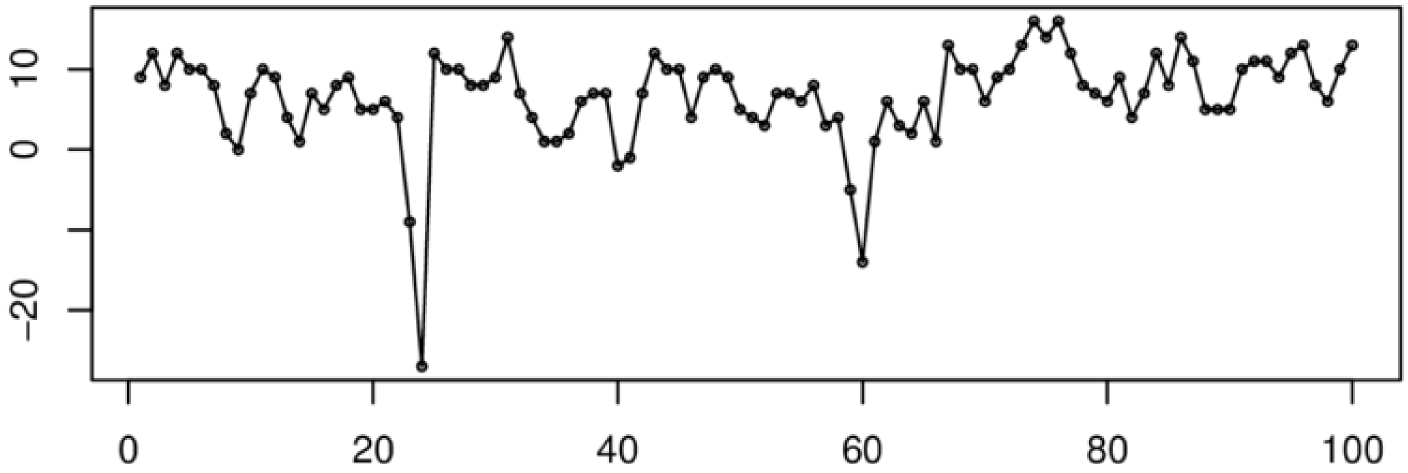
Control

Describe a time-series

- *Mean:* The central value of a finite set of numbers
- *Frequency:* Sampling frequency
- *Variance:* Expectation of the squared deviation of a random variable from its mean
- *Autocorrelation:* Correlation among neighbouring observations

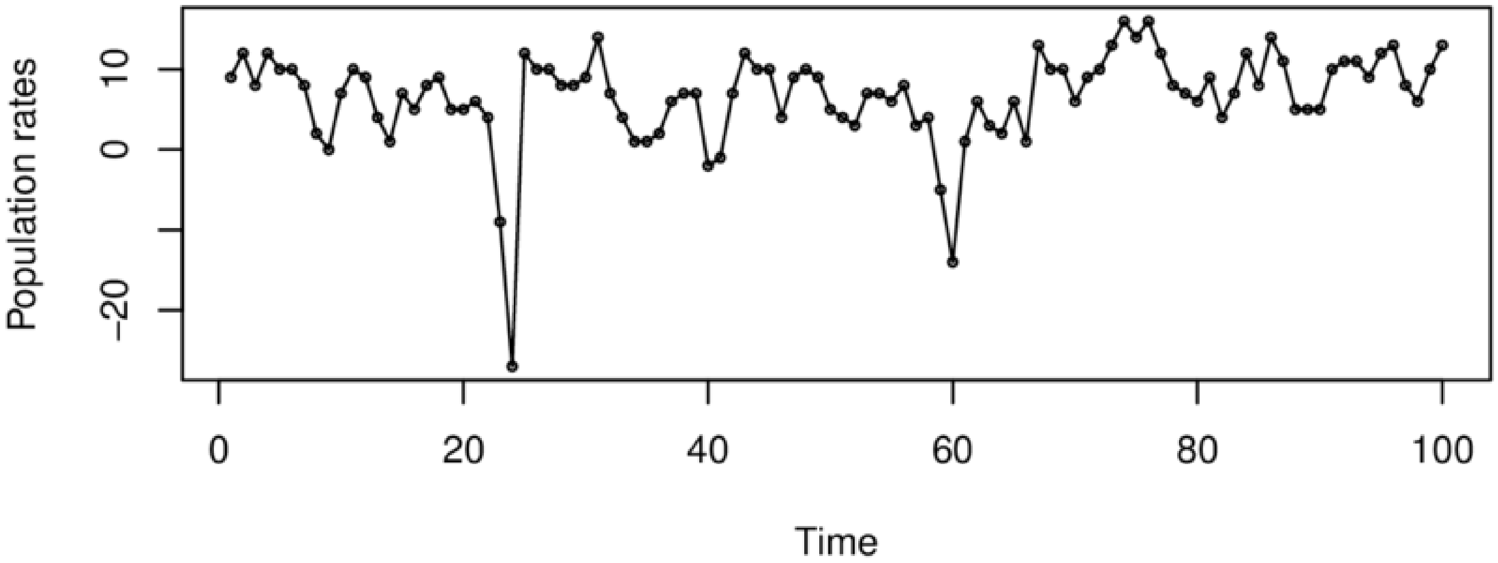
Describe a time-series

Population growth rate:



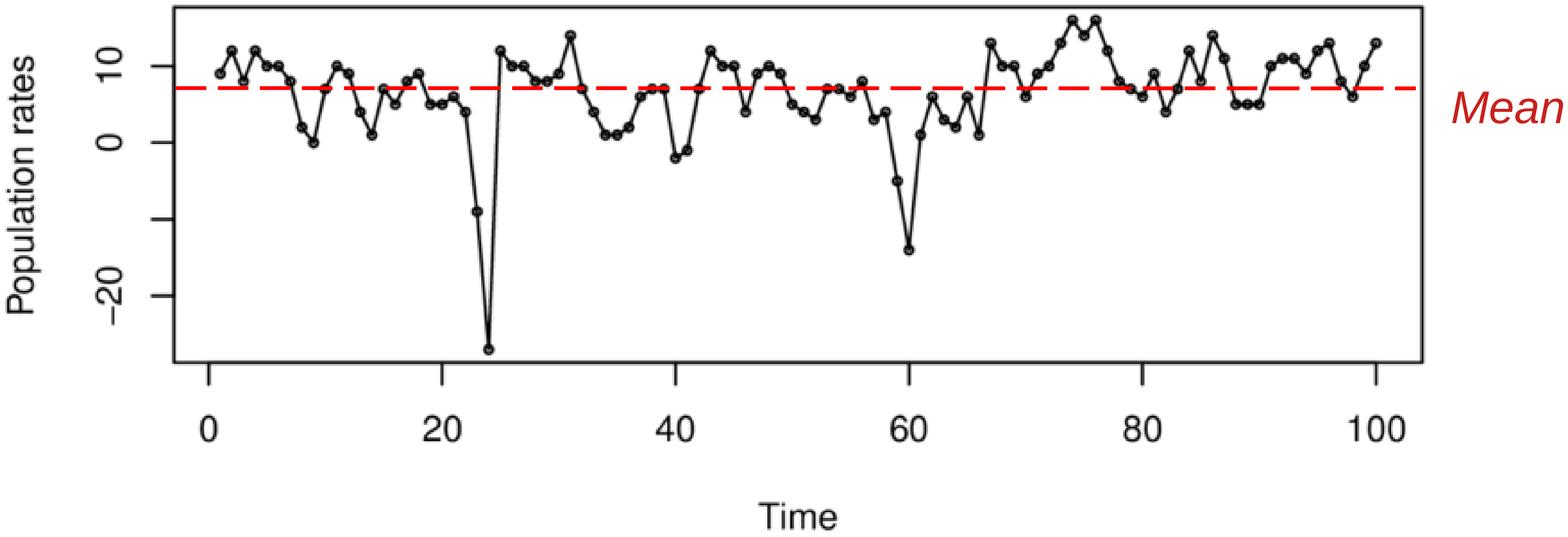
Describe a time-series

Population growth rate:



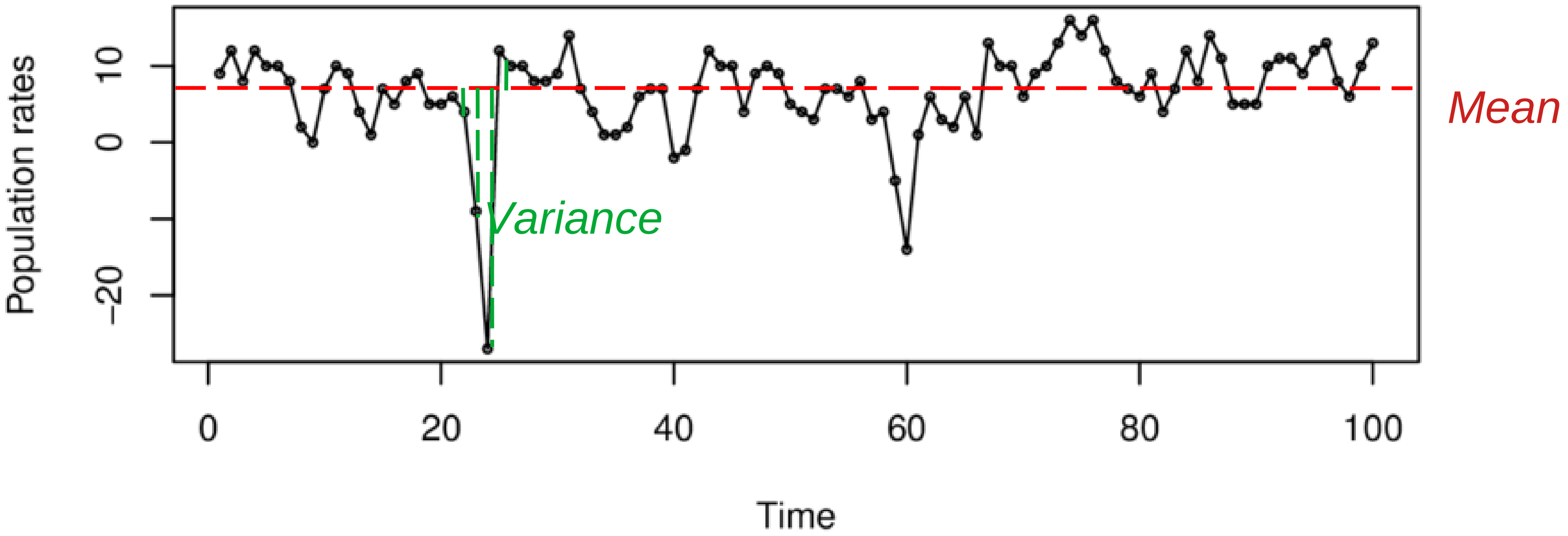
Describe a time-series

Population growth rate:



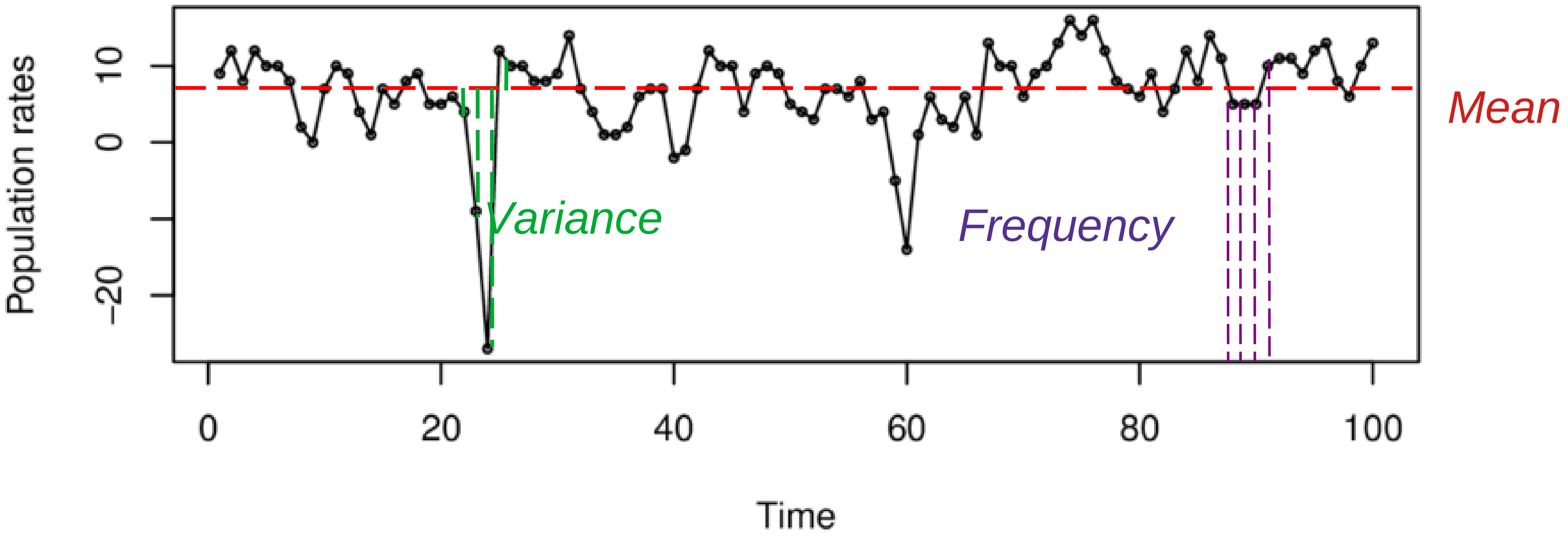
Describe a time-series

Population growth rate:



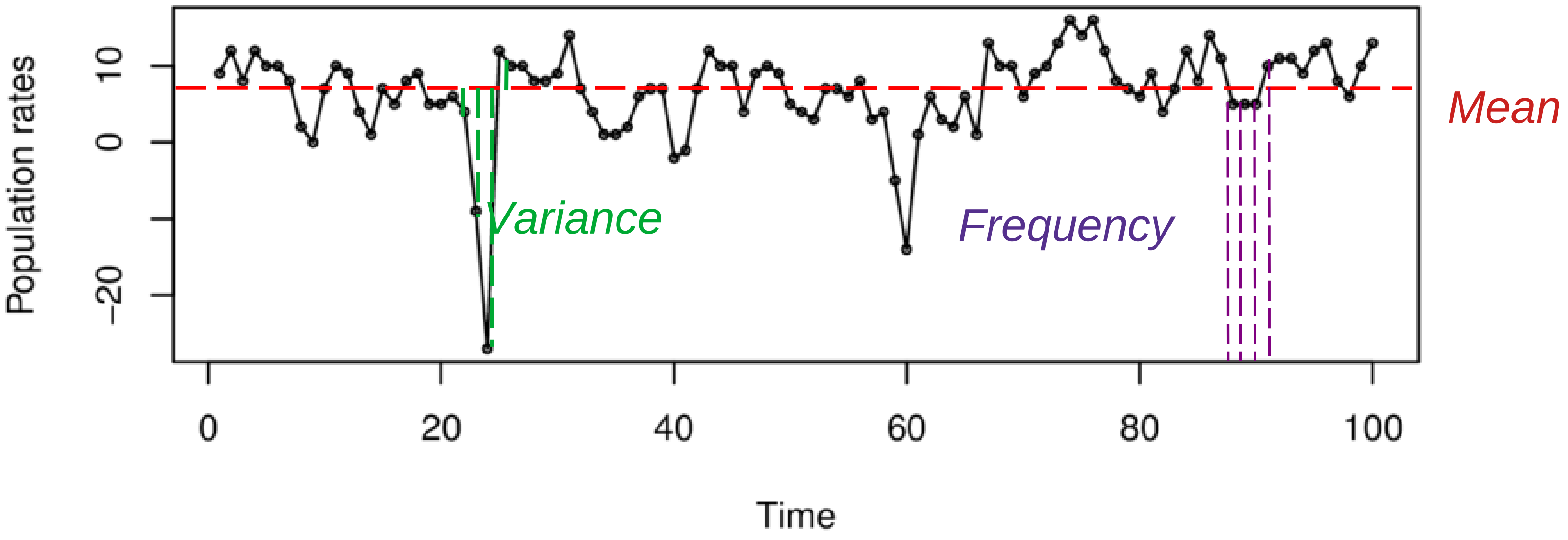
Describe a time-series

Population growth rate:



Describe a time-series

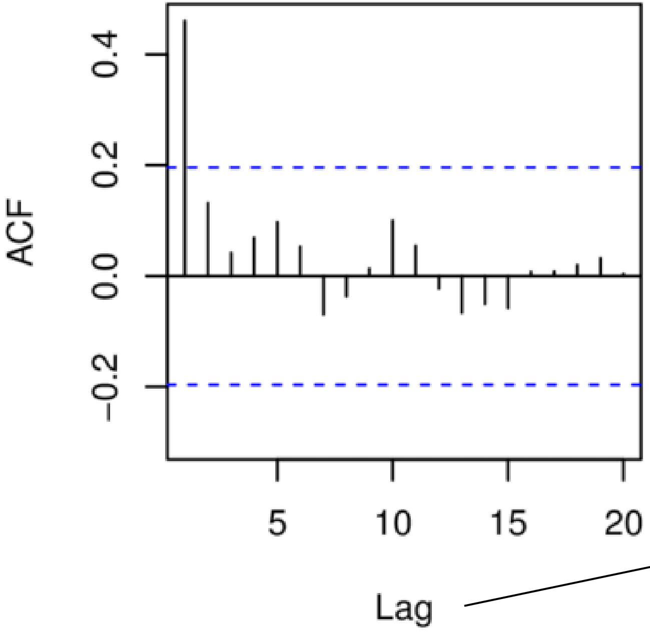
Population growth rate:

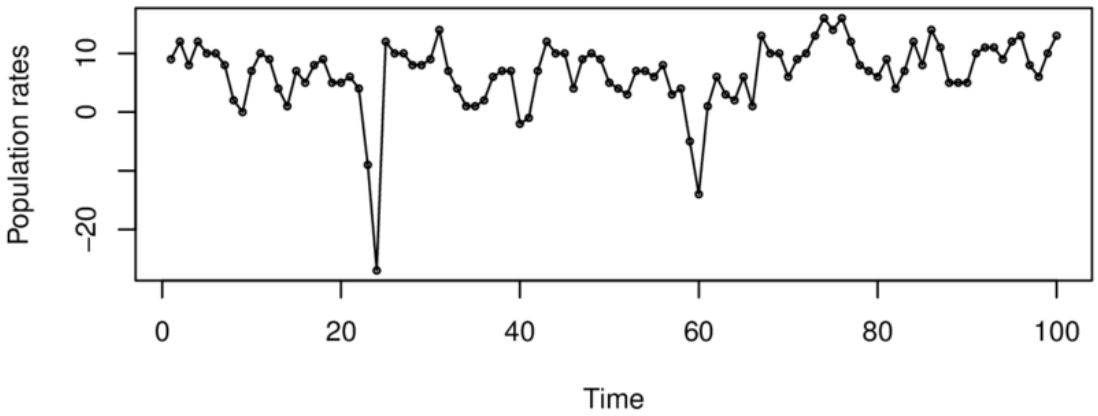


How to plot a time-series in R: *plot.ts(x, xlab, ylab, main)*

Describe a time-series

Autocorrelation plot



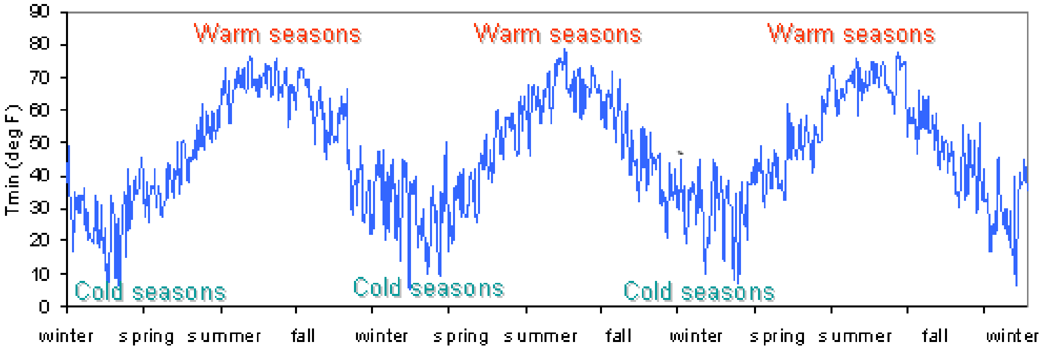


Lag: Delay of the observation is taken

How to do an autocorrelation plot for a specific time-series in R: acf*(x, main)*

Describe a time-series

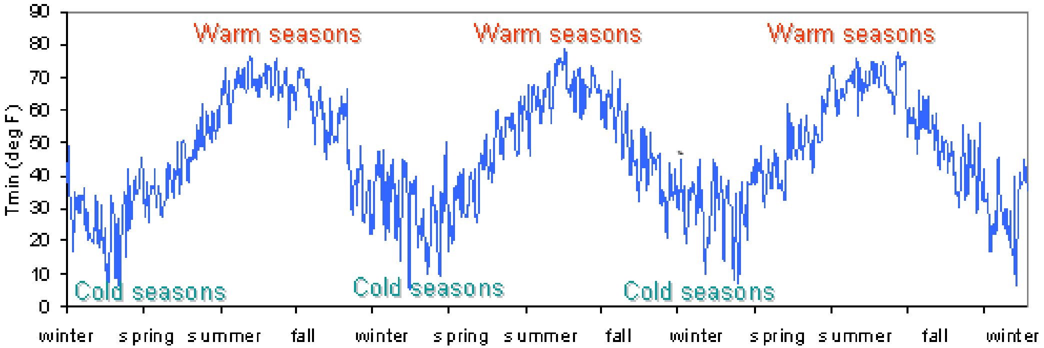
Trend: General development of a feature over time



Describe a time-series

Trend: General development of a feature over time

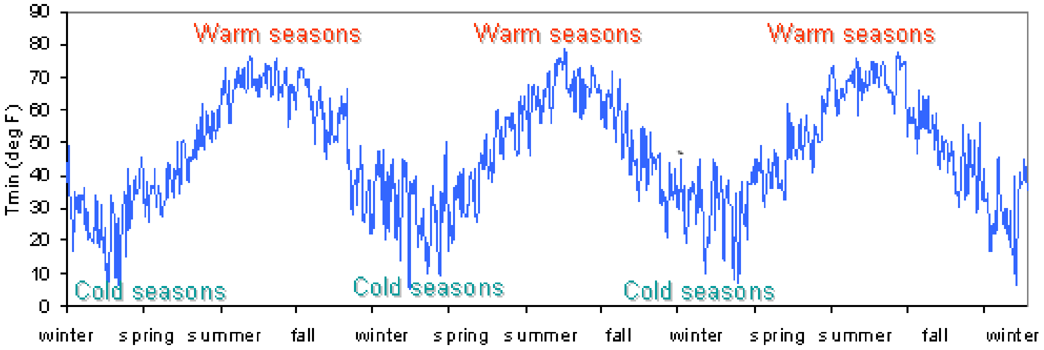
Relation between your data and time



Describe a time-series

Trend: General development of a feature over time

Relation between your data and time



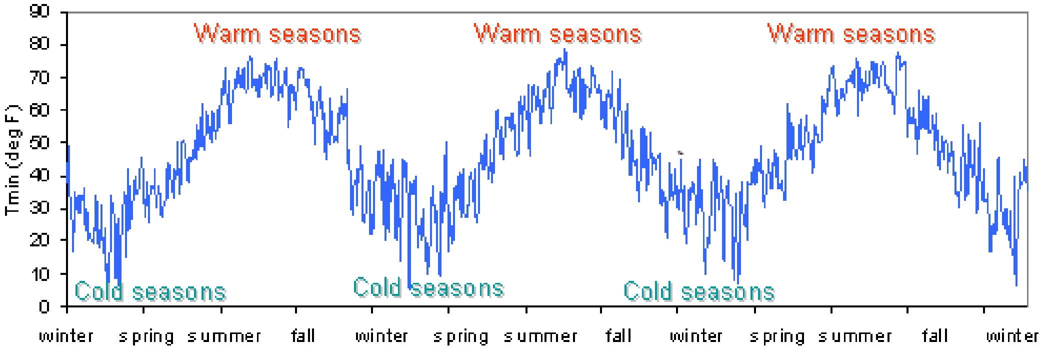
Linear trends

Periodic trends

Describe a time-series

Trend: General development of a feature over time

Relation between your data and time



Linear trends

Periodic trends

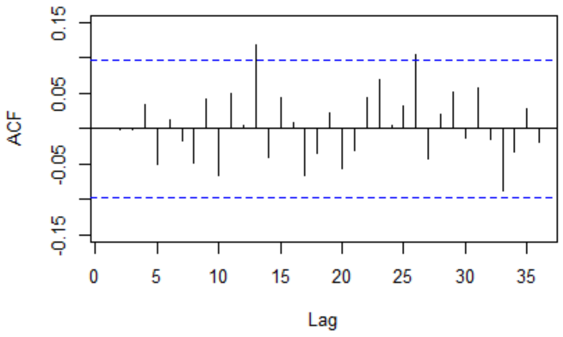
In R you can remove the linear trend with: *diff()*

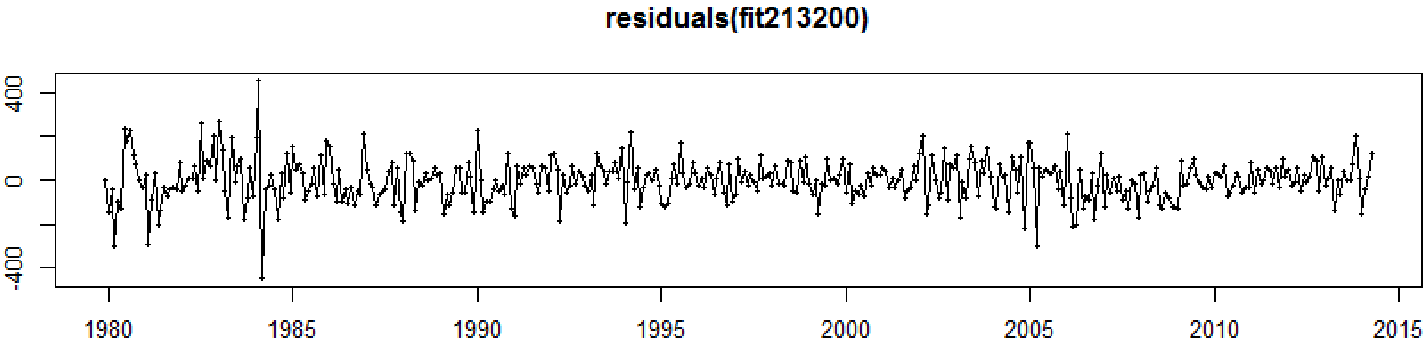
Models of time-series

- White Noise model (WN)
- Random walk model (RW)
- Simple moving average model (MA)
- Autoregressive model (AR)

White Noise model (WN)

The simplest model of time-series

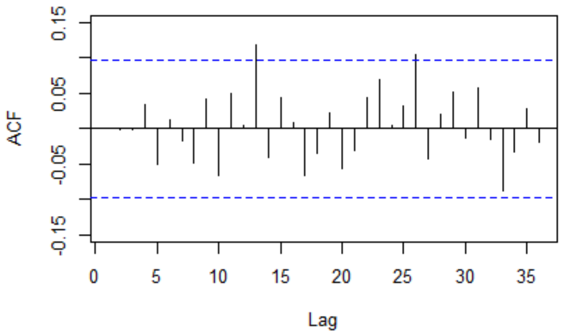


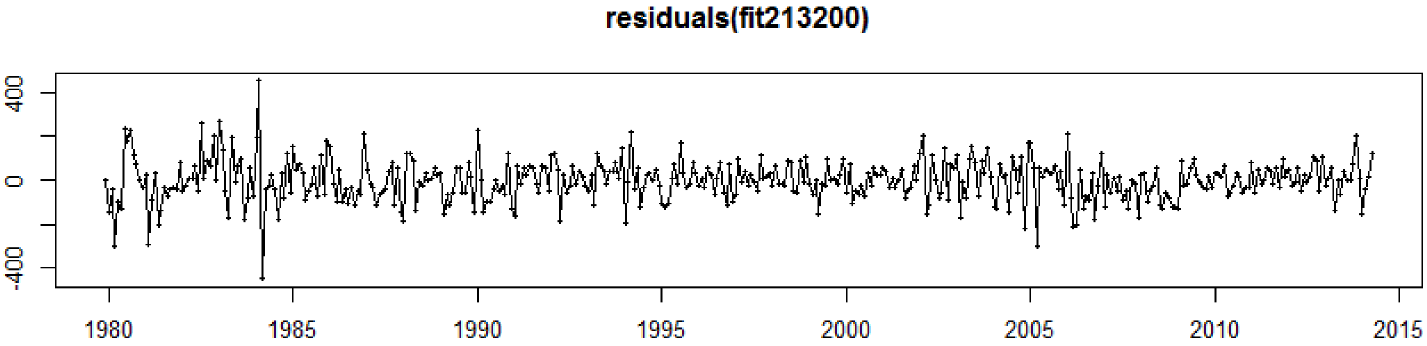


White Noise model (WN)

The simplest model of time-series

- A fixed, constant mean

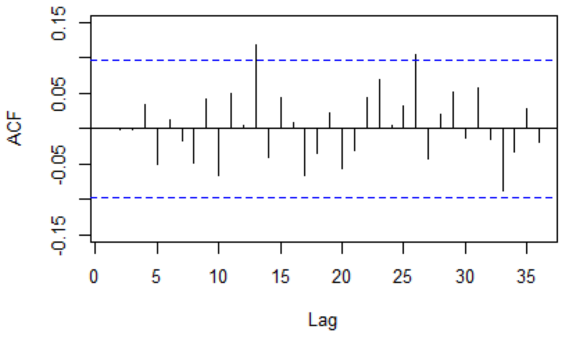


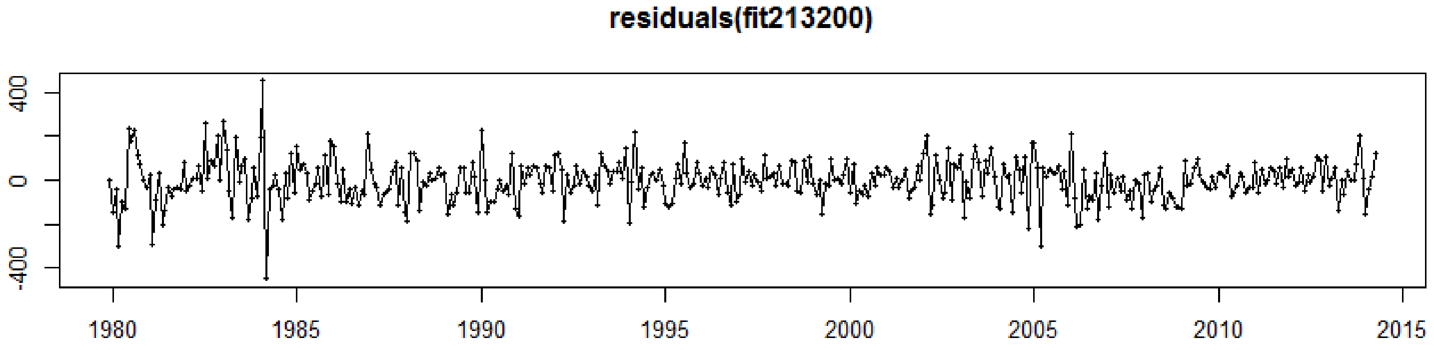


White Noise model (WN)

The simplest model of time-series

- A fixed, constant mean
- A fixed, constant variance

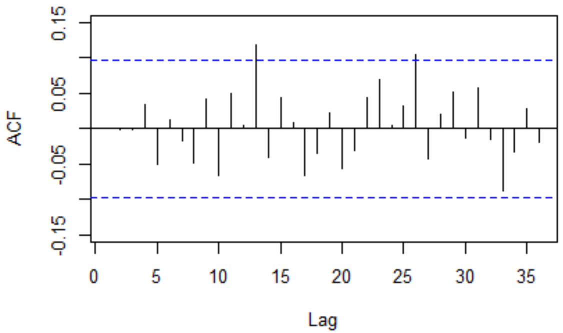


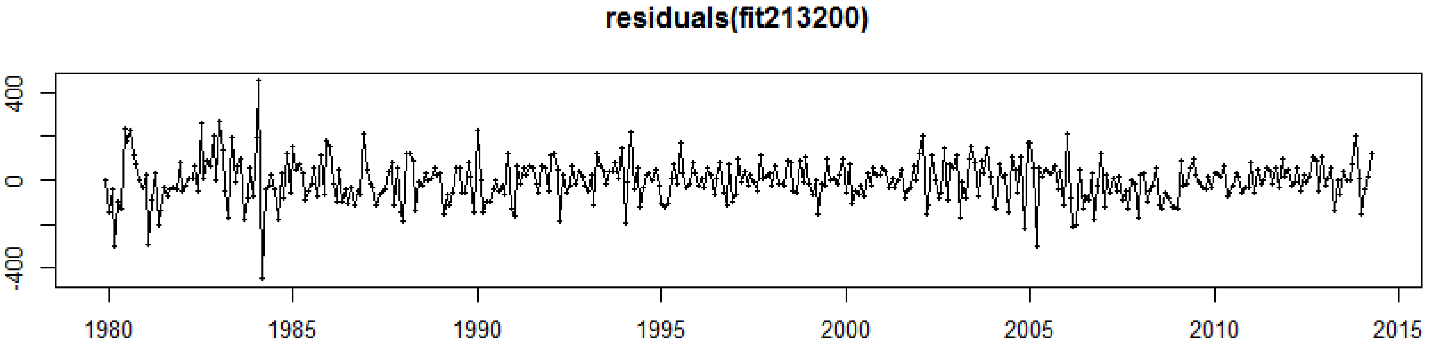


White Noise model (WN)

The simplest model of time-series

- A fixed, constant mean
- A fixed, constant variance
- No correlation over time





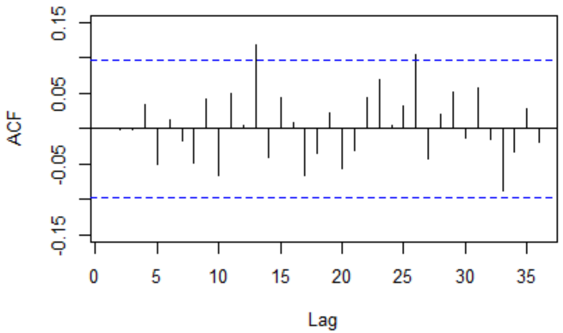
White Noise model (WN)

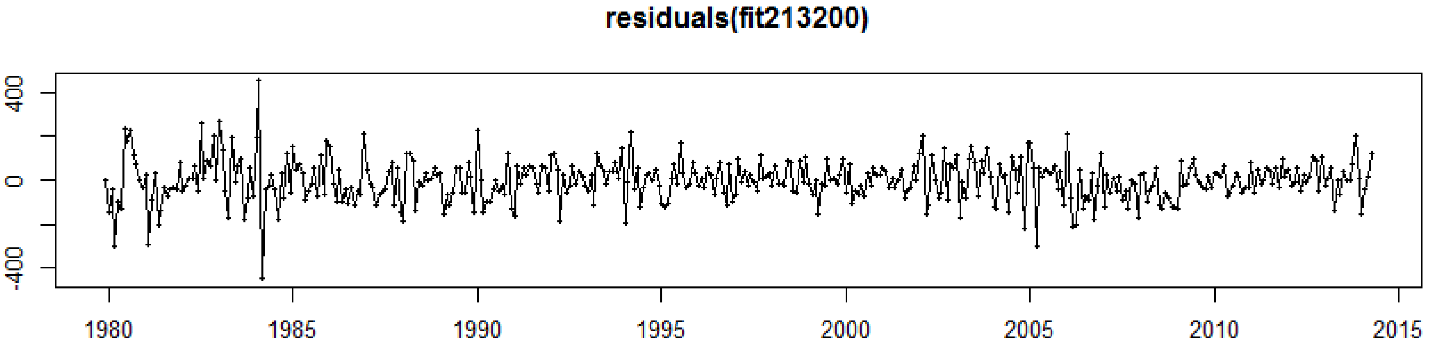
The simplest model of time-series

- A fixed, constant mean
- A fixed, constant variance
- No correlation over time

In R you can fit your data in a WN model using this code:

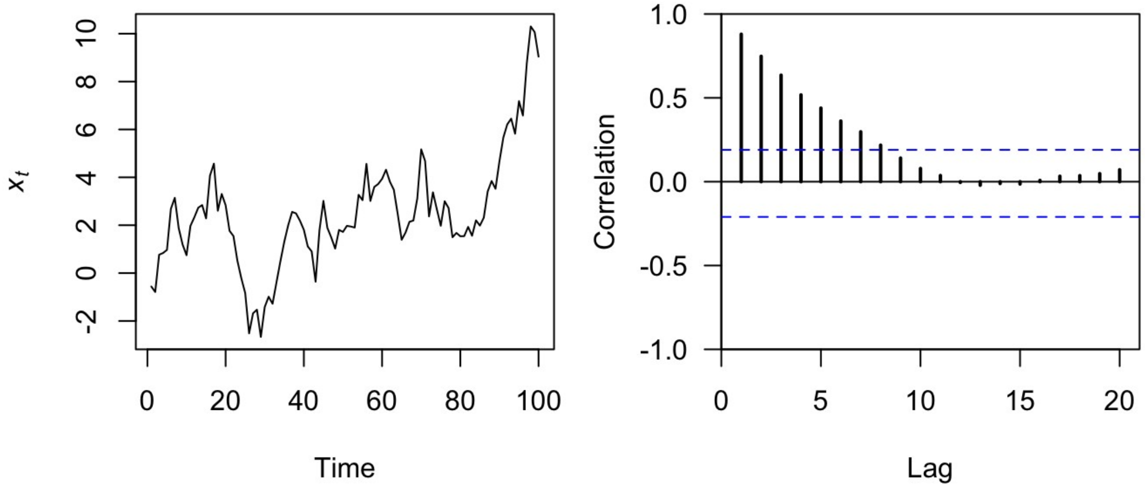
arima(x, order=c(0,0,0))





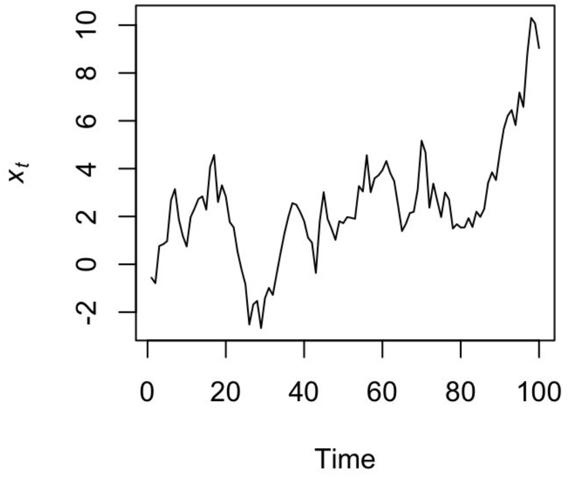
Random walk model (RW)

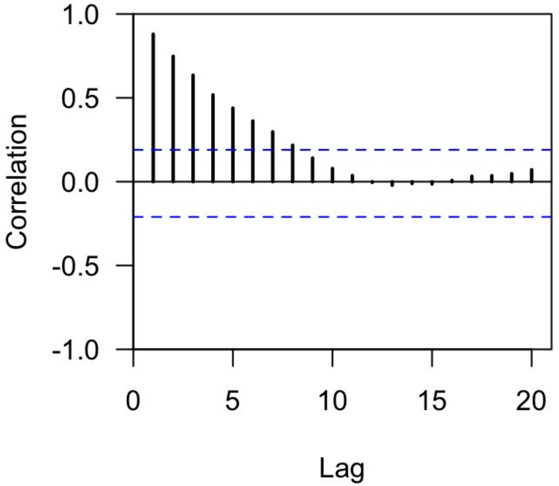
Defined as: Today = Yesterday + Noise



Random walk model (RW)

Defined as: Today = Yesterday + Noise → Today - Yesterday = Noise



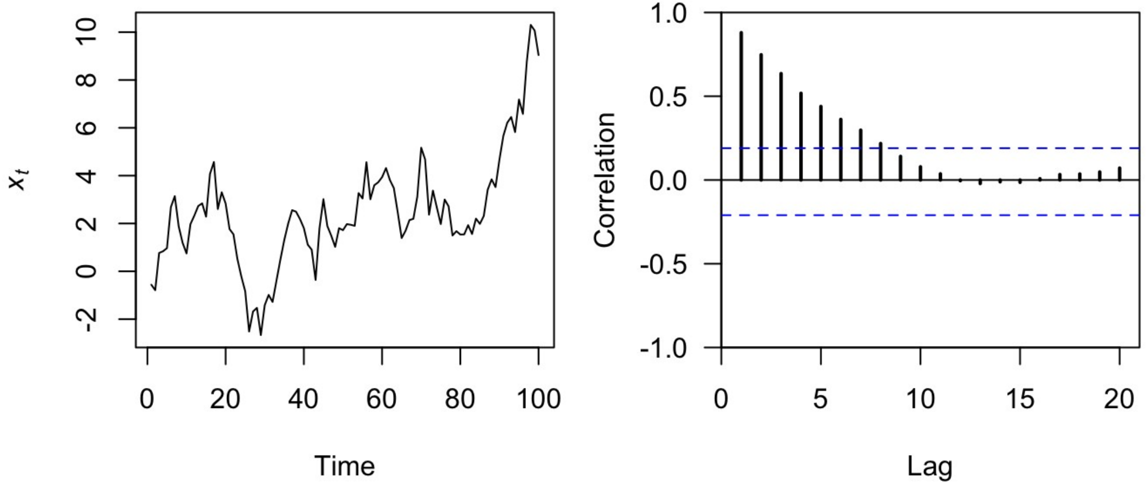


Random walk model (RW)

Defined as: Today = Yesterday + Noise $\longrightarrow$ Today - Yesterday = Noise

White noise

$\boxed{\textit{diff(RW)} \rightarrow \textit{WN model}}$



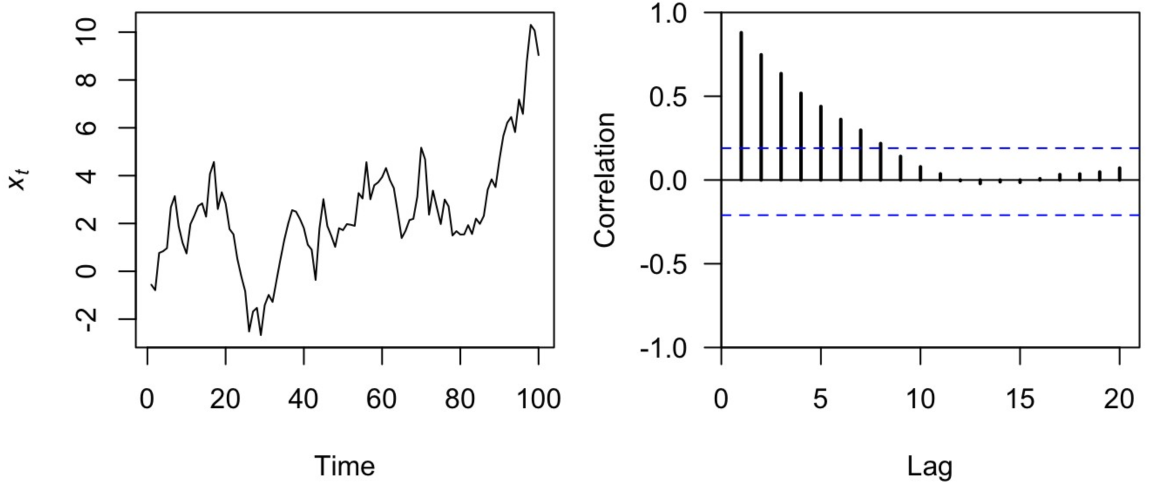
Random walk model (RW)

Defined as: Today = Yesterday + Noise $\longrightarrow$ Today - Yesterday = Noise

White noise

- No specific mean
- No specific variance
- Strong dependence over time

$\boxed{\textit{diff(RW)} \rightarrow \textit{WN model}}$



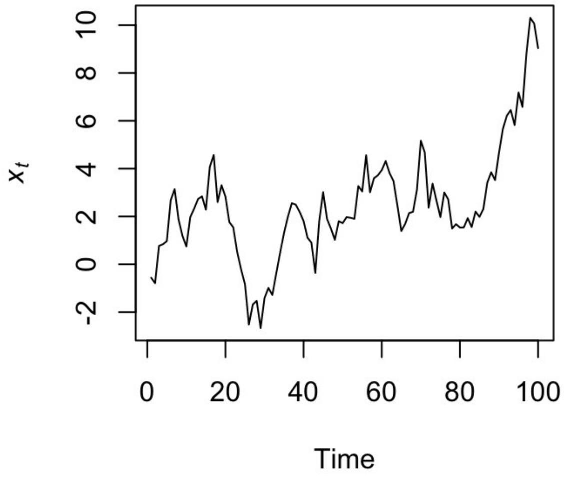
Random walk model (RW)

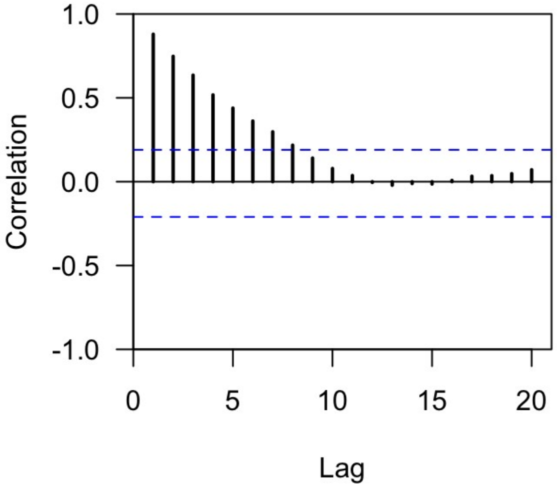
Defined as: Today = Yesterday + Noise → Today - Yesterday = Noise

White noise

- No specific mean
- No specific variance
- Strong dependence over time

diff(RW) → WN model





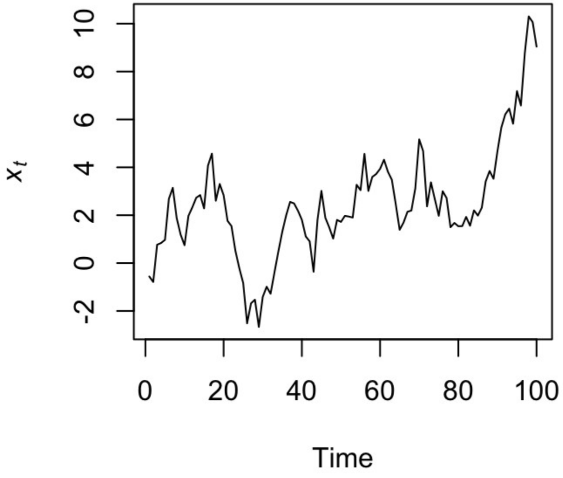
In R you can fit your data in a RW model using this code:

arima(x, order=c(0,1,0))

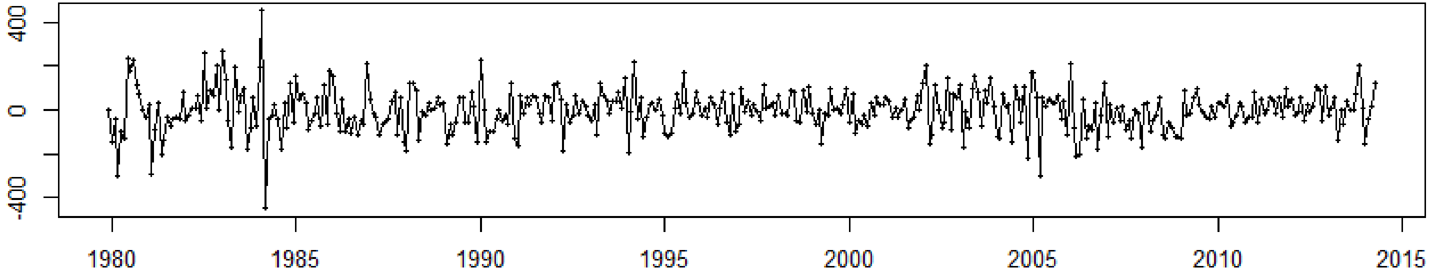
Stationarity

Parsimonious models with distributional stability over time

Random walk model



White Noise model



<u>Stationary models can be modelled with fewer values, however few time series are stationary</u>

Autoregressive models (AR)

Linear trend where each observation is regressed on the previous observation

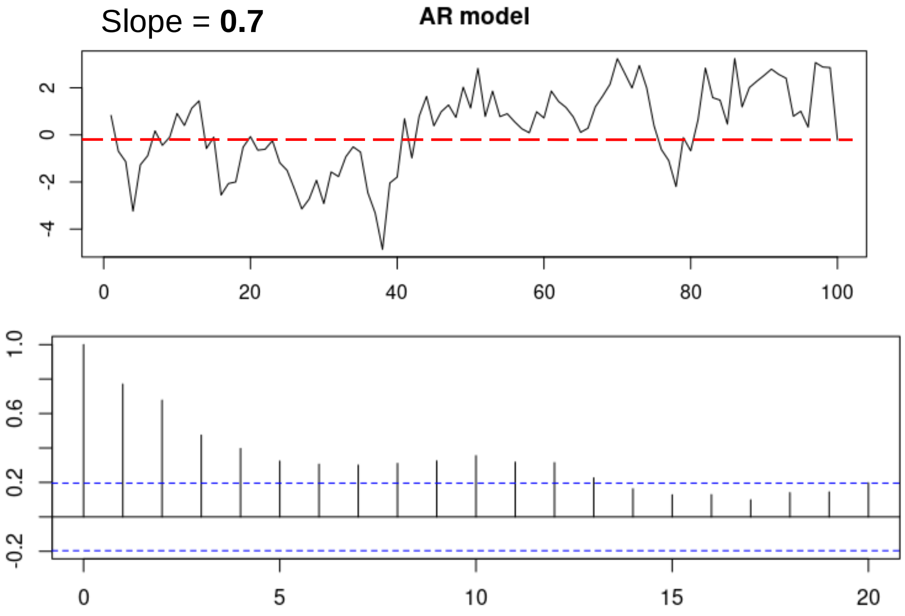
(Today - Mean) = Slope * (Yesterday - Mean) + Noise

Autoregressive models (AR)

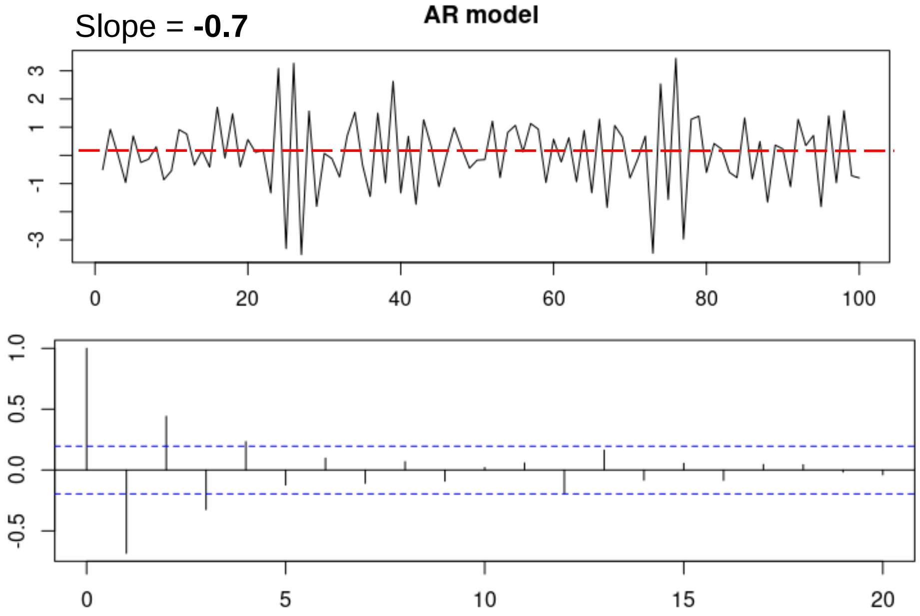
Linear trend where each observation is regressed on the previous observation

(Today - Mean) = Slope * (Yesterday - Mean) + Noise

Slope = **0.7**



Slope = **-0.7**



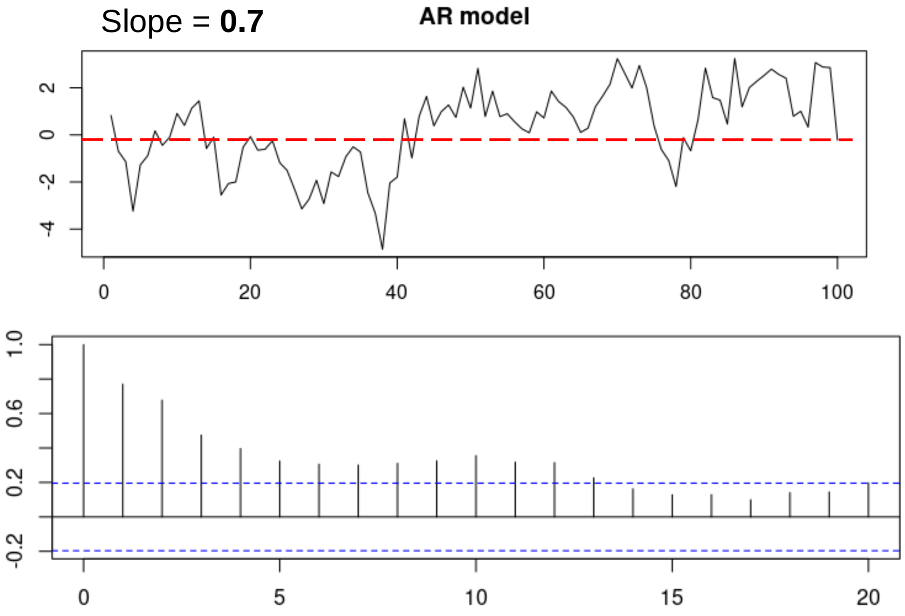
Autoregressive models (AR)

Linear trend where each observation is regressed on the previous observation

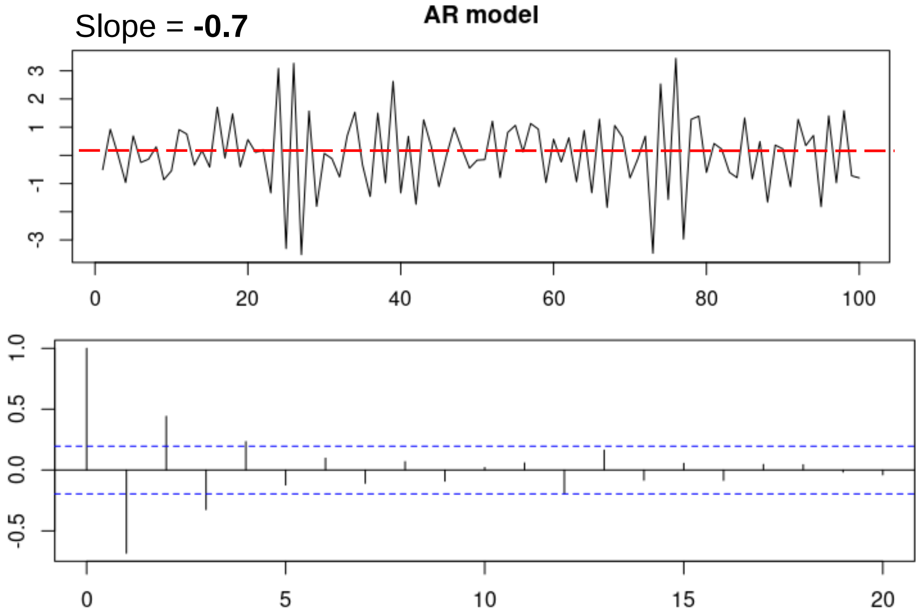
Slope = 1 & Mean = 0

(Today - Mean) = Slope * (Yesterday - Mean) + Noise

Slope = **0.7**



Slope = **-0.7**



Autoregressive models (AR)

Linear trend where each observation is regressed on the previous observation

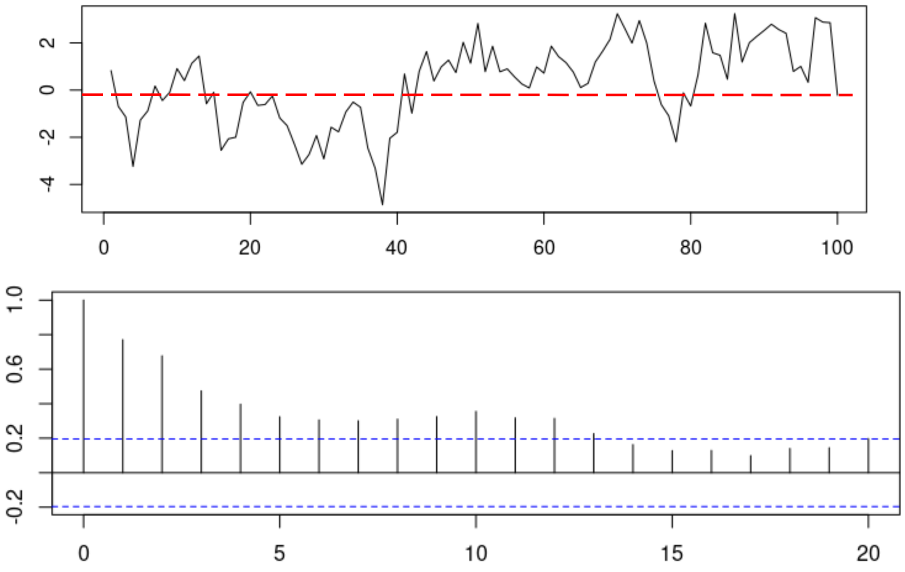
Slope = 1 & Mean = 0

(Today - 0) = 1 * (Yesterday - 0) + Noise

(Today - Mean) = Slope * (Yesterday - Mean) + Noise

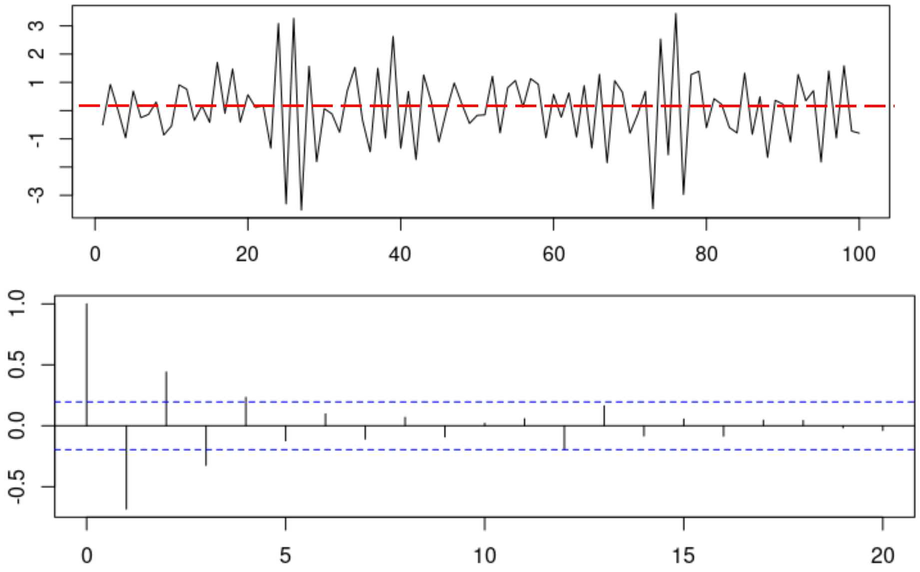
Slope = **0.7**

AR model



Slope = **-0.7**

AR model



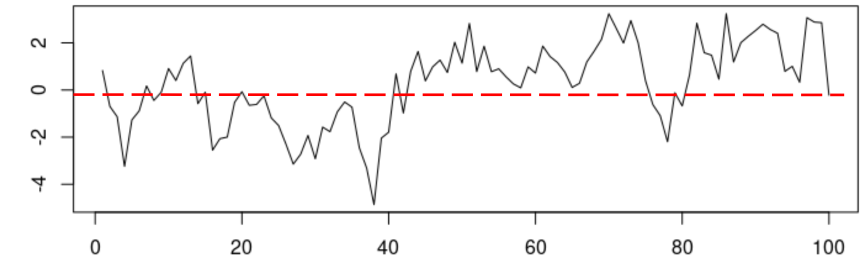
Autoregressive models (AR)

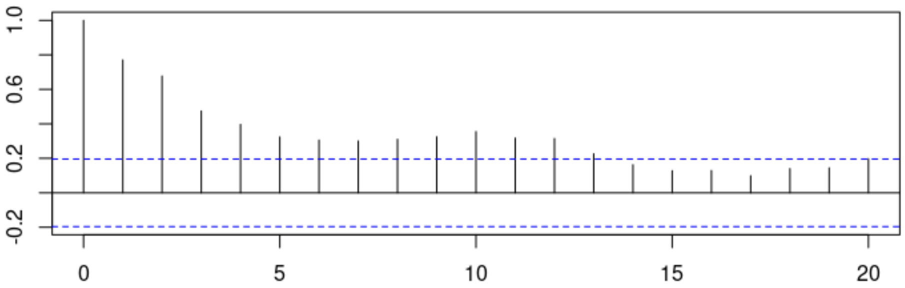
Linear trend where each observation is regressed on the previous observation

Slope = 1 & Mean = 0 Today = Yesterday + Noise

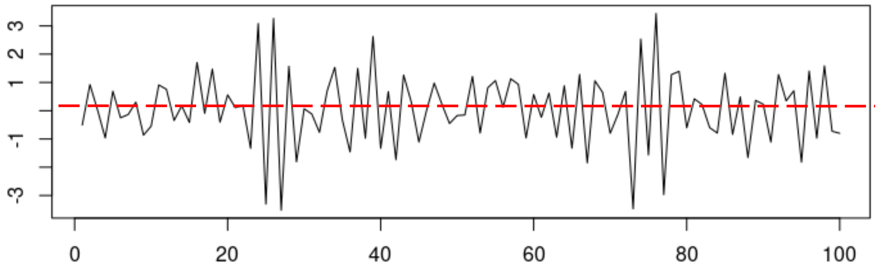
(Today - Mean) = Slope * (Yesterday - Mean) + Noise

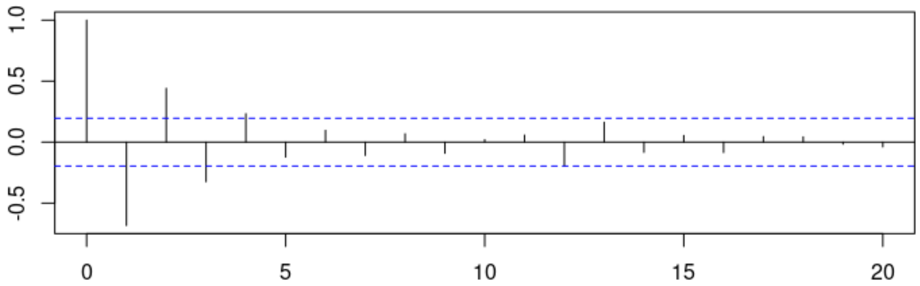
Slope = **0.7** **AR model**





Slope = **-0.7** **AR model**





Autoregressive models (AR)

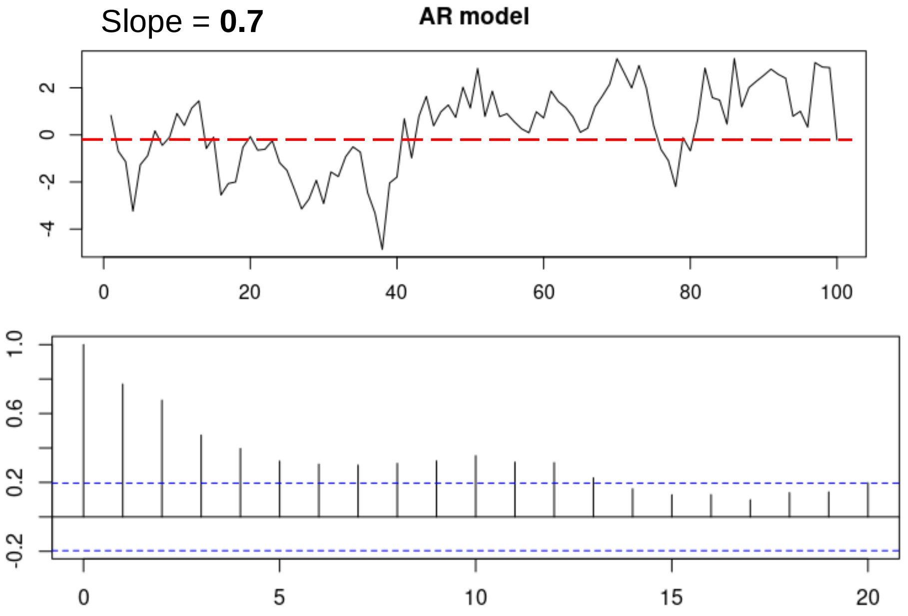
Linear trend where each observation is regressed on the previous observation

Slope = 1 & Mean = 0 — Today = Yesterday + Noise

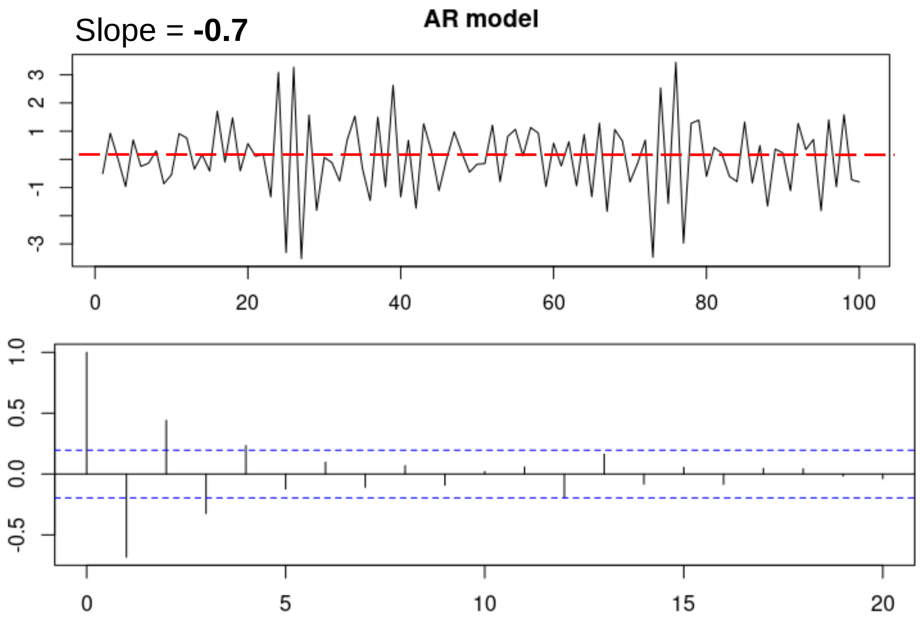
(Today - Mean) = Slope * (Yesterday - Mean) + Noise

Slope = 0

Slope = **0.7**



Slope = **-0.7**



Autoregressive models (AR)

Linear trend where each observation is regressed on the previous observation

Slope = 1 & Mean = 0

Today = Yesterday + Noise

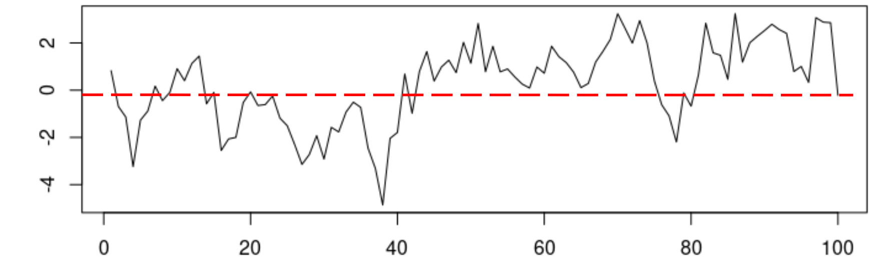
(Today - Mean) = Slope * (Yesterday - Mean) + Noise

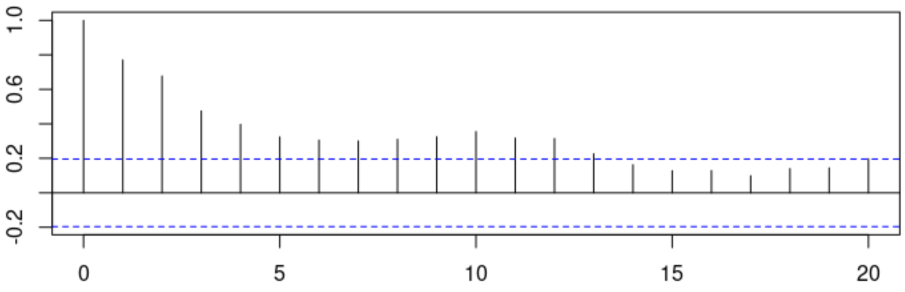
(Today - Mean) = 0 * (Yesterday - Mean) + Noise

Slope = 0

Slope = **0.7**

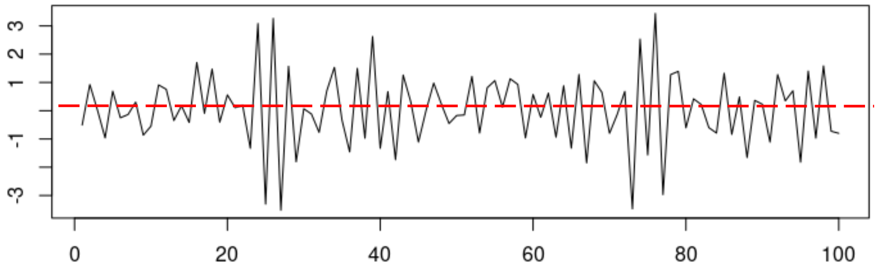
AR model

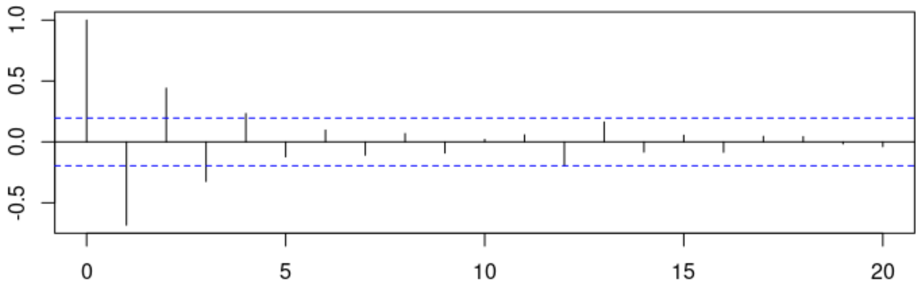




Slope = **-0.7**

AR model





Autoregressive models (AR)

Linear trend where each observation is regressed on the previous observation

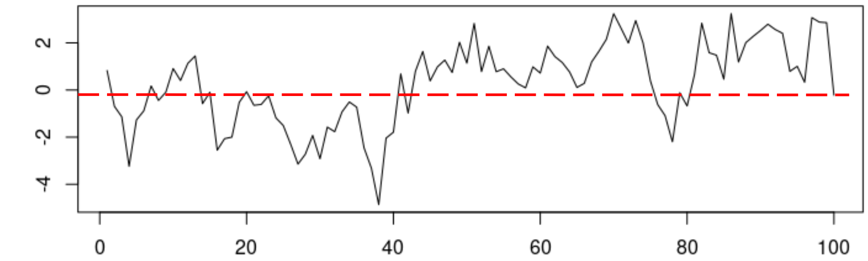
Slope = 1 & Mean = 0 → Today = Yesterday + Noise

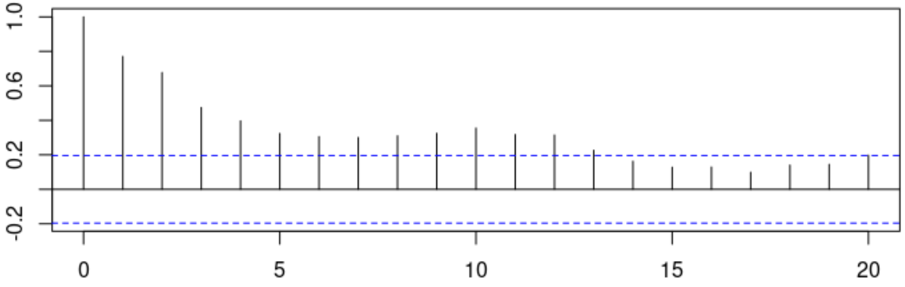
(Today - Mean) = Slope * (Yesterday - Mean) + Noise

Slope = 0 → (Today - Mean) = Noise

Slope = **0.7**

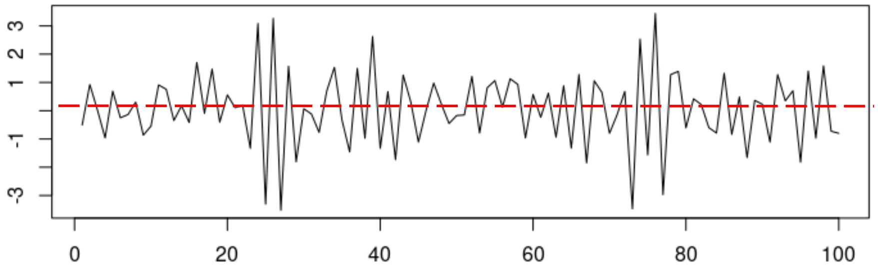
AR model

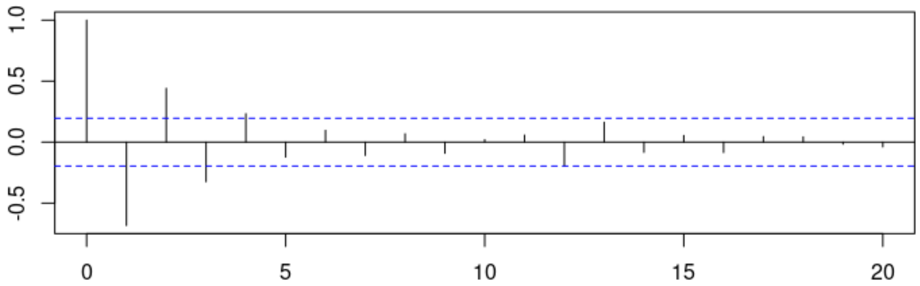




Slope = **-0.7**

AR model





Simple Moving Average models (MA)

Linear trend where each observation is regressed on the previous innovation, which is not actually observed

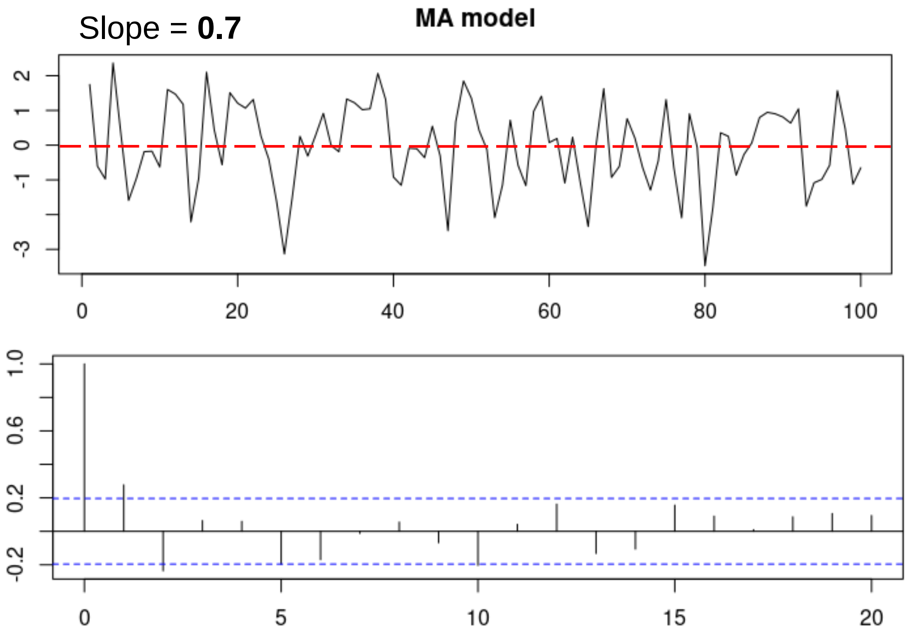
Today = Mean + Noise + Slope * (Yesterday's Noise)

Simple Moving Average models (MA)

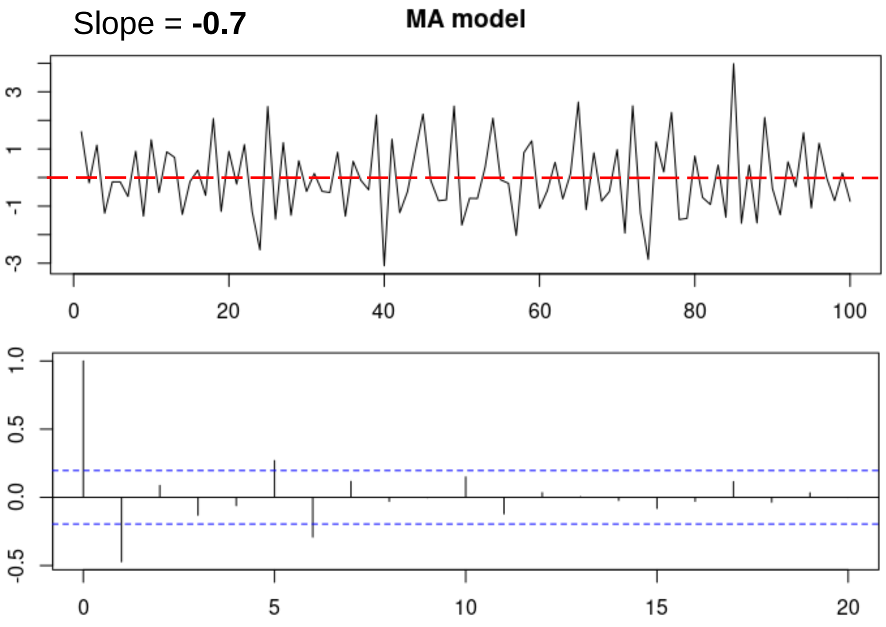
Linear trend where each observation is regressed on the previous innovation, which is not actually observed

Today = Mean + Noise + Slope * (Yesterday’s Noise)

Slope = **0.7**



Slope = **-0.7**



Simple Moving Average models (MA)

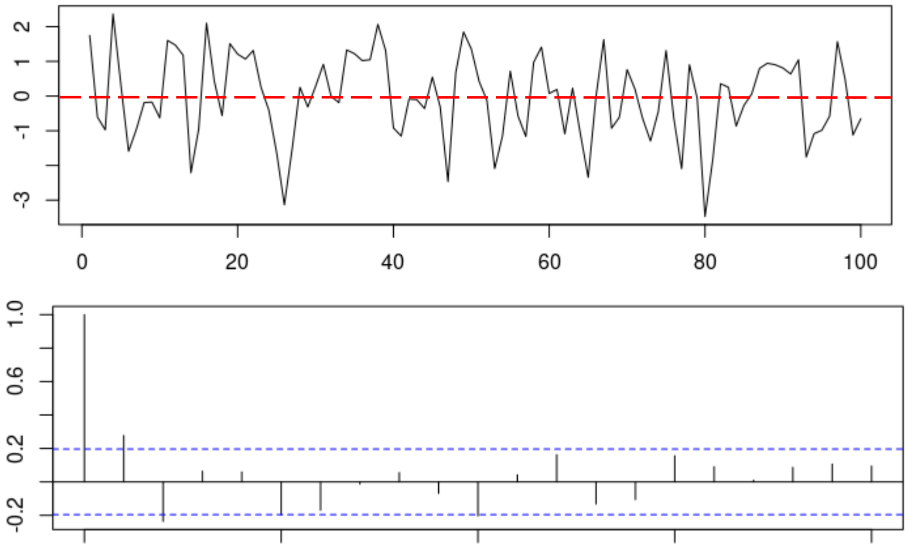
Linear trend where each observation is regressed on the previous innovation, which is not actually observed

Slope = 0

Today = Mean + Noise + Slope * (Yesterday’s Noise)

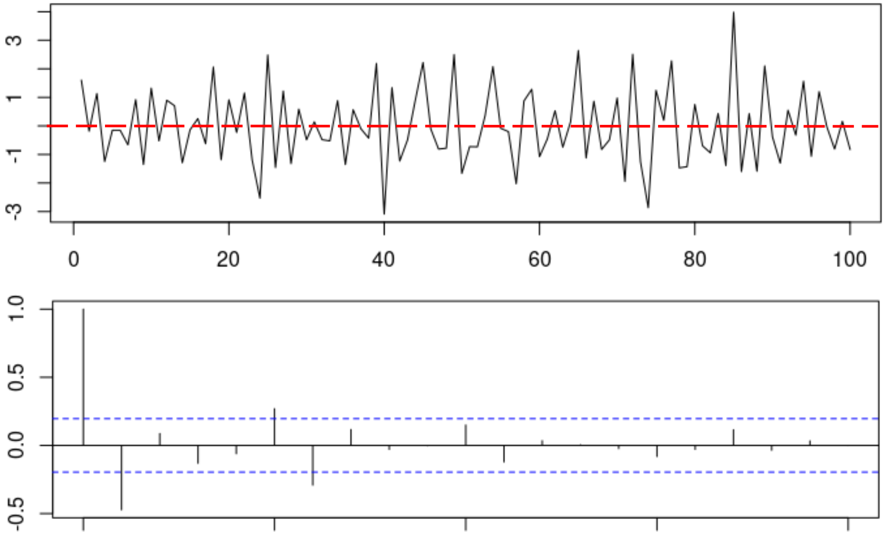
Slope = **0.7**

MA model



Slope = **-0.7**

MA model



Simple Moving Average models (MA)

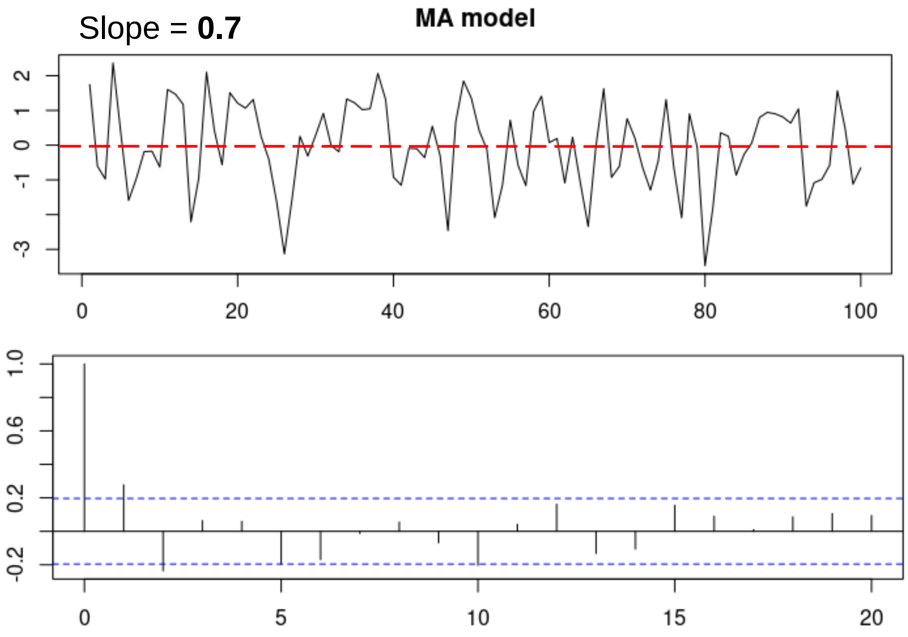
Linear trend where each observation is regressed on the previous innovation, which is not actually observed

Slope = 0

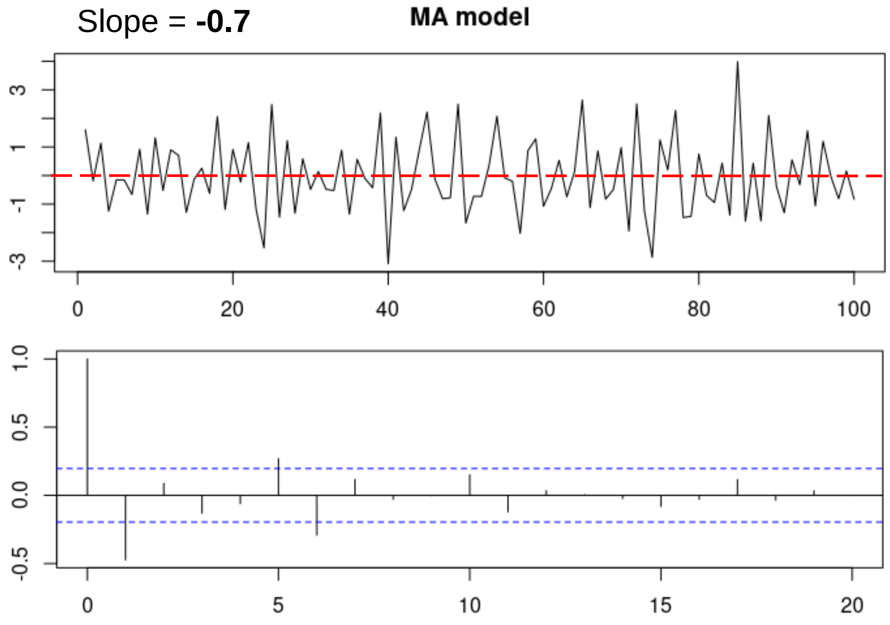
Today = Mean + Noise + 0 * (Yesterday's Noise)

Today = Mean + Noise + Slope * (Yesterday's Noise)

Slope = **0.7**



Slope = **-0.7**



Simple Moving Average models (MA)

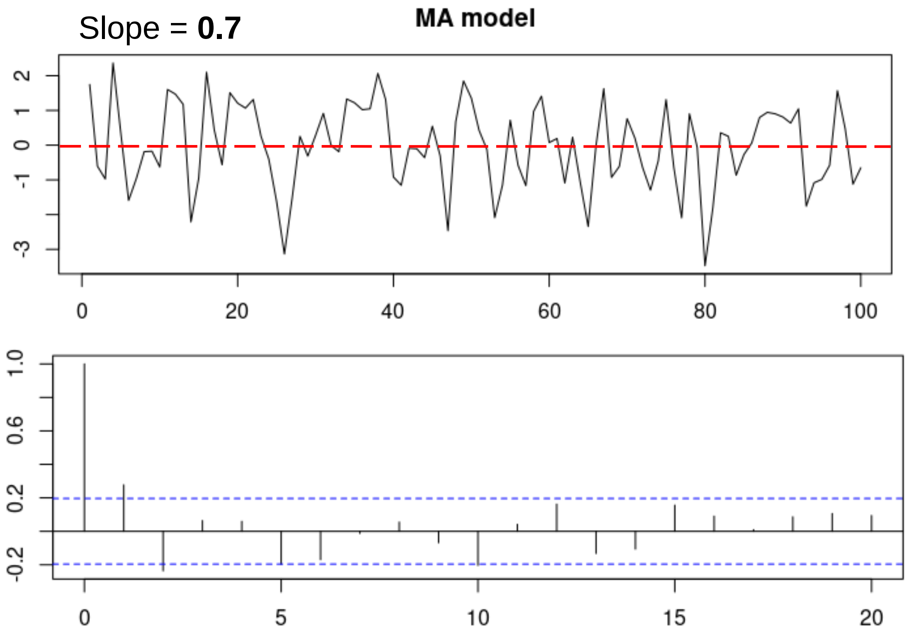
Linear trend where each observation is regressed on the previous innovation, which is not actually observed

Slope = 0

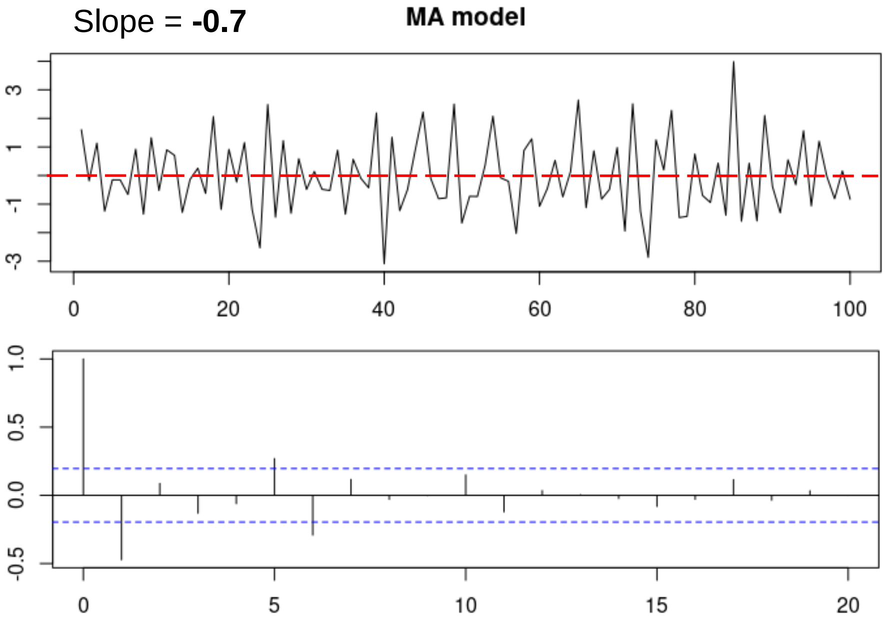
Today = Mean + Noise

Today = Mean + Noise + Slope * (Yesterday’s Noise)

Slope = **0.7**



Slope = **-0.7**



Forecast

We can use the Autoregressive models (AR) and the Simple Moving Average models (MA) models to fit our data and try to forecast our time-series

- *Fitted values:* Forecast (estimation) of an observation using all previous ones

Forecast

We can use the Autoregressive models (AR) and the Simple Moving Average models (MA) models to fit our data and try to forecast our time-series

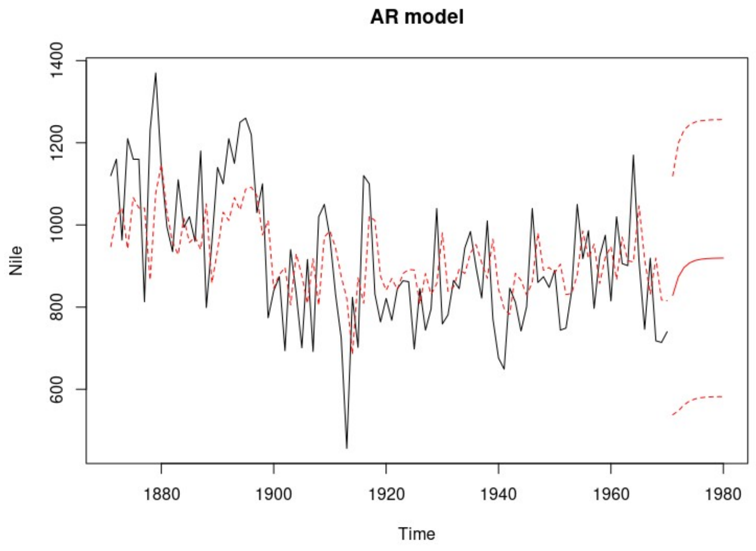
- *Fitted values:* Forecast (estimation) of an observation using all previous ones
- *Residuals:* Difference between the observation and the corresponding fitted values

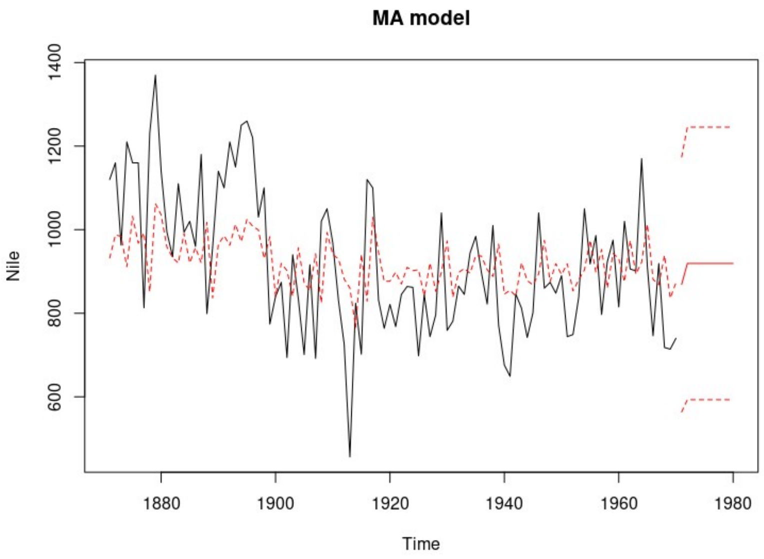
Forecast

We can use the Autoregressive models (AR) and the Simple Moving Average models (MA) models to fit our data and try to forecast our time-series

- *Fitted values:* Forecast (estimation) of an observation using all previous ones

- *Residuals:* Difference between the observation and the corresponding fitted values

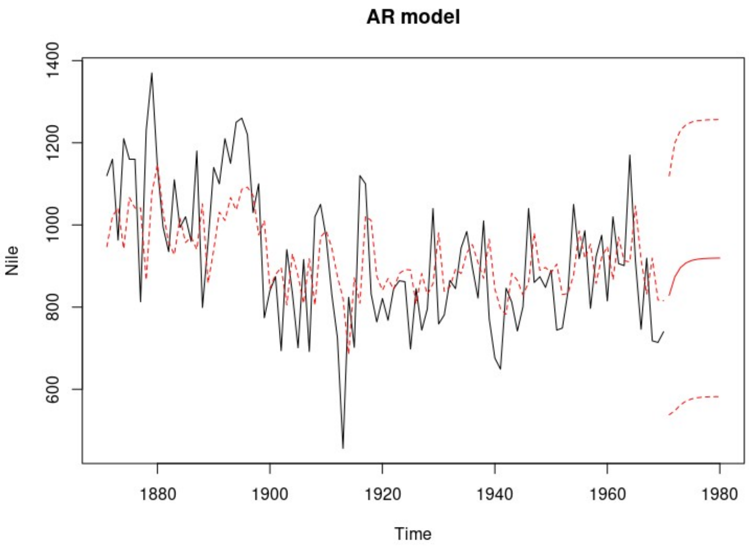


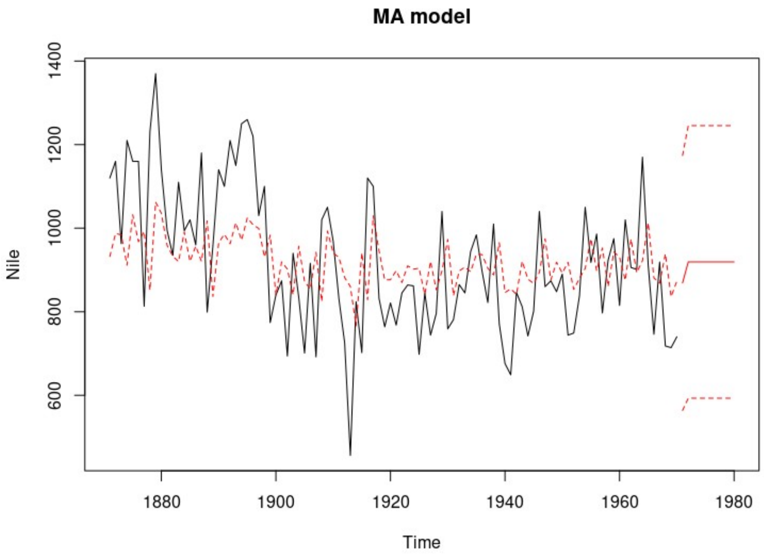


Forecast in R

To fit our time-series on AR model:

AR model ← *arima(x, order=c(1,0,0))*

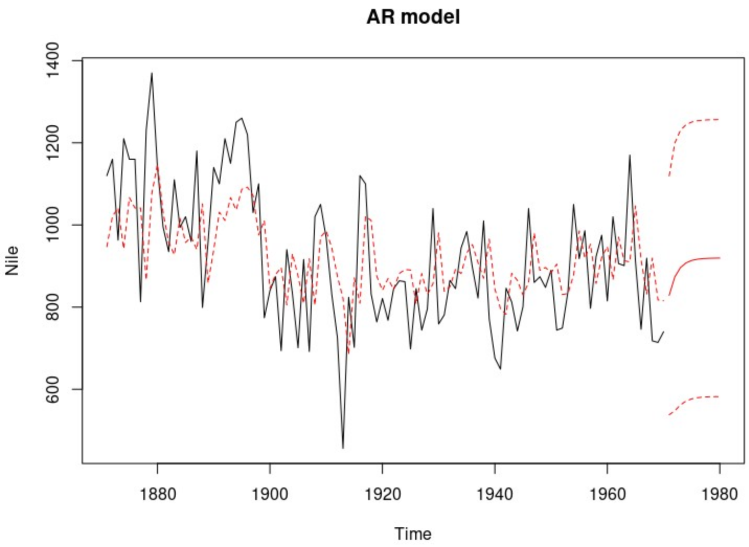




Forecast in R

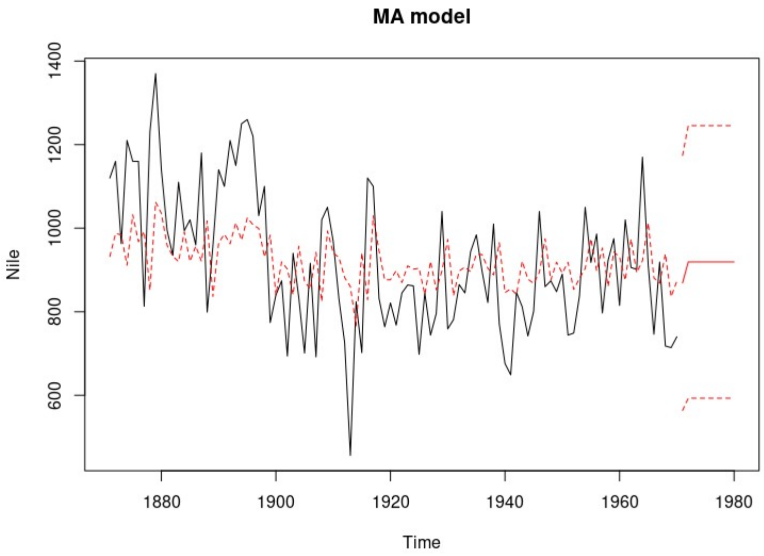
To fit our time-series on AR model:

AR model ← *arima(x, order=c(1,0,0))*



To fit our time-series on MA model:

MA model ← *arima(x, order=c(0,0,1))*



Forecast in R

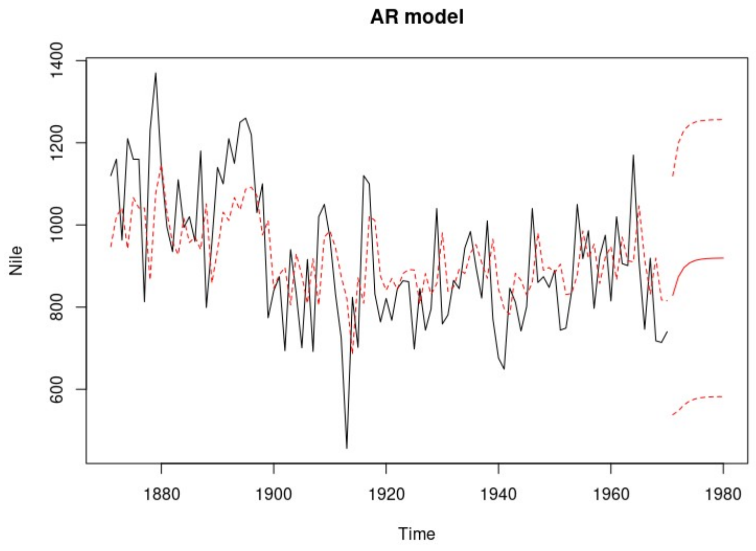
To fit our time-series on AR model:

AR model ← *arima(x, order=c(1,0,0))*

Fitted values ← *x - residuals(x)*

Forecast ← *predict(x)$pred*

Forecast SD ← *predict(x)$sd*



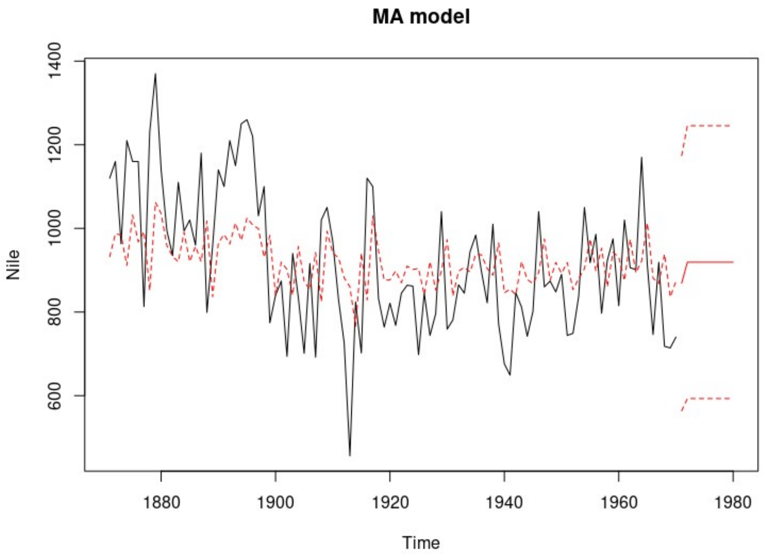
To fit our time-series on MA model:

MA model ← *arima(x, order=c(0,0,1))*

Fitted values ← *x - residuals(x)*

Forecast ← *predict(x)$pred*

Forecast SD ← *predict(x)$sd*



Compare different models

When comparing different models we want to find out which model explain better my data regardless the **Individual independent variables** in the model.

It doesn't mean that there is a right and a wrong model!

Compare different models

When comparing different models we want to find out which model explain better my data regardless the **Individual independent variables** in the model.

It doesn't mean that there is a right and a wrong model!

Akaike's information criteria (AIC):

AIC estimates model complexity. It works estimating the expected performance of model's predictions, for that scope it use observed data and hypothetical sample generated by the same model.

The best model show the smallest value; a difference within **4 - 7** units indicate less support, a difference over **10** indicate that the worse model can be omitted

Limits of time-series

Limits of time-series

- Limited to previous observations

Limits of time-series

- Limited to previous observations
- Seriously affected by NAs values

Limits of time-series

- Limited to previous observations
- Seriously affected by NAs values
- When forecasting doesn’t take into account variables

R