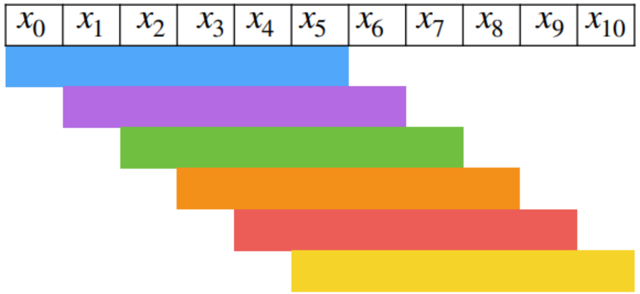
CSC 735 – Data Analytics

Time Series Fundamentals

Windowing

- When analyzing the temporal behaviour of a signal, we often need to evaluate if specific quantities are **time varying** or not
- A common approach is to use **sliding windows** of a given length to evaluate the required values
- So, a sliding window of width **6** on a series of length **11** would look like:



$$= \begin{matrix} x_0 & x_1 & x_2 & x_3 & x_4 & x_5 \\ x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\ x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ x_3 & x_4 & x_5 & x_6 & x_7 & x_8 \\ x_4 & x_5 & x_6 & x_7 & x_8 & x_9 \\ x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} \end{matrix}$$

- And we would calculate the metric of interest **within each window**

Note: first element is represented as x_0 instead of x_1, unlike in some math

Running Values

- We already used **running averages** to detrend a time series
- Recall:

$$m_t = (x_{t-k+1} + x_{t-k+2} + \ldots + x_t)/k.$$

If $t = 16$, $k = 3$, then

$$m_{16} = (x_{14} + x_{15} + x_{16}) / 3$$

- Other common metrics are:
  - Variance
  - Maximum value
  - Minimum value

Running Values

- Notice the formula talks about t where $t \geq k - 1$:

$$m_t = (x_{t-k+1} + x_{t-k+2} + \ ... + \ x_t)/k.$$

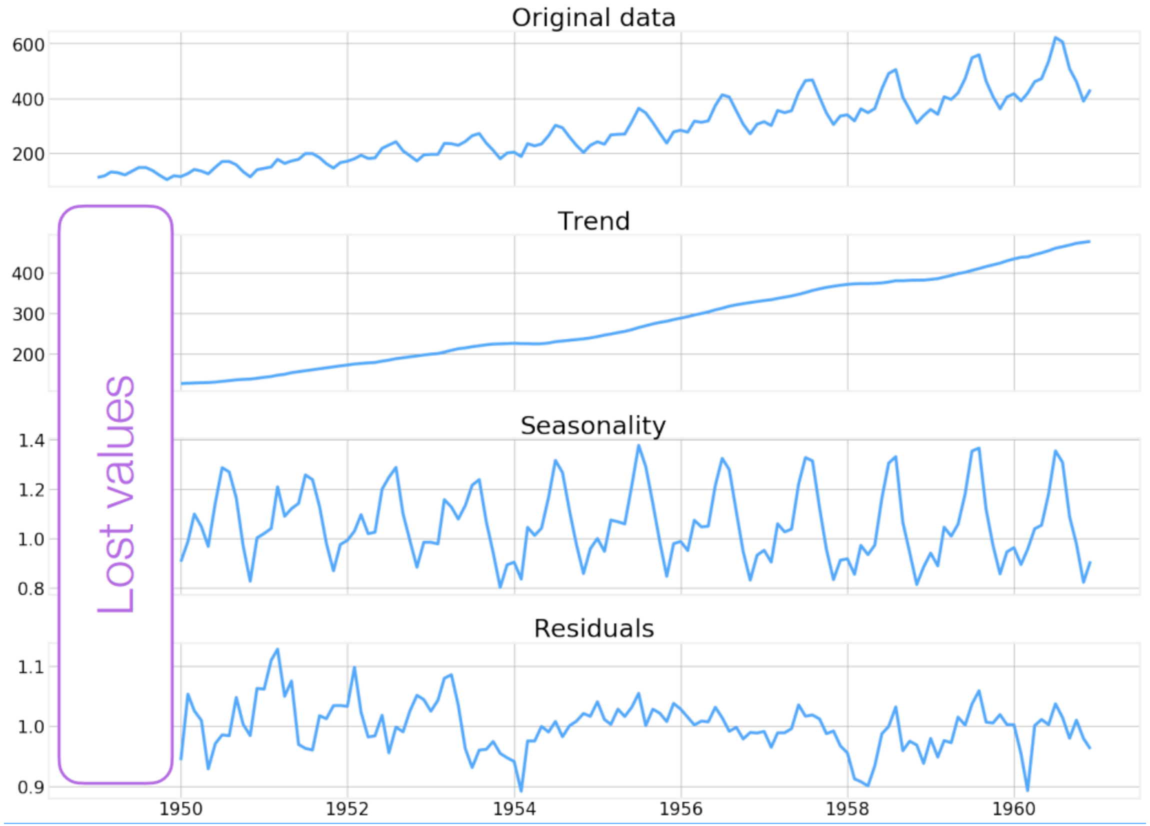
If $t = 3$, $k = 3$, then

$$m_3 = (x_1 + x_2 + x_3) / 3$$

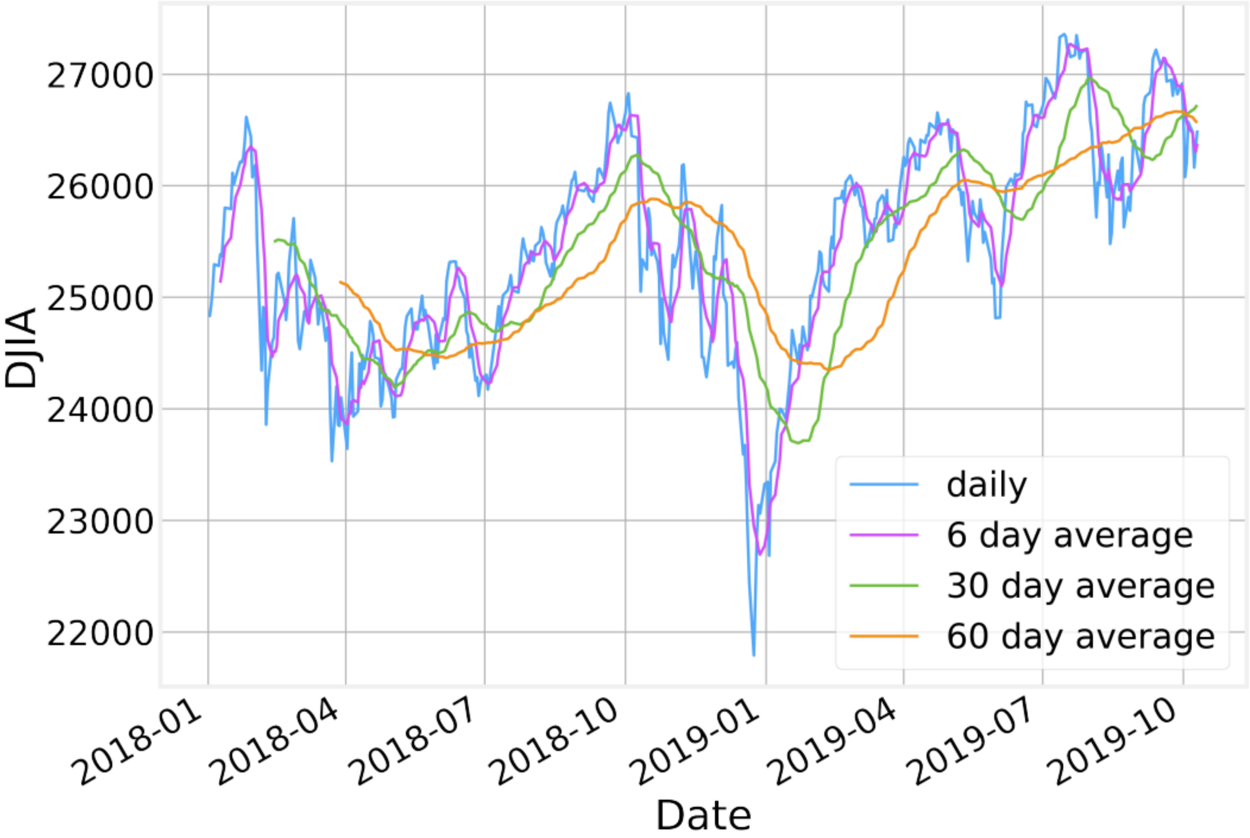
- One important detail to note is that while using windowing to calculate running values, **we "lose" a number of points** equal to the width of the window minus one (*w* – 1)
- Depending on the application, we can choose to consider the missing values to be from either end (or even both ends) of the time interval

Windowing

One common approach is to place all **“lost values”** at the beginning as it avoids **“future leaking”** when splitting the data set



Running Values



Exponential Smoothing

- **Exponential smoothing** (also known as exponential running average) develops a time series model based on declining weights
- Exponential smoothing uses smoothing constants
- The elements of the time series are weighted to ensure that newer observations assume more importance than older observations

Exponential Smoothing

Consider the following formula:

$$\boldsymbol{z_t = \alpha x_t + (1 - \alpha) z_{t-1}}$$

where

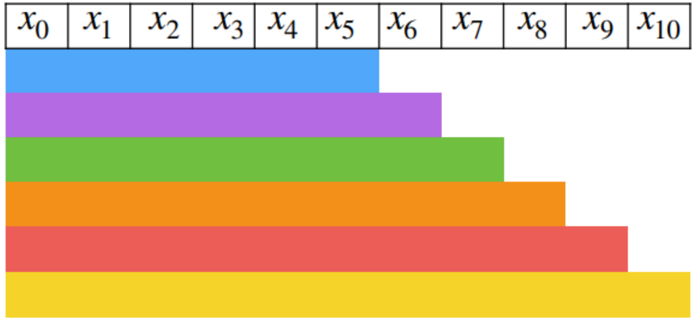
- $\boldsymbol{z_t} =$ Exponentially smoothed value at time $\boldsymbol{t}$
- $\boldsymbol{z_{t-1}} =$ Exponentially smoothed value at time $\boldsymbol{t-1}$
- $\boldsymbol{\alpha} =$ The smoothing constant, $\boldsymbol{\alpha} \in$ **(0,1)**
- $\boldsymbol{x_t} =$ The value of the time series at time $\boldsymbol{t}$
- $\boldsymbol{z_0 = x_0}$ (you cannot go before the start of the time series)

Note: Unlike in some math, we represent the first element as x_0 instead of x_1

Exponential Smoothing

$$\boldsymbol{z_t = \alpha x_t + (1 - \alpha) z_{t-1}}$$

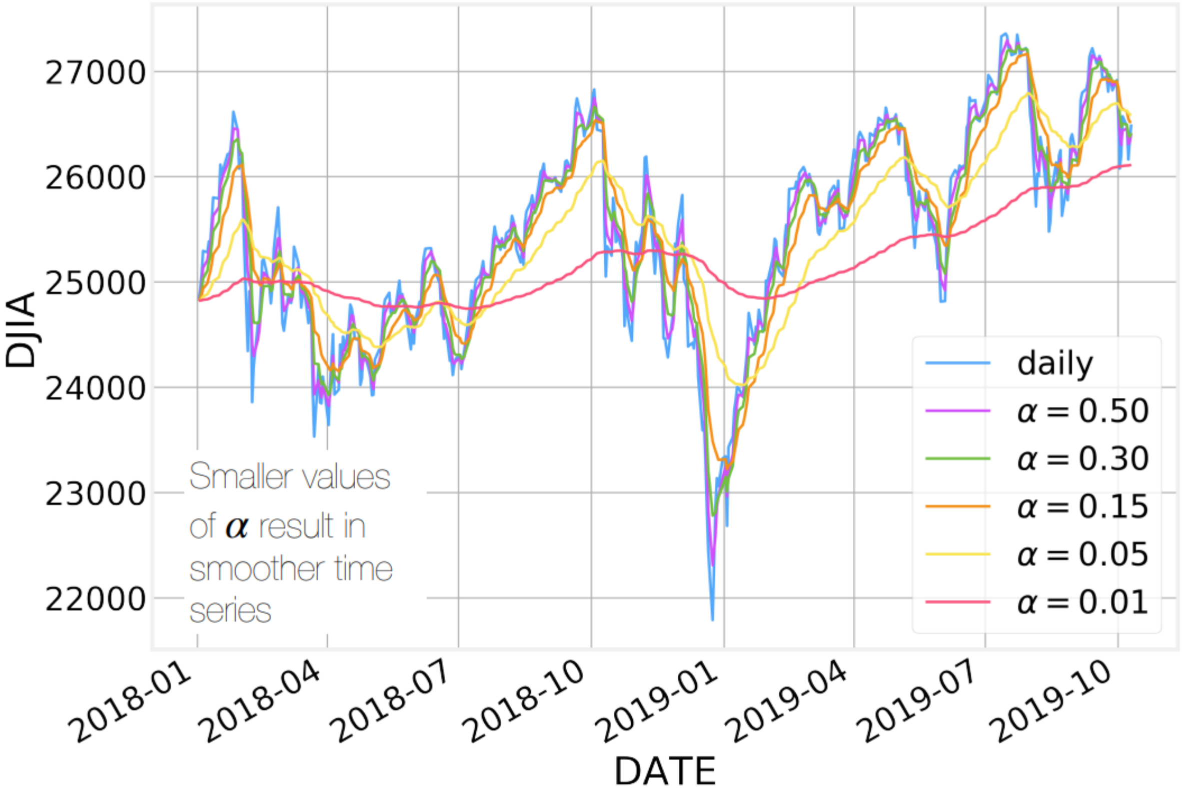
- The smaller the value of the weight $\boldsymbol{\alpha}$, the less influence each point has on the transformed time series.
- Each point depends implicitly on **all previous points**



$$\begin{pmatrix} \alpha & & & & \\ \alpha(1-\alpha)^1 & \alpha & & & \\ \alpha(1-\alpha)^2 & \alpha(1-\alpha)^1 & \alpha & & \\ \vdots & \vdots & \vdots & \ddots & \\ \alpha(1-\alpha)^{n-1} & \alpha(1-\alpha)^{n-2} & \alpha(1-\alpha)^{n-3} & & \alpha \end{pmatrix}$$

Note: Unlike in some math, we represent the first element as x_0 instead of x_1

Exponential Smoothing



Exponential Smoothing Example

End of Day Gas Price in Springfield ($)

t	Date	Actual x_t	Exponential Smoothing $z_t = \alpha x_t + (1-\alpha) z_{t-1}$
0	2022-06-10	2.000	2.000
1	2022-06-11	1.900	-
2	2022-06-12	1.850	-
3	2022-06-13	1.870	-

$$z_1 = \alpha x_1 + (1-\alpha) z_{1-1}$$

NOTE: Assume that $z_0 = x_0$ and $\alpha = 0.50$

This is synthetic data for illustration purposes.

Exponential Smoothing Example

End of Day Gas Price in Springfield ($)

t	Date	Actual x_t	Exponential Smoothing $z_t = \alpha x_t + (1-\alpha) z_{t-1}$
0	2022-06-10	2.000	2.000
1	2022-06-11	1.900	1.950
2	2022-06-12	1.850	-
3	2022-06-13	1.870	-

$$z_1 = \alpha x_1 + (1-\alpha) z_{1-1}$$
$$z_1 = \alpha x_1 + (1-\alpha) z_0$$
$$z_1 = (0.50)(1.900) + (1-0.50)(2.000)$$
$$z_1 = (0.50)(1.900) + (0.50)(2.000)$$
$$z_1 = 0.950 + 1.000$$
$$z_1 = 1.950$$

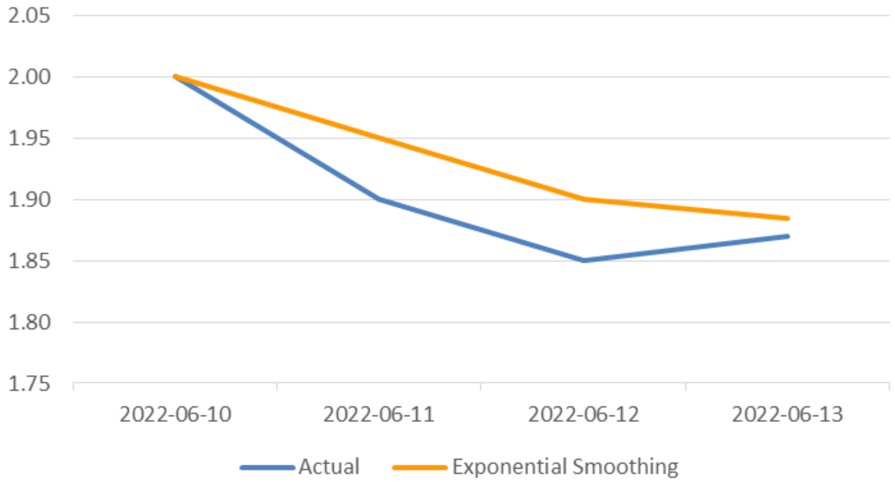
<u>**NOTE:**</u> Assume that $z_0 = x_0$ and $\alpha = 0.50$

This is synthetic data for illustration purposes.

Exponential Smoothing Example

End of Day Gas Price in Springfield ($)

t	Date	Actual x_t	Exponential Smoothing $z_t = \alpha x_t + (1 - \alpha) z_{t-1}$
0	2022-06-10	2.000	2.000
1	2022-06-11	1.900	1.950
2	2022-06-12	1.850	1.900
3	2022-06-13	1.870	1.885



NOTE: Assume that $z_0 = x_0$ and $\alpha = 0.50$

This is synthetic data for illustration purposes.

Forecasting

$$\boldsymbol{z_t = \alpha x_t + (1 - \alpha) z_{t-1}}$$

- We can also use Exponential Smoothing as a simple forecasting tool.

Forecasting

$$\boldsymbol{z_t = \alpha x_t + (1-\alpha) z_{t-1}}$$

- We can also use Exponential Smoothing as a simple forecasting tool.
- Suppose we define a new function $\boldsymbol{z'}$ with a value at time $\boldsymbol{t+1}$ calculated as:

$$\boldsymbol{z'_{t+1} = \alpha x_t + (1-\alpha) z_t}$$

- Which we can consider to be a prediction on the value of $\boldsymbol{x_{t+1}}$

Forecasting

$$\boldsymbol{z_t = \alpha x_t + (1-\alpha) z_{t-1}}$$

- We can also use Exponential Smoothing as a simple forecasting tool.
- Suppose we define a new function $\boldsymbol{z'}$ with a value at time $\boldsymbol{t+1}$ calculated as:

$$\boldsymbol{z'_{t+1} = \alpha x_t + (1-\alpha) z_t}$$

- Which we can consider to be a prediction on the value of $\boldsymbol{x_{t+1}}$
- based on the current value of $\boldsymbol{z_t}$ and some fraction of our current error value $\boldsymbol{x_t - z_t}$:

$$\boldsymbol{z'_{t+1} = z_t + \alpha(x_t - z_t)}$$

Forecasting

$$\boldsymbol{z_t = \alpha x_t + (1 - \alpha) z_{t-1}}$$

- We can also use Exponential Smoothing as a simple forecasting tool.
- Suppose we define a new function $\boldsymbol{z'}$ with a value at time $\boldsymbol{t+1}$ calculated as:

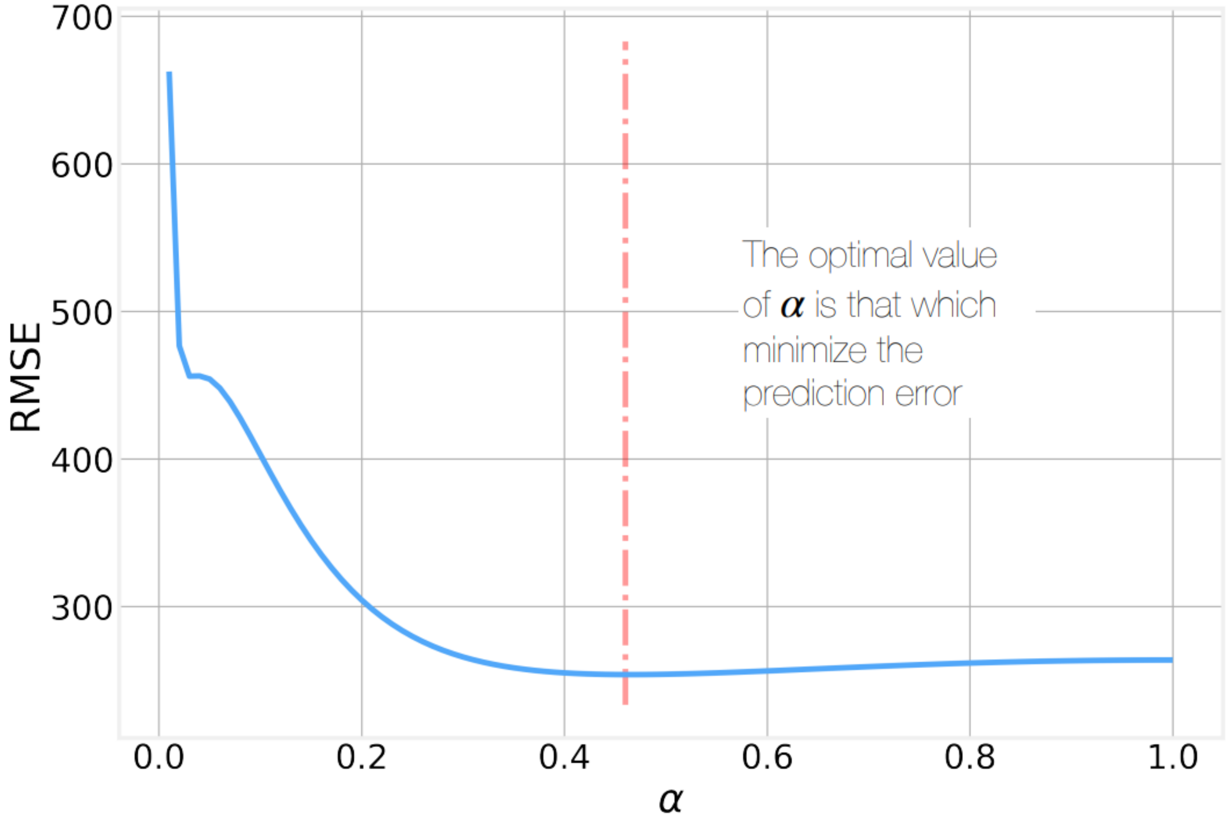
$$\boldsymbol{z'_{t+1} = \alpha x_t + (1 - \alpha) z_t}$$

- Which we can consider to be a prediction on the value of $\boldsymbol{x_{t+1}}$
- based on the current value of $\boldsymbol{z_t}$ and some fraction of our current error value $\boldsymbol{x_t - z_t}$:

$$\boldsymbol{z'_{t+1} = z_t + \alpha(x_t - z_t)}$$

- In this notation, $\boldsymbol{z'_{t+1}}$ represents the forecasted value at time $\boldsymbol{t+1}$ (and it can also be understood to be an estimate of the actual value $\boldsymbol{z_{t+1}}$)

Forecasting



We will cover RMSE in more detail later

Forecasting with Exponential Smoothing Example

End of Day Gas Price in Springfield ($)

t	Date	Actual x_t	Exponential Smoothing $z_t = \alpha x_t + (1 - \alpha) z_{t-1}$
0	2022-06-10	2.000	2.000
1	2022-06-11	1.900	1.950
2	2022-06-12	1.850	1.900
3	2022-06-13	1.870	1.885
4	2022-06-14	?	?

$$\boldsymbol{z'_{t+1} = z_t + \alpha(x_t - z_t)}$$
$$\boldsymbol{z'_4 = z_3 + \alpha(x_3 - z_3)}$$

NOTE: Assume that $z_0 = x_0$ and $\alpha = 0.50$

This is synthetic data for illustration purposes.

Forecasting with Exponential Smoothing Example

Assuming calculations are rounded to decimal points (for easy of human understanding)

End of Day Gas Price in Springfield ($)

t	Date	Actual x_t	Exponential Smoothing $z_t = \alpha x_t + (1-\alpha) z_{t-1}$
0	2022-06-10	2.000	2.000
1	2022-06-11	1.900	1.950
2	2022-06-12	1.850	1.900
3	2022-06-13	1.870	1.885
4	2022-06-14	?	?

NOTE: Assume that $z_0 = x_0$ and $\alpha = 0.50$

$$\boldsymbol{z'_{t+1} = z_t + \alpha(x_t - z_t)}$$
$$\boldsymbol{z'_4 = z_3 + \alpha(x_3 - z_3)}$$
$$\boldsymbol{z'_4 = 1.885 + 0.50(1.870 - 1.885)}$$
$$\boldsymbol{z'_4 = 1.885 + 0.50\,(-0.015)}$$
$$\boldsymbol{z'_4 = 1.885 - 0.0075}$$
$$\boldsymbol{z'_4 = 1.8775} \rightarrow 1.878$$

The estimated value of $\boldsymbol{z_4}$ is $\boldsymbol{1.878}$.

This is synthetic data for illustration purposes.

Forecasting with Exponential Smoothing Example

End of Day Gas Price in Springfield ($)

t	Date	Actual x_t	Exponential Smoothing $z_t = \alpha x_t + (1-\alpha) z_{t-1}$	Forecast $z'_{t+1} = z_t + \alpha(x_t - z_t)$
0	2022-06-10	2.000	2.000	2.000
1	2022-06-11	1.900	1.950	2.000
2	2022-06-12	1.850	1.900	1.925
3	2022-06-13	1.870	1.885	1.875
4	2022-06-14	**?**	**?**	**1.878**

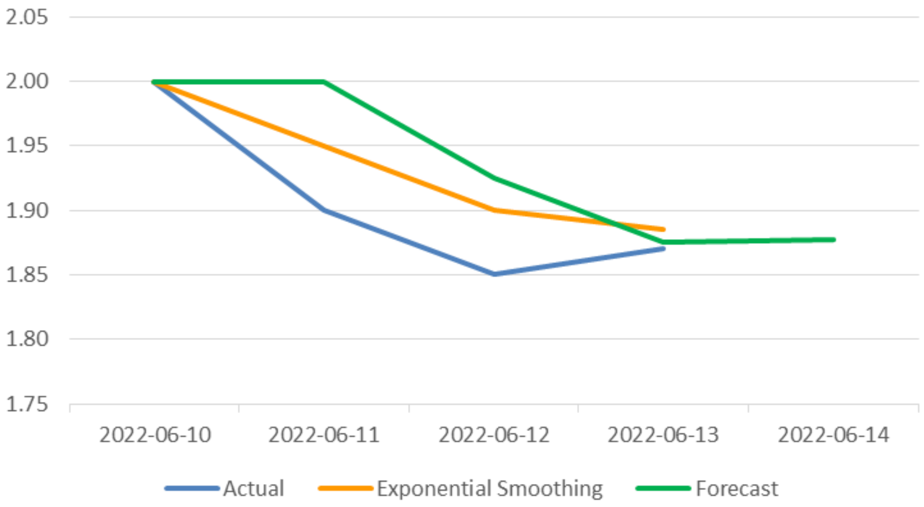
NOTE: Assume that $z'_0 = z_0 = x_0$ and $\alpha = 0.50$

This is synthetic data for illustration purposes.

Forecasting with Exponential Smoothing Example

End of Day Gas Price in Springfield ($)

t	Date	Actual	Exponential Smoothing	Forecast
0	2022-06-10	2.000	2.000	2.000
1	2022-06-11	1.900	1.950	2.000
2	2022-06-12	1.850	1.900	1.925
3	2022-06-13	1.870	1.885	1.875
4	2022-06-14	?	?	**1.878**



NOTE: Assume that $z'_0 = z_0 = x_0$ and $\alpha = 0.50$

This is synthetic data for illustration purposes.

Autoregressive (AR) Model

- A **univariate autoregressive** (AR) **model** takes a certain number of values of a univariate time series as an input and generates a future value using a weighted linear combination of these values and a stochastic term
- A model that takes into account $\boldsymbol{p}$ previous values of the series from before time $\boldsymbol{t}$ (known as lags) is referred to as an AR model of order $\boldsymbol{p}$, an AR($\boldsymbol{p}$) model
- Since it is used to **model the trend** in a dataset, it does NOT generate a stationary time series.

Autoregressive (AR) Model

- In other words, we compute a weighted average over a certain number of **previous values** in the time series
- The weights are derived by a separate process
- For a historical dataset, the weights could be derived from the whole dataset
- An AR model creates a trend for the time series to always revert back to an average behaviour and prevents large fluctuations

Autoregressive (AR) Model

$$\boldsymbol{x_t = c + \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ldots + \phi_p x_{t-p} + \varepsilon_t}$$

$$\boldsymbol{x_t = c + \varepsilon_t + \sum_{i=1}^{p} \phi_i x_{t-i}}$$

where
- $\boldsymbol{x_t}$ = The value that is being generated/predicted
- $\boldsymbol{c}$ = A constant
- $\boldsymbol{\varepsilon_t}$ = A white noise term at time $\boldsymbol{t}$
- $\boldsymbol{\phi_t}$ = The coefficient of a lagged value of $\boldsymbol{x}$ at time $\boldsymbol{t}$ ($\boldsymbol{\phi}$ = phi = FIE)

  $\boldsymbol{p}$ = The number of lags in the process

Note: The phis can be thought of as weights; they are NOT required to add up to one. Formula tells how to generate synthetic data similar to your dataset, assuming your dataset was generated by an AR process.

AR Toy Example

- Assume $c = 0, p = 2$, $\phi_1 = -0.14$, $\phi_2 = -0.28$

$$x'_3 = c + \phi_1 x_{3-1} + \phi_2 x_{3-2} + \varepsilon_3$$

t	x	ε	$x'_t = c + \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ldots + \phi_p x_{t-p} + \varepsilon_t$
0	-0.20	-0.20	-0.20
1	0.40	0.25	0.28
2	1.20	1.24	1.24
3	0.30	-0.78	**??**

AR Toy Example

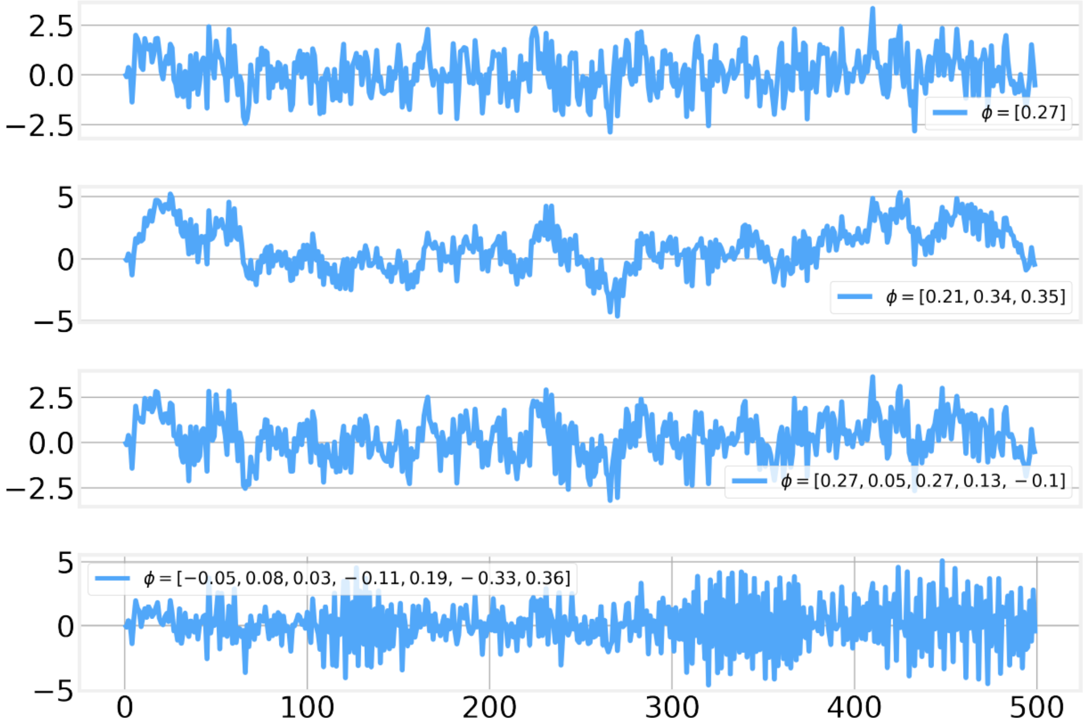
- Assume $\boldsymbol{c} = \boldsymbol{0}, \boldsymbol{p} = \boldsymbol{2}$, $\boldsymbol{\phi_1} =$ -0.14, $\boldsymbol{\phi_2} =$ -0.28

$$\boldsymbol{x'_3 = c + \phi_1 x_{3-1} + \phi_2 x_{3-2} + \varepsilon_3}$$

0 + (-0.14) * 1.20 + (-0.28) * 0.40 + (-0.78) = -1.06

$\boldsymbol{t}$	$\boldsymbol{x}$	$\boldsymbol{\varepsilon}$	$\boldsymbol{x'_t = c + \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ldots + \phi_p x_{t-p} + \varepsilon_t}$
0	-0.20	-0.20	-0.20
1	0.40	0.25	0.28
2	1.20	1.24	1.24
3	0.30	-0.78	**-1.06**

AR Example



Assume we consistently reuse some data set for each experiment here.

Notice that as you increase the number of weights (the autoregressive components that push the time series back to the average behaviour), the more fluctuations you see in the time series.

https://learning.oreilly.com/videos/times-series-analysis/9780136944515/9780136944515-TSA1_01_09_02/

Moving Average (MA) Model

- A **univariate moving average model**, also known as an MA model, is a composed of a linear combination of white **noise** terms
- It is NOT the same idea as what is commonly called a moving average
- Like the AR model, the MA model takes into account a finite number of previous data points
- MA($\boldsymbol{q}$) model denotes a moving average model of order $\boldsymbol{q}$

Moving Average (MA) Model

$$\boldsymbol{x_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \ldots + \theta_q \varepsilon_{t-q}}$$

$$\boldsymbol{x_t = \mu + \varepsilon_t + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i}}$$

- $\boldsymbol{x_t}$ = The value that is being generated/predicted
- $\boldsymbol{\mu}$ = A constant sometimes referred to as the **bias**
- $\boldsymbol{\varepsilon_t}$ = A white noise term at time $\boldsymbol{t}$
- $\boldsymbol{\theta_t}$ = The coefficient of the white noise term at time $\boldsymbol{t}$
- $\boldsymbol{q}$ = The number of lags in the process

MA Toy Example

- Assume that $\boldsymbol{\mu = 0, q = 1, x'_0 = \varepsilon_0}$, and $\boldsymbol{\theta_1 = 0.681}$.

$$\boldsymbol{x_3 = \mu + \varepsilon_3 + \theta_1 \varepsilon_{3-1}}$$

$\boldsymbol{t}$	$\boldsymbol{\varepsilon}$	$\boldsymbol{x_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \ldots + \theta_q \varepsilon_{t-q}}$
0	-0.589	-0.589
1	-0.574	-0.975
2	-1.069	-1.460
3	0.650	**?**

MA Toy Example

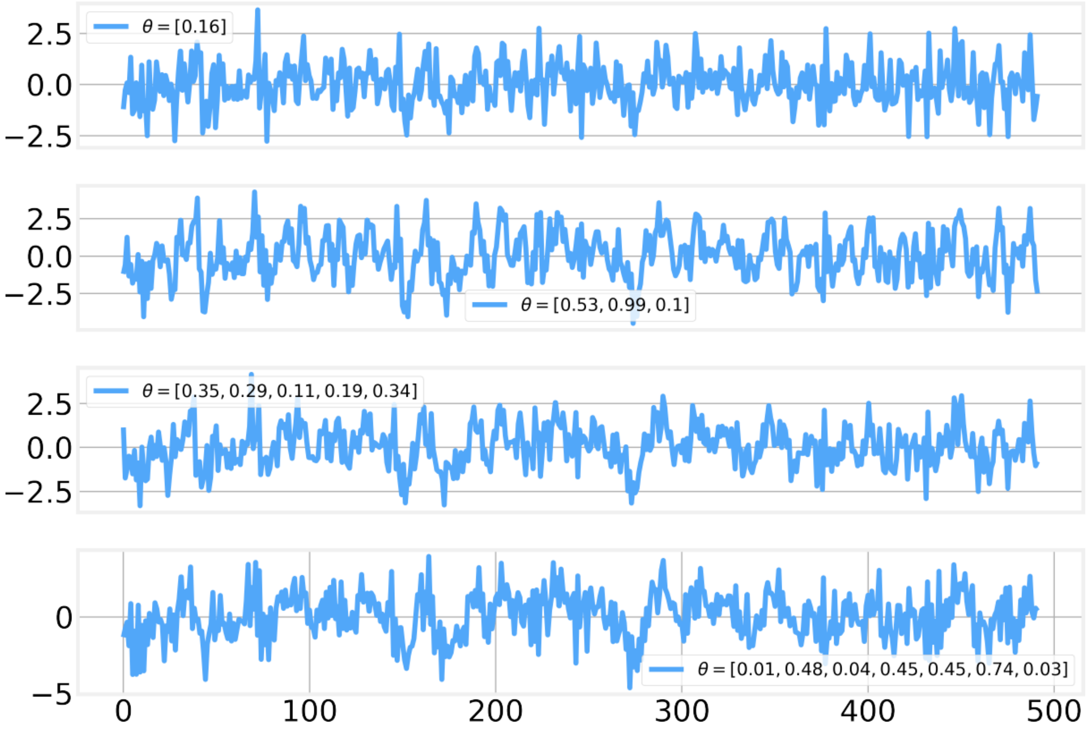
- Assume that $\boldsymbol{\mu = 0, q = 1}$, $\boldsymbol{x'_0 = \varepsilon_0}$, and $\boldsymbol{\theta_1 = 0.681}$.

$$\boldsymbol{x_3 = \mu + \varepsilon_3 + \theta_1 \varepsilon_{3-1}}$$
$$\boldsymbol{x_3 = \mu + \varepsilon_3 + \theta_1 \varepsilon_2}$$
$$\boldsymbol{x_3 = 0 + 0.650 + (0.681)(-1.069)}$$
$$\boldsymbol{x_3 = -0.078}$$

The data values enter the process as “error values”, i.e., stochastic (randomly chosen) values.

$\boldsymbol{t}$	$\boldsymbol{\varepsilon}$	$\boldsymbol{x'_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \ldots + \theta_q \varepsilon_{t-q}}$
0	-0.589	-0.589
1	-0.574	-0.975
2	-1.069	-1.460
3	0.650	**-0.078**

MA Example

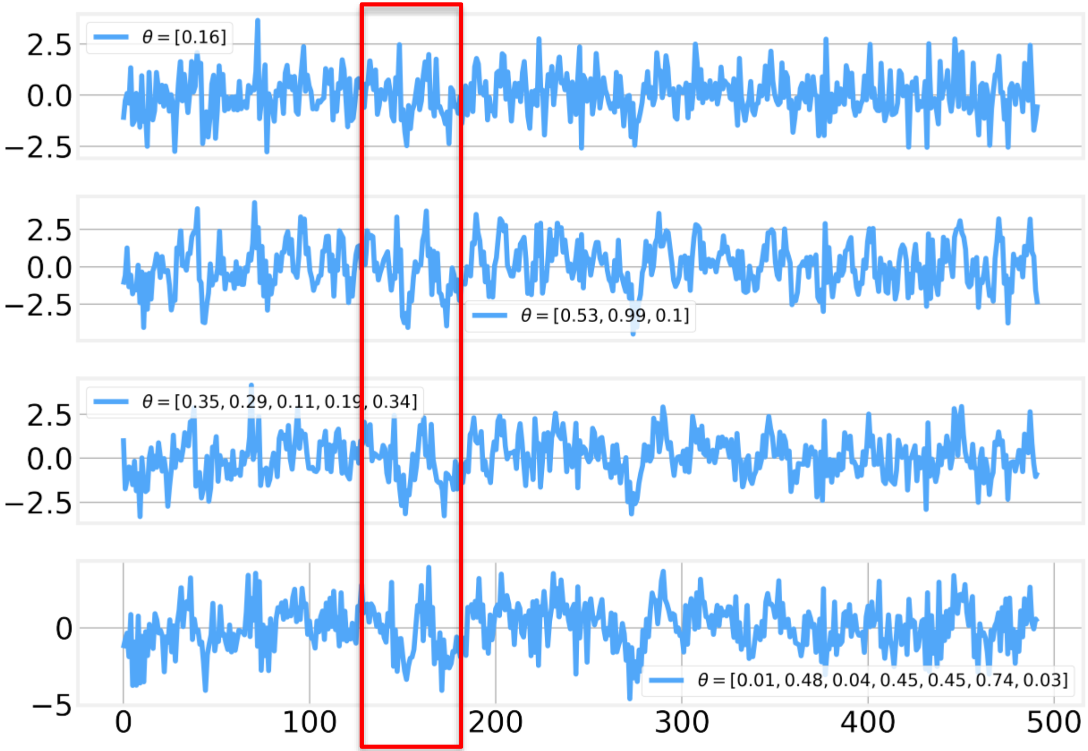


Assume we consistently reuse some data set for each experiment here.

Notice that as you increase the number of weights (lags), you add extra periods in the time series where the time series will increase or decrease.

https://learning.oreilly.com/videos/times-series-analysis/9780136944515/9780136944515-TSA1_01_09_01/

MA Example



Assume we consistently reuse some data set for each experiment here.

Notice that as you increase the number of weights (lags), you add extra periods in the time series where the time series will increase or decrease.

https://learning.oreilly.com/videos/times-series-analysis/9780136944515/9780136944515-TSA1_01_09_01/

Autoregressive Moving Average (ARMA) Model

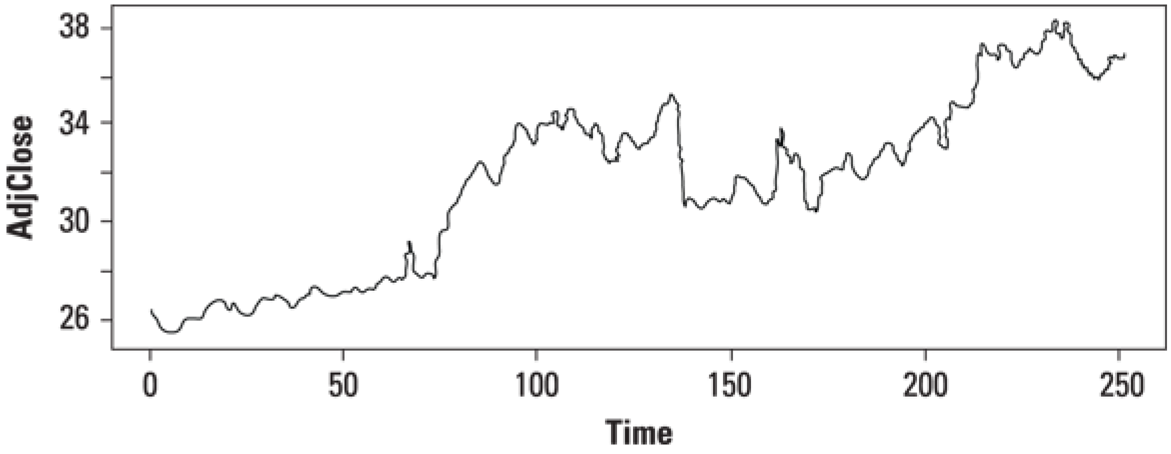
- An **autoregressive moving average (ARMA) model** is a combination of an autoregressive model and a moving average model:

$$\boldsymbol{x_t = c + \varepsilon_t + \sum_{i=1}^{p} \phi_i x_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i}}$$

- This model is denoted as ARMA($\boldsymbol{p, q}$)

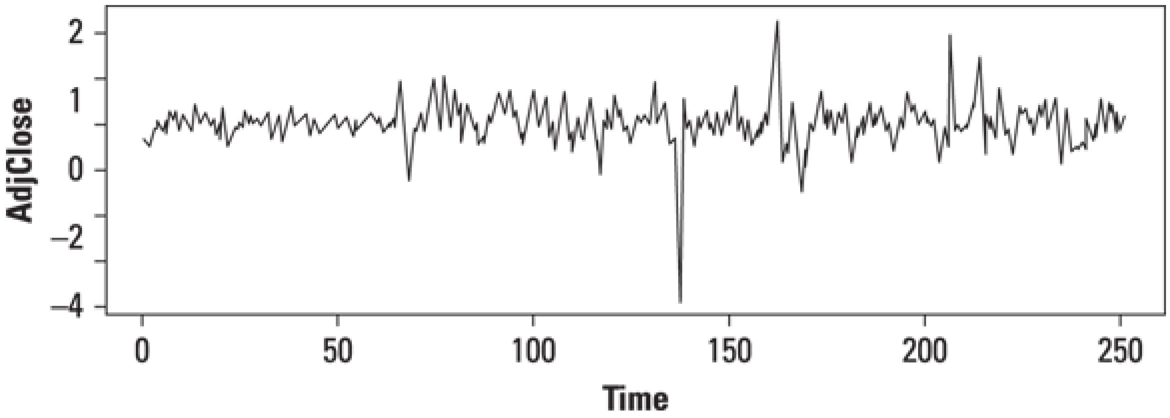
ARMA Example

- Consider this plot of Microsoft stock prices during a specific time period



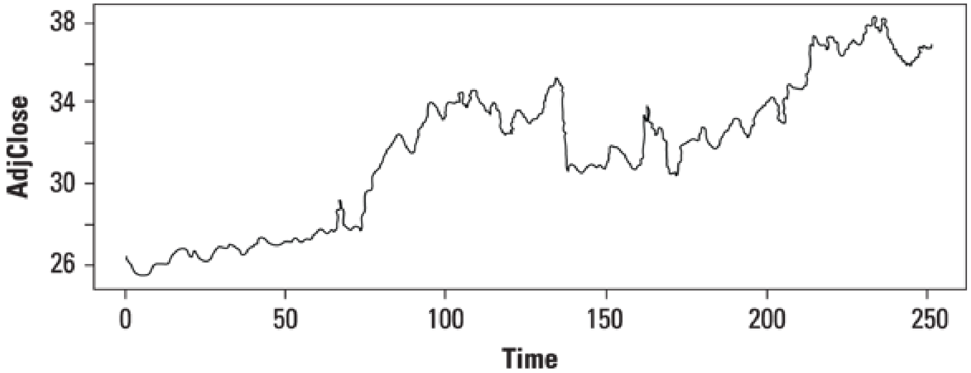
ARMA Example

- First-differencing is applied to convert the prices to a stationary series

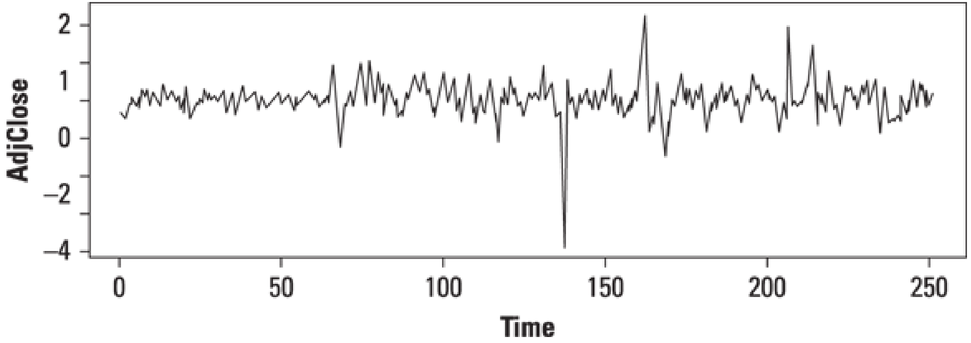


ARMA Example

BEFORE

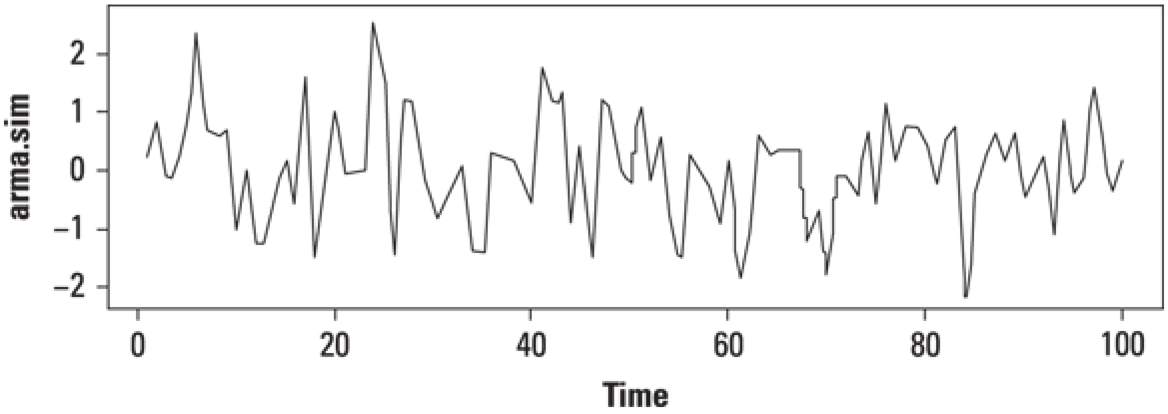


AFTER



ARMA Example

A simulated version of the ARMA$(\mathbf{1}, \mathbf{1})$ model on Microsoft stock prices

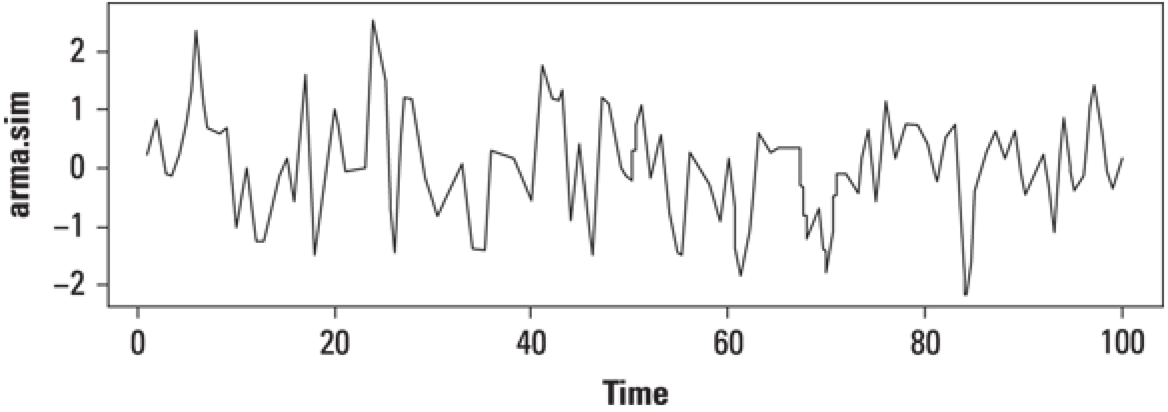


ARMA Example

A simulated version of the ARMA($\mathbf{1}, \mathbf{1}$) model on Microsoft stock prices

- Applying linear regression and other complicated methods finds constants to fit into this equation:

$$\boldsymbol{x_t = c + \varepsilon_t + 0.6x_{t-1} + 0.8\varepsilon_{t-1}}$$



ARMA Toy Example

- Generate data, assuming that $\boldsymbol{c = 0}$, $\boldsymbol{\phi_1 = 0.6}$, $\boldsymbol{\theta_1 = 0.8}$, and that we are using an ARMA$(\mathbf{1,1})$ model

$$\boldsymbol{x_t = c + \varepsilon_t + 0.6x_{t-1} + 0.8\varepsilon_{t-1}}$$

$\boldsymbol{t}$	$\boldsymbol{x}$	$\boldsymbol{\varepsilon}$	
…	…	…	
5	1.20	0.20	
6	?	0.30	
…	…	…	

ARMA Toy Example

- Generate data, assuming that $\boldsymbol{c = 0}$, $\boldsymbol{\phi_1 = 0.6}$, $\boldsymbol{\theta_1 = 0.8}$, and that we are using an ARMA$(\boldsymbol{1, 1})$ model

$$\boldsymbol{x_t = c + \varepsilon_t + 0.6x_{t-1} + 0.8\varepsilon_{t-1}}$$

$$\boldsymbol{x_6 = c + \varepsilon_6 + 0.6x_{6-1} + 0.8\varepsilon_{6-1}}$$

t	x	ε	
…	…	…	
5	1.20	0.20	
6	?	0.30	
…	…	…	

ARMA Toy Example

- Assume that $\boldsymbol{c = 0}$, $\boldsymbol{\phi_1 = 0.6}$, $\boldsymbol{\theta_1 = 0.8}$, and that we are using an ARMA$(\boldsymbol{1, 1})$ model

$$\boldsymbol{x_t = c + \varepsilon_t + 0.6x_{t-1} + 0.8\varepsilon_{t-1}}$$

$$\boldsymbol{x_6 = c + \varepsilon_6 + 0.6x_{6-1} + 0.8\varepsilon_{6-1}}$$

$$\boldsymbol{x_6 = c + \varepsilon_6 + 0.6x_5 + 0.8\varepsilon_5}$$

$$\boldsymbol{x_6 = 0 + 0.30 + (0.6)(1.20) + (0.8)(0.20)}$$

$$\boldsymbol{x_6 = 1.18}$$

$\boldsymbol{t}$	$\boldsymbol{x}$	$\boldsymbol{\varepsilon}$	
…	…	…	
5	1.20	0.20	
6	**1.18**	0.30	
…	…	…	…

Autoregressive Integrated Moving Average (ARIMA) Model

- The autoregressive integrated moving average (ARIMA) model is similar to ARMA, but it incorporates differencing
- The ARIMA model, denoted ARIMA($\boldsymbol{p, d, q}$), specifies that the order of differencing is $\boldsymbol{d}$
- The **order of differencing** is a measure of how many times the data is differenced before being used by the model
- Recall that differencing can "stationarize" the time series

Autoregressive Integrated Moving Average (ARIMA) Model

Auto Regressive	The time series is regressed with its previous values. The order of the lag is $\boldsymbol{p}$.	
	$\boldsymbol{x_t = c + \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ldots + \phi_p x_{t-p} + \varepsilon_t}$	
Integrated	The time series uses differencing to make it stationary. The order of the difference is $\boldsymbol{d}$.	
Moving Average	The time series is regressed with residuals of the past observations. The order of the error lag is $\boldsymbol{q}$.	
	$\boldsymbol{x_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \ldots + \theta_q \varepsilon_{t-q}}$	
ARIMA($\boldsymbol{p, d, q}$)	$\boldsymbol{x_t = c + \varepsilon_t + \sum_{i=1}^{p} \phi_i x_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i}}$	

Vector Autoregressive (VAR) Model

- The **Vector** Autoregressive (VAR) model is one of the most commonly used models for **multivariate** time series analysis and prediction
- A VAR model of order $\boldsymbol{p}$, denoted VAR($\boldsymbol{p}$), is similar to an AR($\boldsymbol{p}$) model but it is defined over more than one variable and predicts values for more than one variable (multivariate)

Vector Autoregressive (VAR) Model

- Consider the following formula:

$$\boldsymbol{x_t} = \boldsymbol{c} + \boldsymbol{\phi_1}\boldsymbol{x_{t-1}} + \boldsymbol{\phi_2}\boldsymbol{x_{t-2}} + \ldots + \boldsymbol{\phi_p}\boldsymbol{x_{t-p}} + \boldsymbol{\varepsilon_t}$$

where

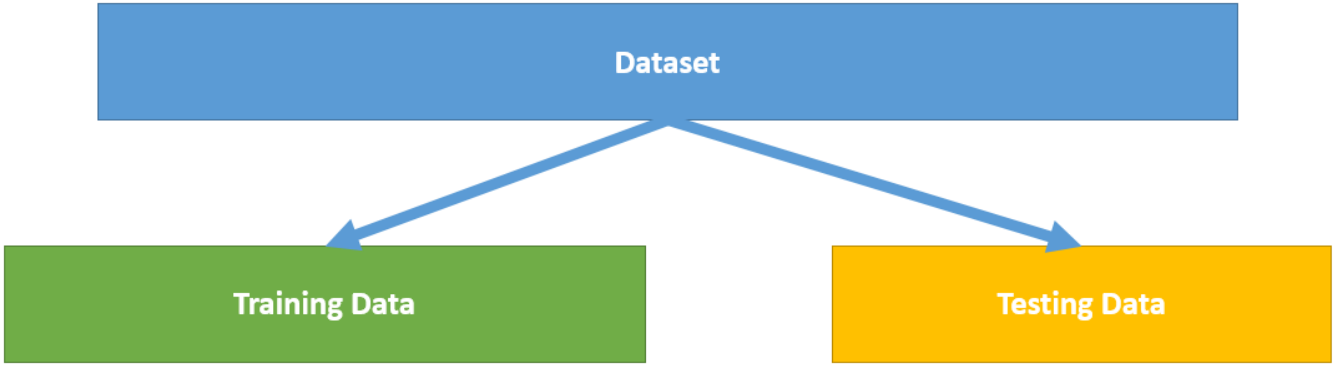
- $\boldsymbol{x_t} =$ A set of $\boldsymbol{n}$ x $\boldsymbol{1}$ vector of variables $(\boldsymbol{x_{1t}}, \boldsymbol{x_{2t}}, \ldots, \boldsymbol{x_{nt}})$
- $\boldsymbol{c} =$ A set of $\boldsymbol{n}$ x $\boldsymbol{1}$ vector of constants
- $\boldsymbol{\varepsilon_t} =$ A set of $\boldsymbol{n}$ x $\boldsymbol{1}$ vector of error terms
- $\boldsymbol{\phi_i}$ for $\boldsymbol{1} \leq \boldsymbol{i} \leq \boldsymbol{p} =$ A set of $\boldsymbol{n}$ x $\boldsymbol{n}$ matrix of model parameters with a parameter for each lag value of each variable
- $\boldsymbol{p} =$ The number of lags in the process

Evaluating Your Model

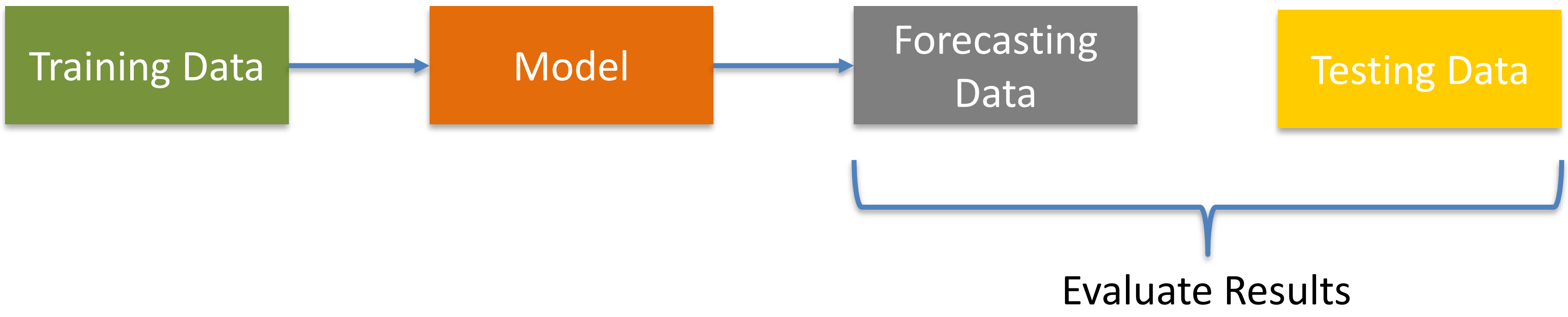
- There are several evaluation methods for evaluating the performance of a time series model
- We will first discuss how to divide your data set so that you can evaluate your time series model

Dividing the Data Set

- We can split our data set into two sets: commonly, we split 80% for training and 20% for testing



Model Evaluation



Process diagram of how a model's performance can be evaluated by comparing the forecasting data to the testing data

Evaluation Methods

- We can evaluate the results of our experiments using the **Mean Absolute Percentage Error** (MAPE) as our error metric
- We can also use **Root Mean Square Error** (RMSE)

Mean Absolute Percentage Error

- Consider the following formula:

$$MAPE = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{x_i - x'_i}{x_i}\right|$$

where

- $N =$ number of predicted data points
- $x_i =$ the actual value of the i^{th}data point
- $x'_i =$ the forecast (predicted) value
- Multiplying the value by 100 gives a percentage error

Root Mean Square Error (RMSE)

- Consider the following formula:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - x'_i)^2}$$

where

- N = number of predicted data points
- x_i = the actual value of the i^{th}data point
- x'_i = the forecast (predicted) value

Root Mean Square Error (RMSE)

- A value of **0** (zero) for RMSE indicates a perfect fit to the data (this rarely occurs in practice)
- A “good” value of RMSE depends on the data set you are working with
- In general, RMSE values close to **0** indicate better fitting models

Normalized RMSE

- You can normalize RMSE to have a better idea of how your model fits the data set
- Consider the following formula:

$$Normalized\ RMSE = \frac{RMSE}{(Maximum\ Value\ - Minimum\ Value)}$$