Data Computing – Time Series

Chapter 8 from "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data"
1st Edition by EMC Education Services

Ka-Chun Wong, Department of Computer Science, City University of Hong Kong
Charles Tappert Seidenberg, School of CSIS, Pace University

Outline

- 8.1 Overview of Time Series Analysis
  - 8.1.1 Box-Jenkins Methodology
- 8.2 ARIMA Model
  - 8.2.1 Autocorrelation Function (ACF)
  - 8.2.2 Autoregressive Models
  - 8.2.3 Moving Average Models
  - 8.2.4 ARMA and ARIMA Models
  - 8.2.5 Building and Evaluating an ARIMA Model
  - 8.2.6 Reasons to Choose and Cautions
- 8.3 Additional Methods
- Summary

8 Time Series Analysis

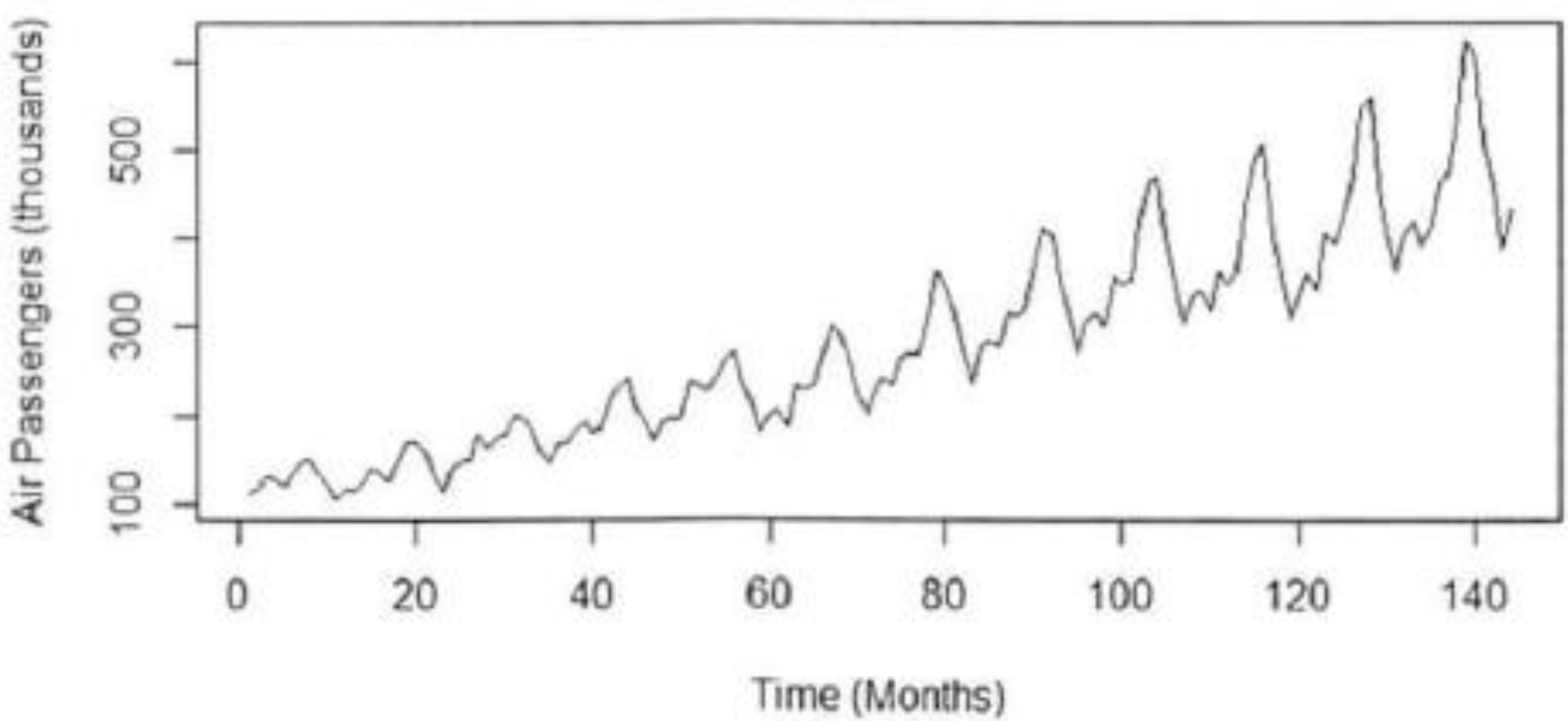
- This lecture’s emphasis is on
  - Identifying the underlying structure of the time series
  - Fitting an appropriate Autoregressive Integrated Moving Average (ARIMA) model

8.1 Overview of Time Series Analysis

- Time series analysis attempts to model the underlying structure of observations over time.
- A *time series* is an ordered sequence of equally spaced values over time.
- The analyses presented are limited to equally spaced time series of one variable.

8.1 Overview of Time Series Analysis

- The time series below plots #passengers vs months (144 months or 12 years)



8.1 Overview of Time Series Analysis

- Why are time series different from other data types you have learned before?
- Data are not independent
  - Much of the statistical theory relies on the data being independent and identically distributed
- Large samples sizes are good, but long time series are not always the best
  - Series often change with time, so bigger isn't always better

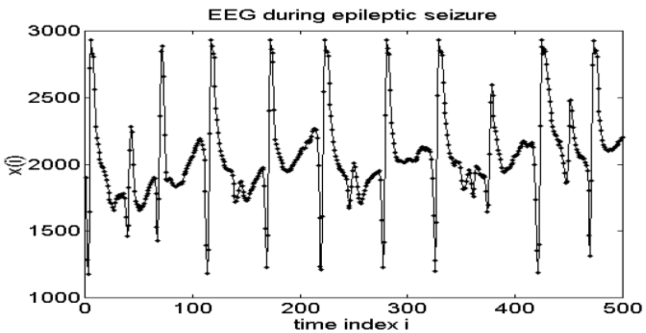
8.1 Overview of Time Series Analysis

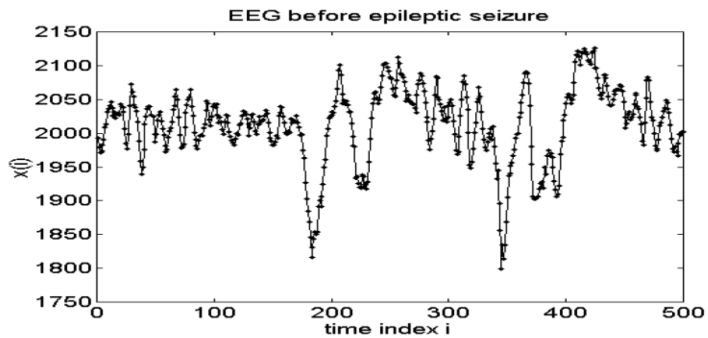
- The goals of time series analysis:
  - To model the structure of the time series
  - To forecast future values in the time series
- Time series analysis has many applications in finance, economics, biology, engineering, retail, and manufacturing

8.1 Overview of Time Series Analysis

- Time Series Examples
  - Number of babies born in each hour
  - Daily closing price of a stock
  - Monthly trade balance for each year.
  - GDP of the country, measured every year.
  - Your GPA, measured every semester.
  - Your youth height, measured every year.
  - Traveling time to work every weekday.
  - Blood pressure , measured every second or day.

physiology

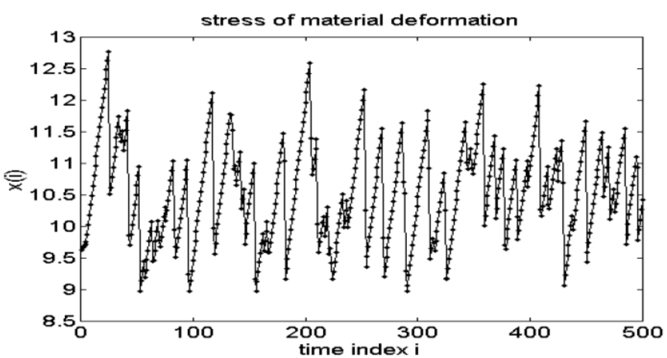


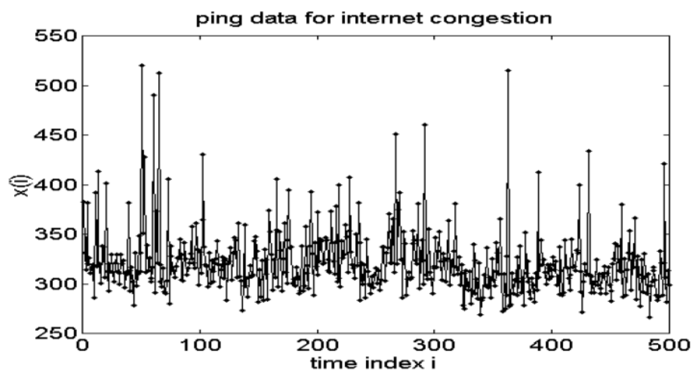


univariate time series

mechanics

electronics



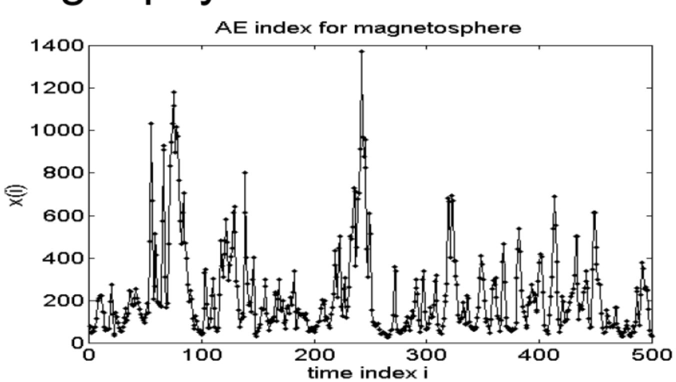


only one time series

limited length

geophysics

economy



non-stationarity

noise

https://www.slideshare.net/Whitaker_Institute/20130617-timeseriespart1?from_action=save

8.1 Overview of Time Series Analysis

- How the time series data and time (t) is recorded and presented

$(y_1, y_2, y_3, \dots, y_T)$

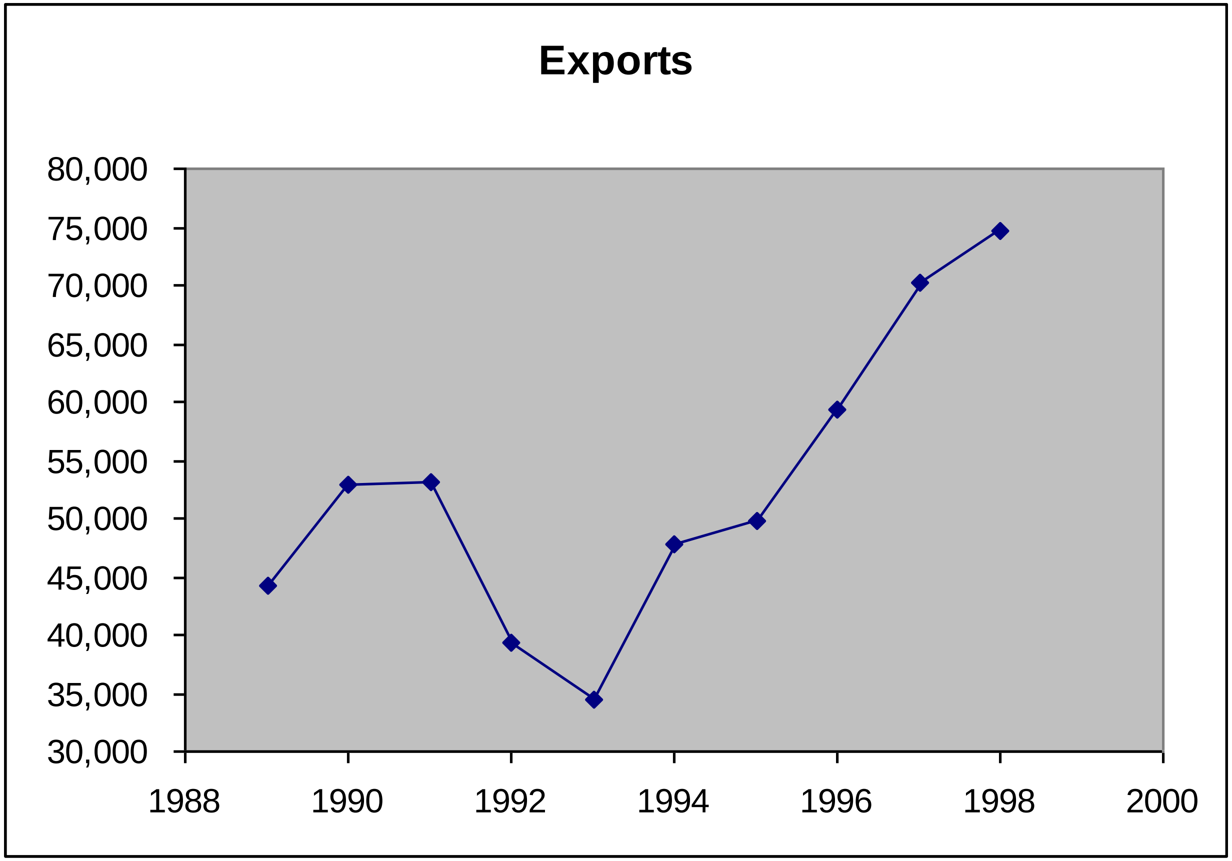
Exports, 1989-1998

Year	t	
1989	1	$y_1 =$ 44,320
1990	2	$y_2 =$ 52,865
1991	3	$y_3 =$ 53,092
1992	4	$y_4 =$ 39,424
1993	5	$y_5 =$ 34,444
1994	6	$y_6 =$ 47,870
1995	7	$y_7 =$ 49,805
1996	8	$y_8 =$ 59,404
1997	9	$y_9 =$ 70,214
1998	10	$y_{10} =$ 74,626

8.1 Overview of Time Series Analysis

Exports, 1989-1998

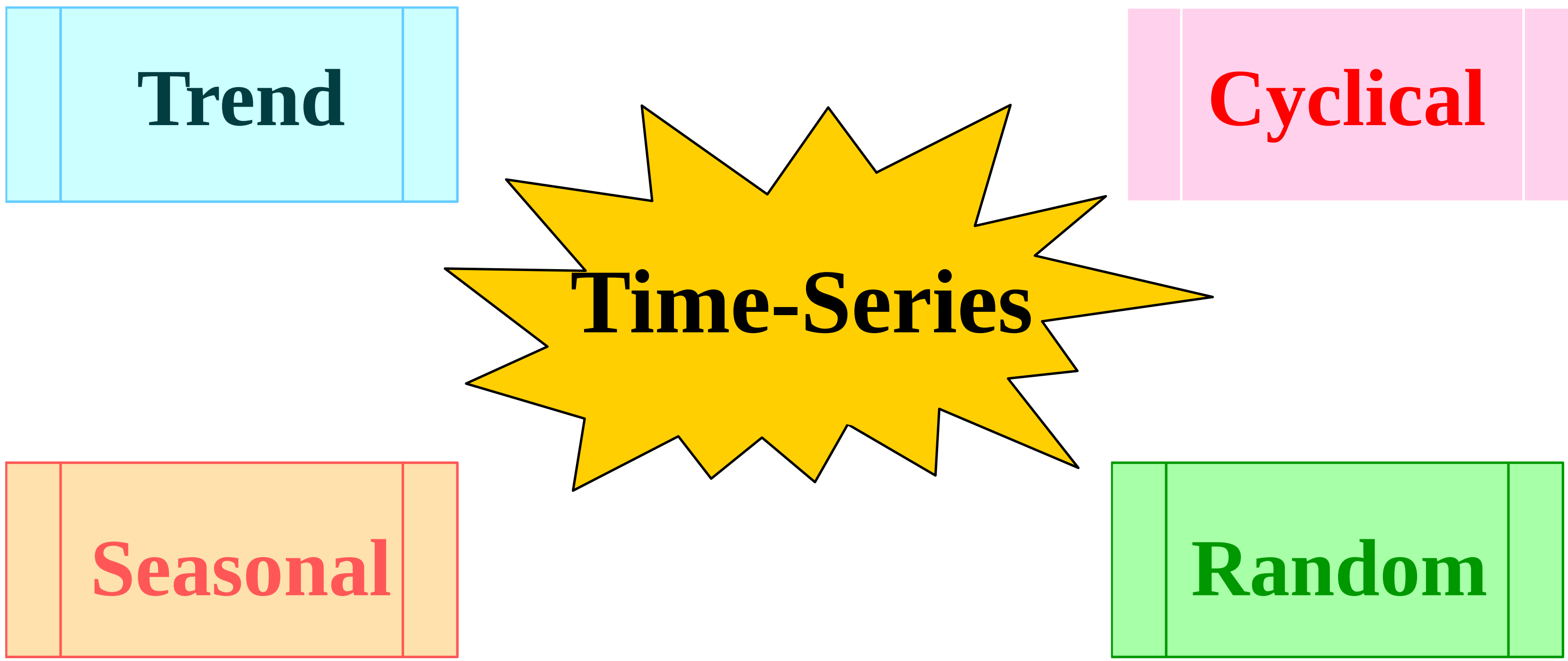
Year	t	
1989	1	44,320
1990	2	52,865
1991	3	53,092
1992	4	39,424
1993	5	34,444
1994	6	47,870
1995	7	49,805
1996	8	59,404
1997	9	70,214
1998	10	74,626



8.1 Overview of Time Series Analysis

- Important features of time series include:
  - Direction
  - Turning points
  - In addition, we want to see if the series is increasing/decreasing more slowly/faster than it was before

Time-Series Components



Components of Time Series

- Trend (T_t)
- Seasonal variation (S_t)
- Cyclical variation (C_t)
- Random variation (R_t) or irregular

Components of Time Series

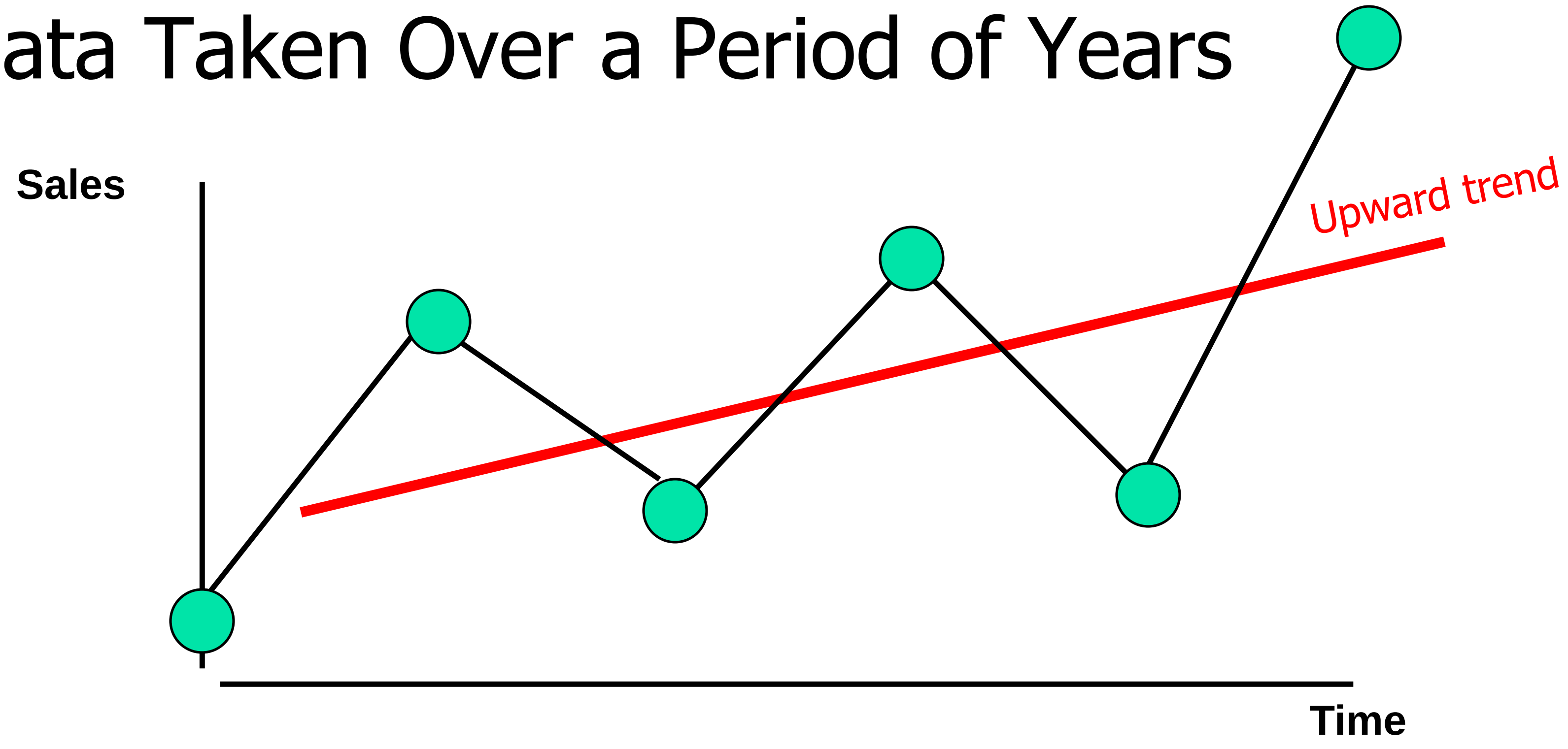
Trend (T_t)

- Trend: the long-term patterns or movements in the data.
- Overall or persistent, long-term upward or downward pattern of movement.
- The trend of a time series is not always linear.

Components of Time Series

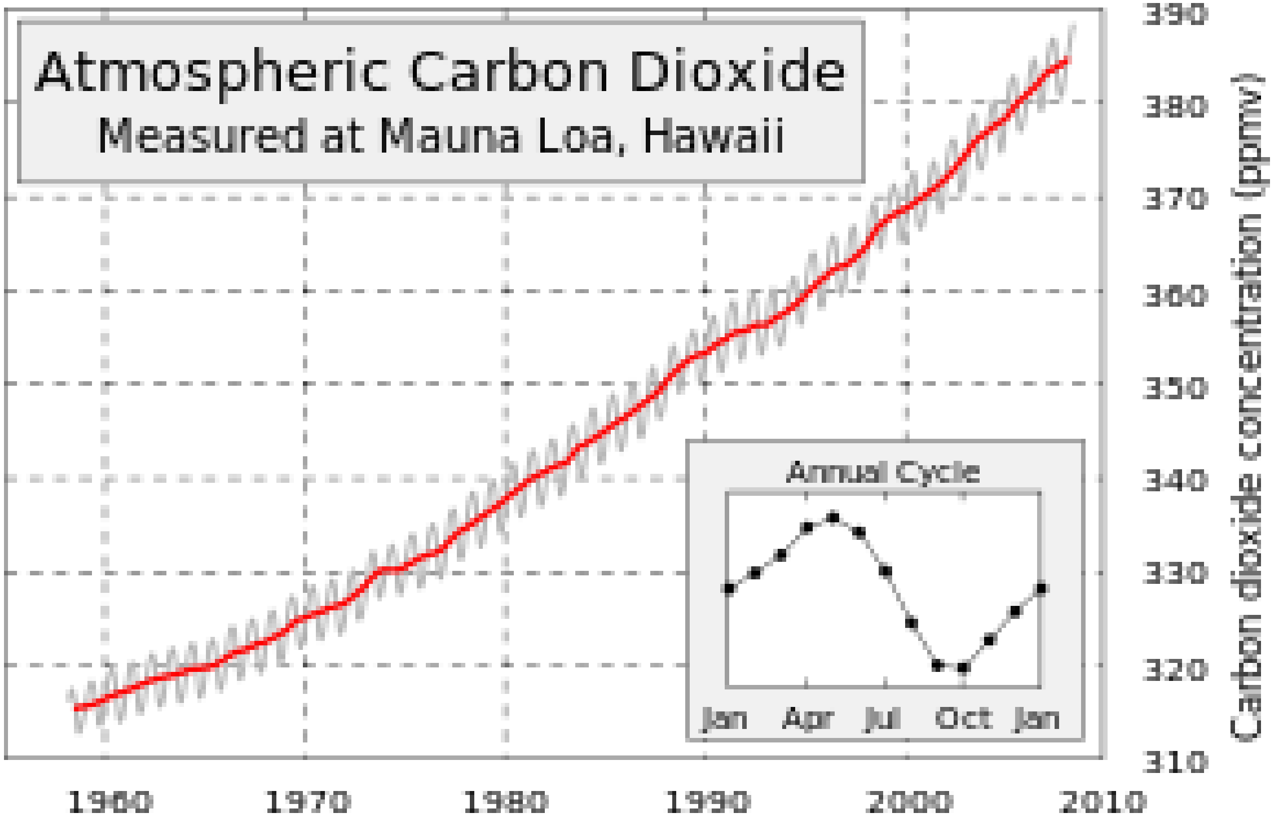
Trend (T_t)

- Overall Upward or Downward Movement
- Data Taken Over a Period of Years



Components of Time Series

Trend (T_t)



“Keeling curve”, from Wikipedia

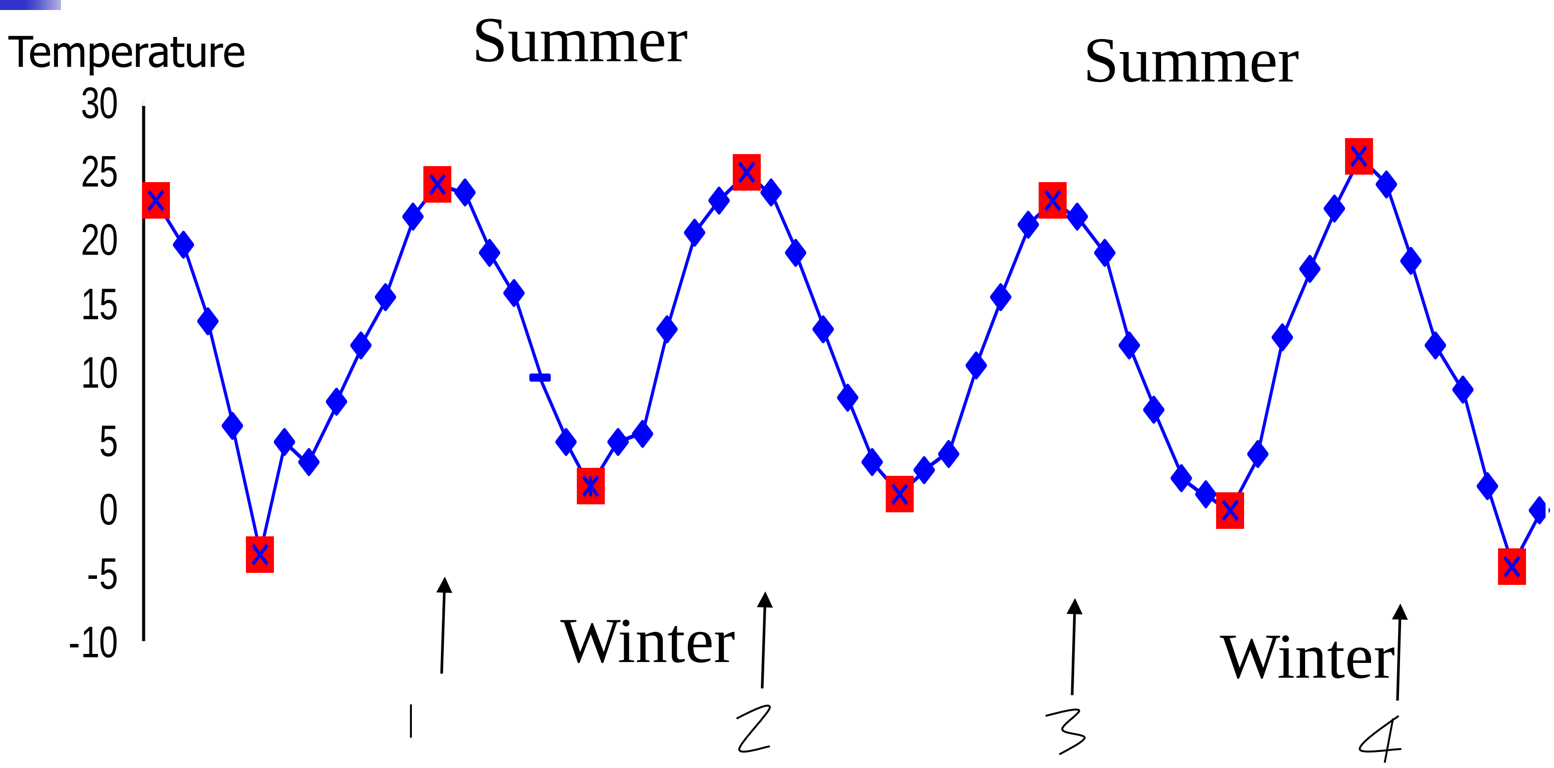
Components of Time Series

Seasonal variation (S_t)

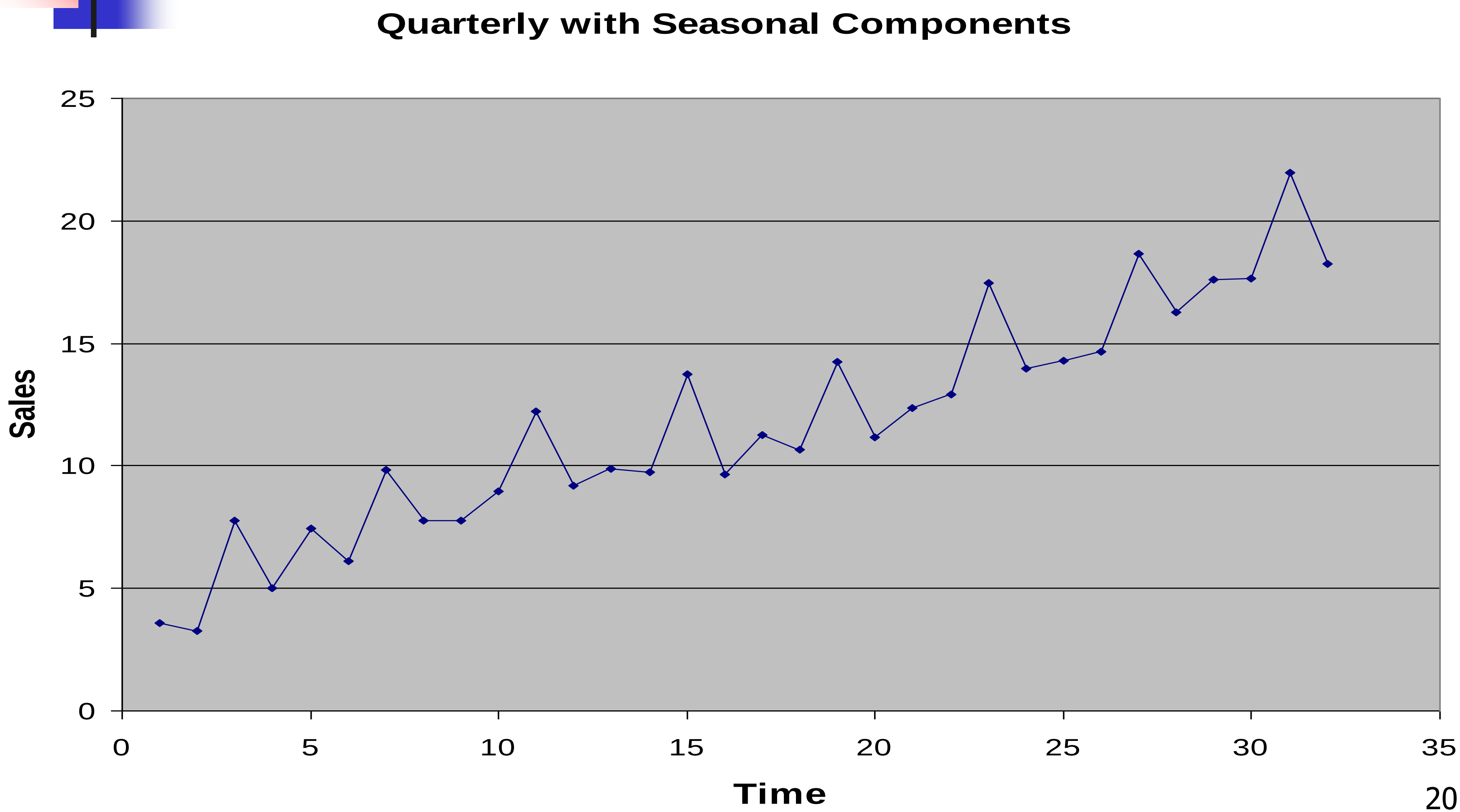
- Regular periodic fluctuations that occur within year.
- Examples:
- Consumption of heating oil, which is high in winter, and low in other seasons of year.
- Gasoline consumption, which is high in summer when most people go on vacation.

Components of Time Series

Seasonal variation (S_t)

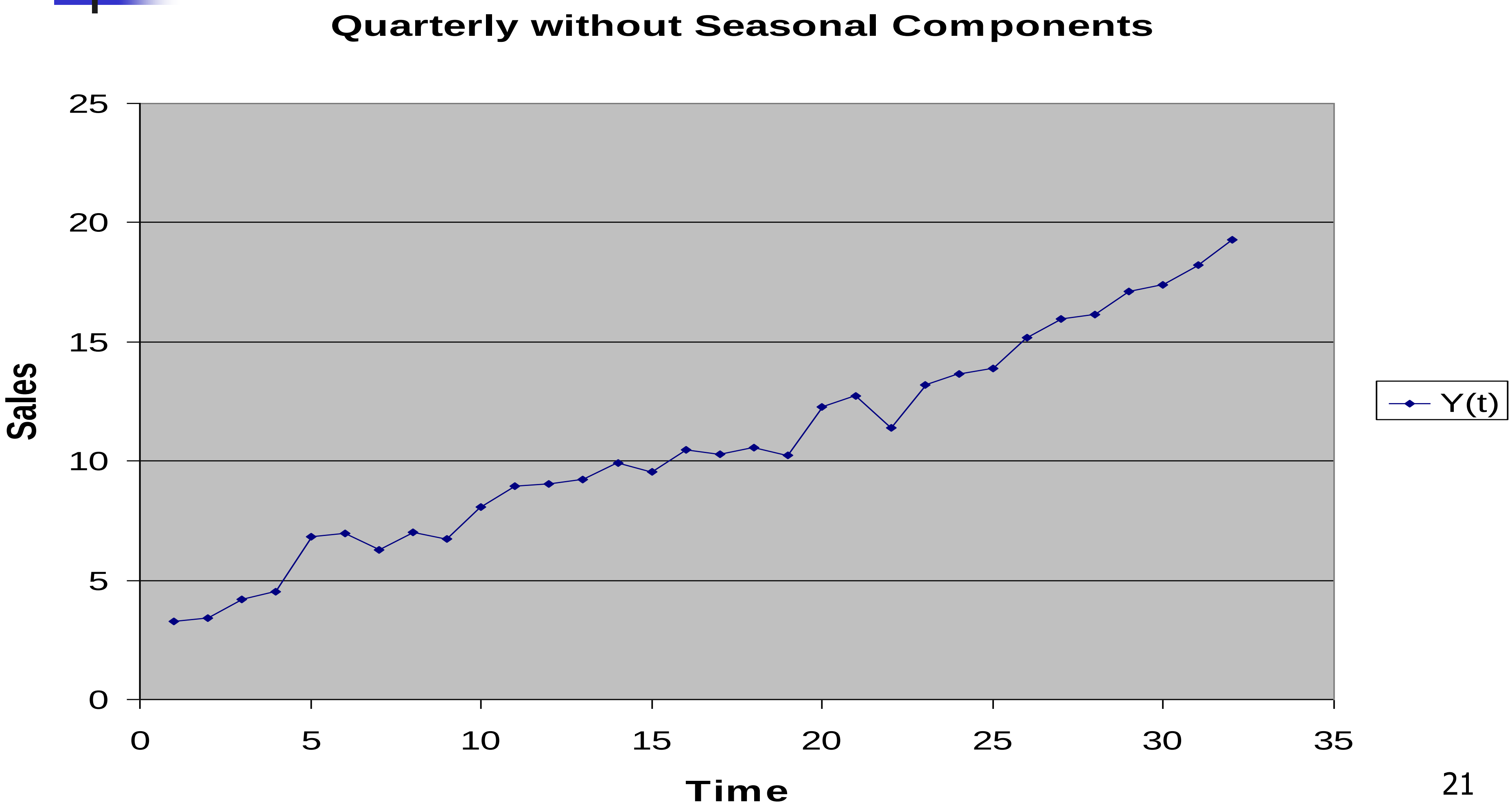


Example



Quarterly with Seasonal Components
Sales
Time
25
20
15
10
5
0
0
5
10
15
20
25
30
35

Seasonal Components Removed



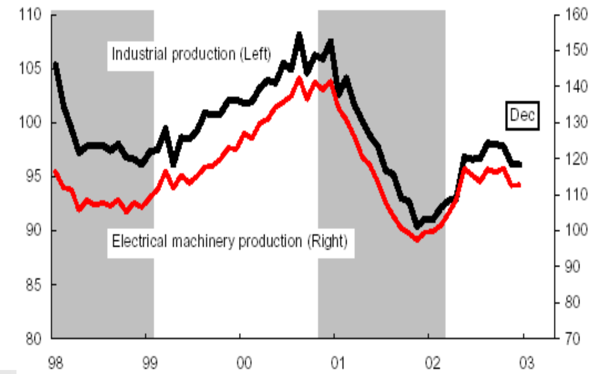
Causes of Seasonal Effects

- Possible causes are
  - Natural factors
  - Administrative or legal measures
  - Social/cultural/religious traditions (e.g., fixed holidays, timing of vacations)

Components of Time Series

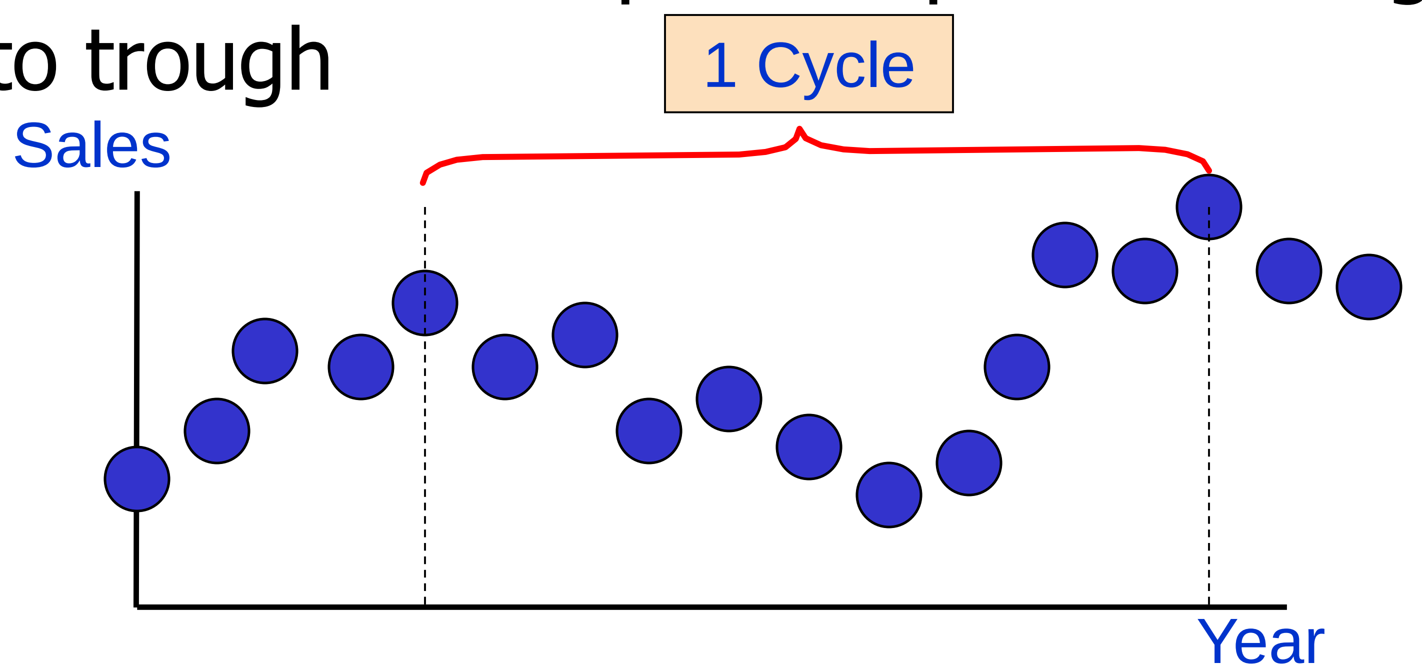
Cyclical variation (C_t)

- Cyclical variations are similar to seasonal variations. Cycles are often irregular both in height of peak and duration.
- Examples:
  - Long-term product demand cycles.
  - Cycles in the monetary and financial sectors. (Important for economists!)



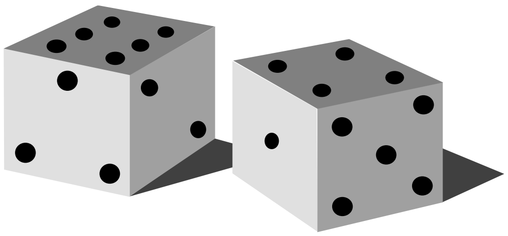
Cyclical Component

- Long-term wave-like patterns
- Regularly occur but may vary in length
- Often measured peak to peak or trough to trough



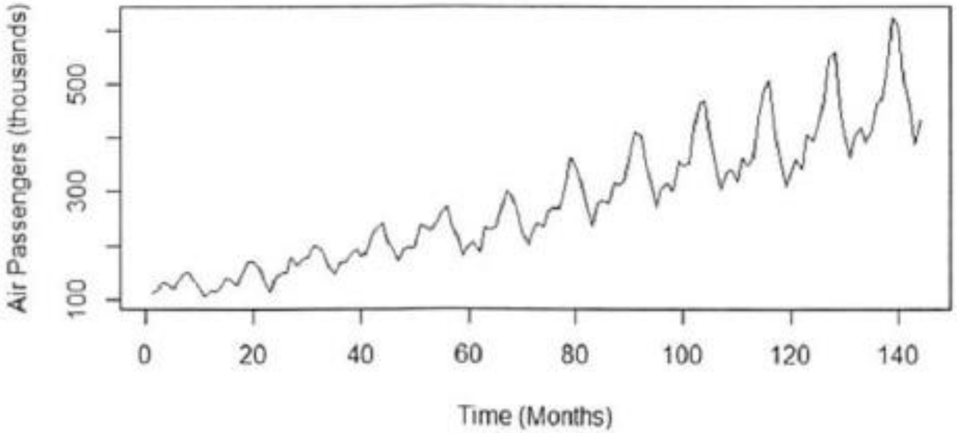
Irregular Component

- Unpredictable, random, “residual” fluctuations
- Due to random variations of
  - Nature
  - Accidents or unusual events
- “Noise” in the time series



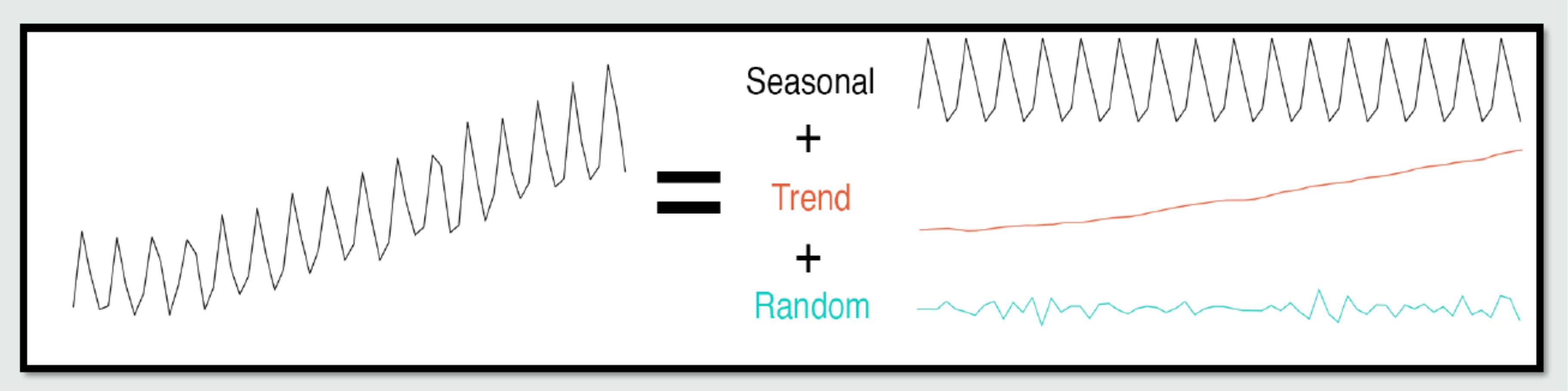
Causes of Irregular Effects

- Possible causes
  - Unseasonable weather/natural disasters
  - Strikes
  - Sampling error
  - Nonsampling error



- A time series can consist of the components:
  - **<u>Trend</u>** – long-term movement in a time series, increasing or decreasing over time – for example,
    - Steady increase in sales month over month
    - Annual decline of fatalities due to car accidents
  - **<u>Seasonality</u>** – describes the fixed, periodic fluctuation in the observations over time
    - Usually related to the calendar – e.g., airline passenger example
  - **<u>Cyclicity</u>** – also periodic but not as fixed
    - E.g., retail sales versus the boom-bust cycle of the economy
  - **<u>Randomness</u>** – is what remains
    - Often an underlying structure remains but usually with significant noise
    - This structure is what is modeled to obtain forecasts

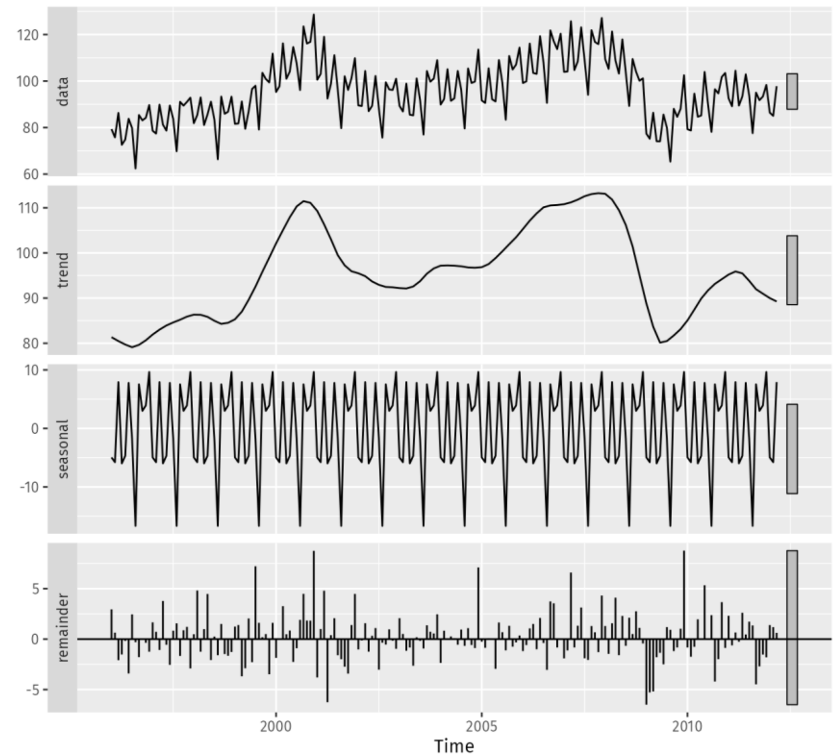
STL Decomposition



STL Decomposition in R

```
input_data %>%
  stl(t.window=13, s.window="periodic", robust=TRUE) %>%
  autoplot()
```

Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. J. (1990). STL: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, *6*(1), 3–33. http://bit.ly/stl1990



https://otexts.com/fpp2/stl.html

The two main parameters to be chosen when using STL are the trend-cycle window (t.window) and the seasonal window (s.window). These control how rapidly the trend-cycle and seasonal components can change.

8.1 Overview of Time Series Analysis
8.1.1 Box-Jenkins Methodology

- The Box-Jenkins methodology has three main steps:
  1. Condition data and select a model
     - Identify/account for trends/seasonality in time series
     - Examine remaining time series to determine a model
  2. Estimate the model parameters.
  3. Assess the model, return to Step 1 if necessary
- The Box-Jenkins methodology is often used to apply an ARIMA model to a given time series

8.2 ARIMA Model

ARIMA = Autoregressive Integrated Moving Average

- Remove any trend/seasonality in time series
- Achieve a time series with certain properties to which autoregressive and moving average models can be applied
- Such a time series is known as a **stationary** time series

8.2 ARIMA Model

ARIMA = Autoregressive Integrated Moving Average

- A time series, ($y_1, y_2, y_3, \ldots, y_T$), $\{y_t\}$ *for* $t = 1,2,3, \ldots T$, is a **stationary** time series if the following three conditions are met
  1. The expected value (mean) of y_t is constant for all values
  2. The variance of y_t is finite
  3. The covariance between y_t *and* y_{t+h} depends only on the value of $h = 0, 1, 2, \ldots$ *for all* t
     - The covariance of y_t *and* y_{t+h} *is a measure of how the two variables* y_t *and* y_{t+h} *vary together*

8.2 ARIMA Model

ARIMA = Autoregressive Integrated Moving Average

Exports, 1989-1998

Year	t	y_t	y_{t+1}	y_{t+2}	y_{t+3}
1989	1	y_1=44,320	52,865	53,092	39,424
1990	2	y_2=52,865	53,092	39,424	34,444
1991	3	y_3=53,092	39,424	34,444	47,870
1992	4	y_4=39,424	34,444	47,870	49,805
1993	5	y_5=34,444	47,870	49,805	59,404
1994	6	y_6=47,870	49,805	59,404	70,214
1995	7	y_7=49,805	59,404	70,214	74,626
1996	8	y_8=59,404	70,214	74,626	
1997	9	y_9=70,214	74,626		
1998	10	y_{10}=74,626			

8.2 ARIMA Model

ARIMA = Autoregressive Integrated Moving Average

- The covariance of y_t *and* y_{t+h} *is a measure of how the two variables,* y_t *and* y_{t+h} *vary together*
- $$cov\left(y_t, y_{t+h}\right) = E[(y_t - \mu_t)(y_{t+h} - \mu_{t+h})]$$
- If two variables are independent, covariance is zero.
- If the variables change together in the same direction, cov is positive; conversely, if the variables change in opposite directions, cov is negative

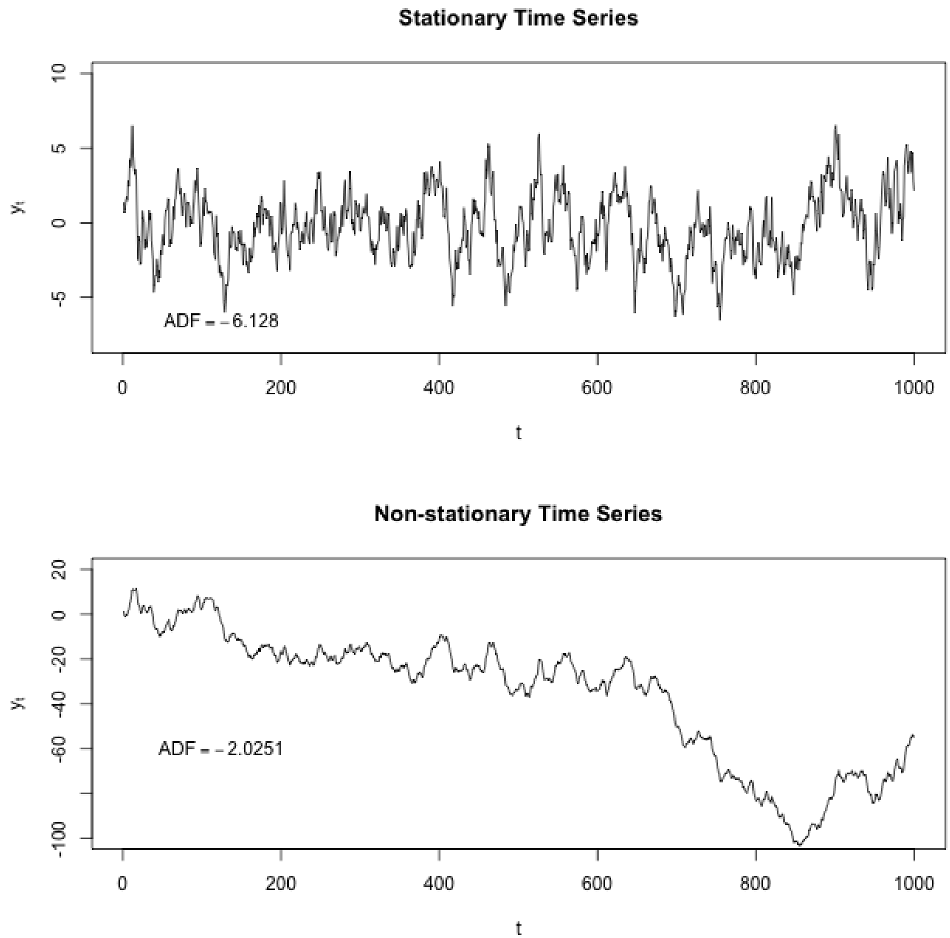
8.2 ARIMA Model

ARIMA = Autoregressive Integrated Moving Average

- A stationary time series, by condition (1), has constant mean, say μ, so covariance simplifies to
- $$cov(h) = E[(y_t - \mu)(y_{t+h} - \mu)]$$
- By condition (3), cov between two points can be nonzero, but cov is only function of *h (e.g. h=3)*
- $$cov(3) = cov(y_1, y_4) = cov(y_2, y_5) = \ldots$$
- If $h=0$, $cov(0) = cov(y_t, y_t) = var(y_t)$ *for all t*

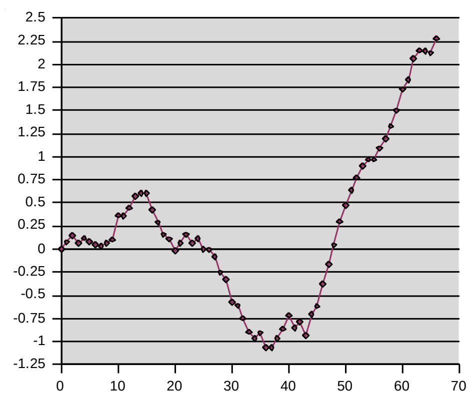
8.2 ARIMA Model

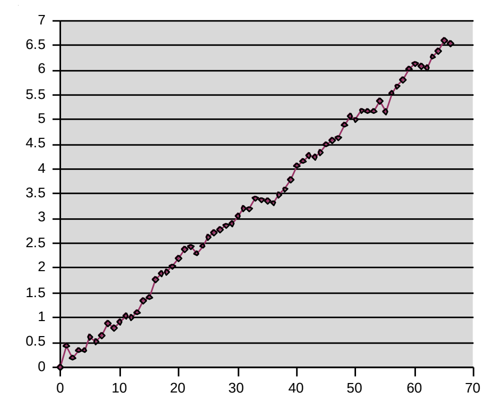
ARIMA = Autoregressive Integrated Moving Average



https://en.wikipedia.org/wiki/Stationary_process#/media/File:Stationarycomparison.png

8.2 ARIMA Model
Which one is stationary?





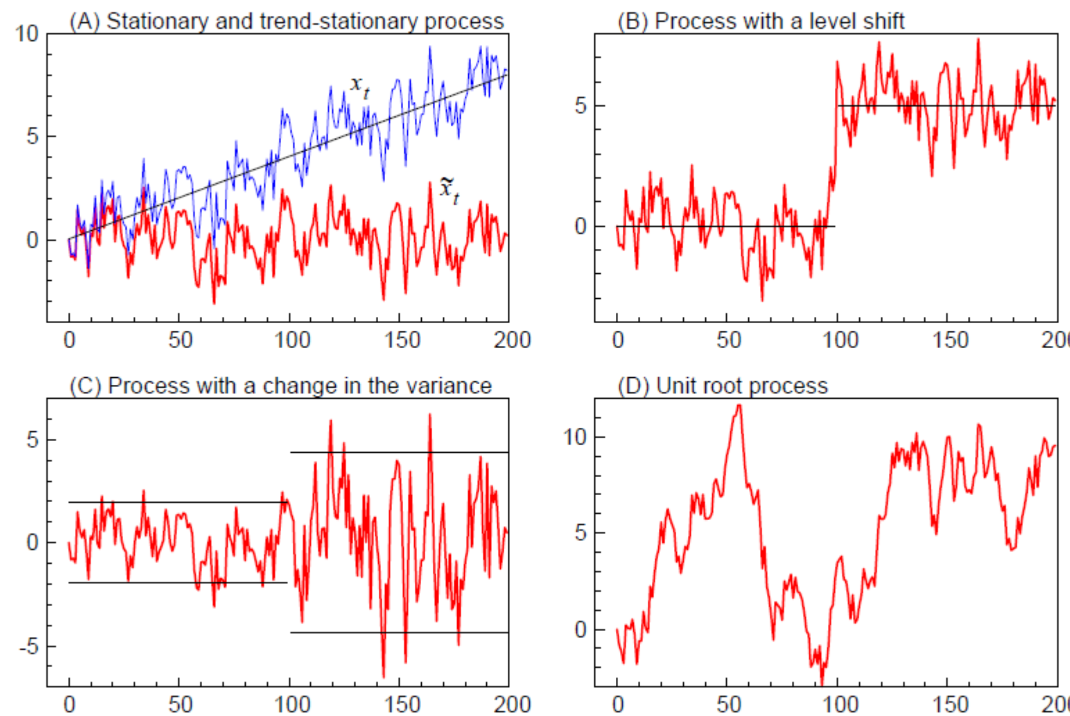


Figure 1: Simulated examples of non-stationary time series.

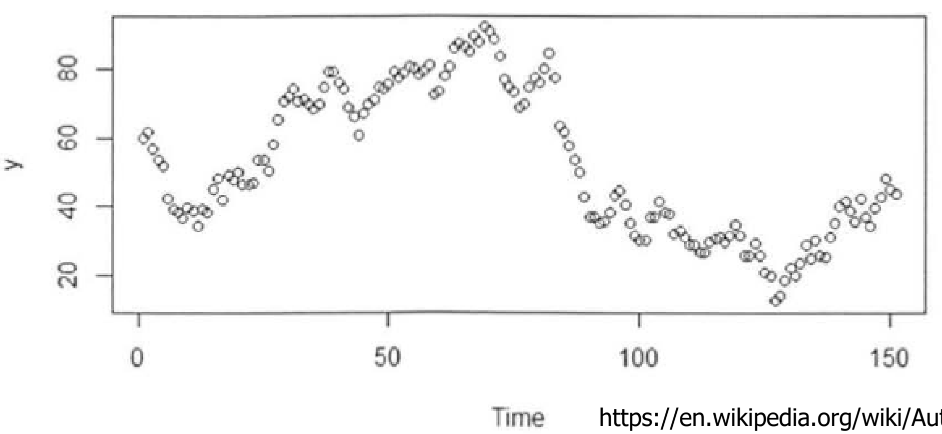
https://1.bp.blogspot.com/-uFTl3_QjEz4/Vjrl4RoBxvI/AAAAAAAA-SQ/-C-d-PxwlVg/s1600/T9H98.png

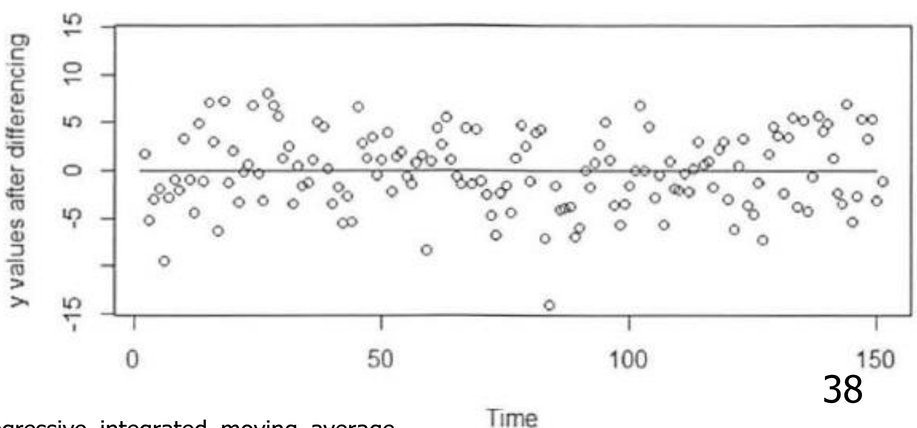
8.2 ARIMA Model
If not, make it stationary!

Differencing in statistics is a transformation applied to time-series data in order to make it stationary. A stationary time series' properties do not depend on the time at which the series is observed. In order to difference the data, the difference between consecutive observations is computed. Mathematically, this is shown as

$$y'_t = y_t - y_{t-1}$$

Differencing removes the changes in the level of a time series, eliminating trend and seasonality and consequently stabilizing the mean of the time series. Sometimes it may be necessary to difference the data a second time to obtain a stationary time series, which is referred to as second order differencing.





https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average

8.2 ARIMA Model
If not, make it stationary!

What is the purpose of differencing in time-series models?

Answer · Follow · 4 · Request

4 Answers

Anonymous
Answered Jun 23, 2015

Differencing is a type of transformation that accomplishes several things:

1. Making a time series stationary.
2. Stabilizing the mean of the time series.

Stationarity is a very useful statistical property; it is important to understand why. It means that the effect of time is removed, and now you can reason about the statistical distribution as you would with a standard probability distribution function.

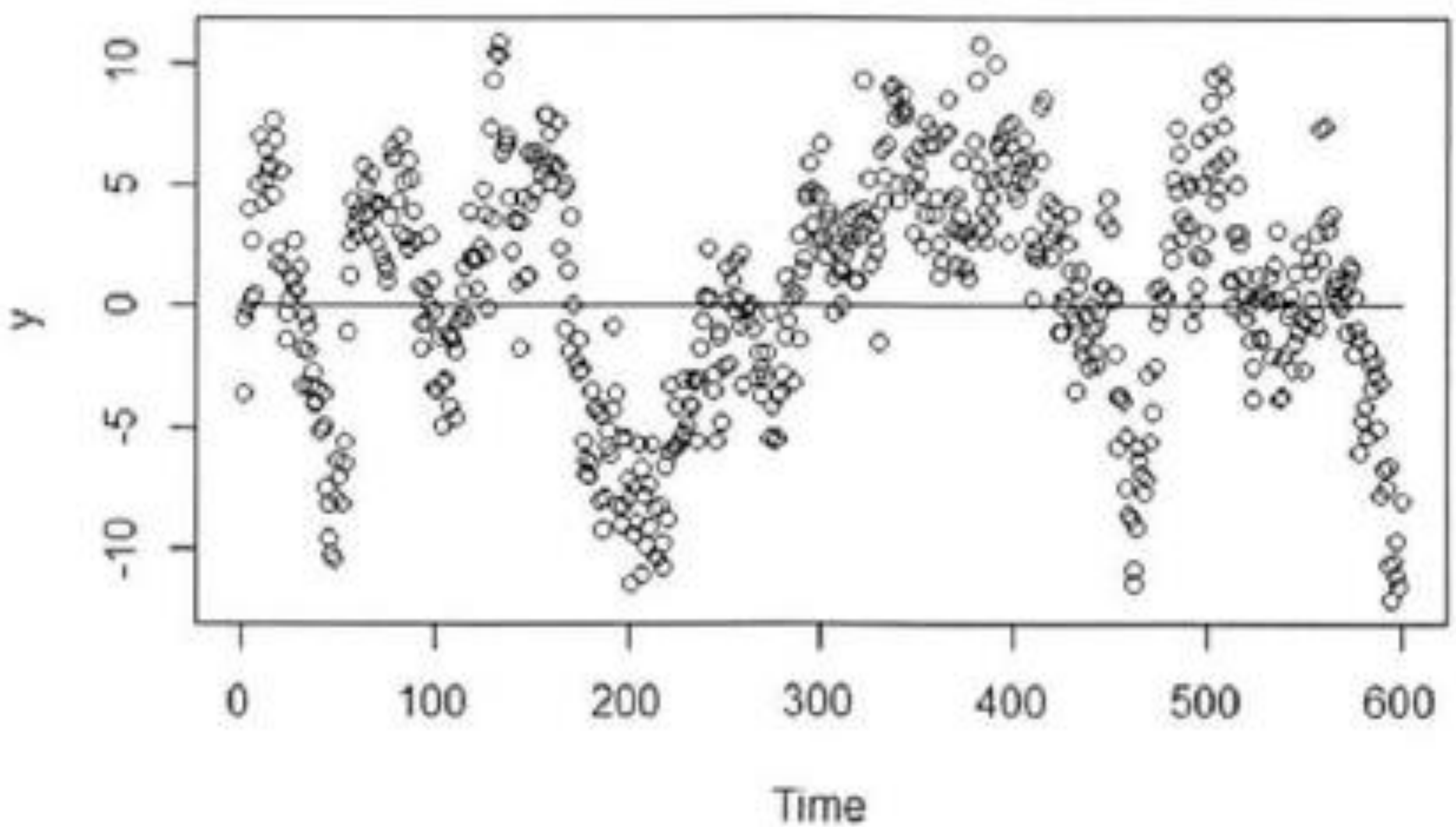
Let's say you have a $y_t \sim N(\mu(t), \sigma^2)$, where the baseline shifts over time t. This is hard to reason about statistically, because the mean keeps changing. Instead, if you difference until you get a stationary series, a standard distributional form emerges: $\Delta^n y_t \sim N(0, \sigma^2)$, where n is the number of times to difference to get a stationary series (rarely more than 2). Now you have a better idea of the statistical properties of your data, without it being confounded with the effect of time.

https://www.quora.com/What-is-the-purpose-of-differencing-in-time-series-models

8.2 ARIMA Model

ARIMA = Autoregressive Integrated Moving Average

- A plot of a stationary time series



8.2 ARIMA Model

8.2.1 Autocorrelation Function (ACF)

- From the figure, it appears that each point is somewhat dependent on the past points, but does not provide insight into the cov and its structure
- The plot of *autocorrelation function (ACF)* provides this insight
- For a stationary time series, the ACF is defined as

$$ACF(h) = \frac{cov(y_t, y_{t+h})}{\sqrt{cov(y_t, y_t)\,cov(y_{t+h}, y_{t+h})}} = \frac{cov(h)}{cov(0)}$$

8.2 ARIMA Model

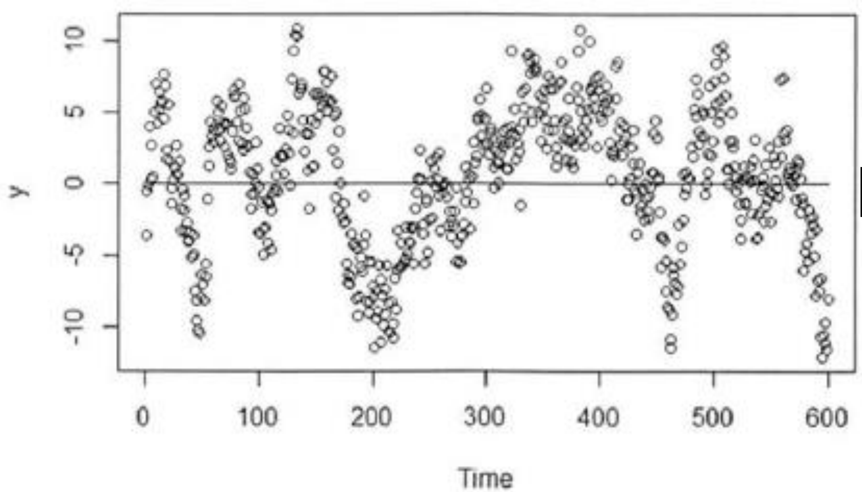
8.2.1 Autocorrelation Function (ACF)

- Because the cov(0) is the variance, the ACF is analogous to the correlation function of two variables, *corr* (y_t, y_{t+h}), and the value of the ACF falls between -1 and 1
- Thus, the closer the absolute value of ACF(h) is to 1, the more useful y_t can be as a predictor of y_{t+h}

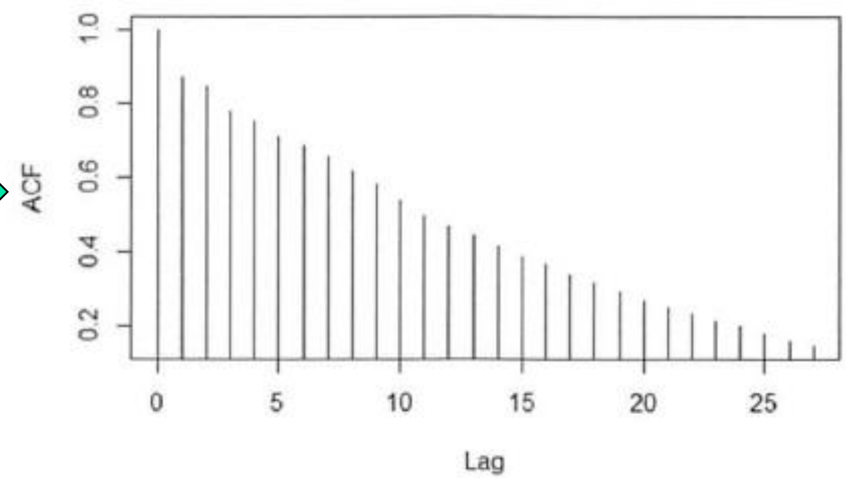
8.2 ARIMA Model

8.2.1 Autocorrelation Function (ACF)

- Time Series Example



ACF Example



$$ACF(h) = \frac{cov(y_t, y_{t+h})}{\sqrt{cov(y_t, y_t)\,cov(y_{t+h}, y_{t+h})}} = \frac{cov(h)}{cov(0)}$$

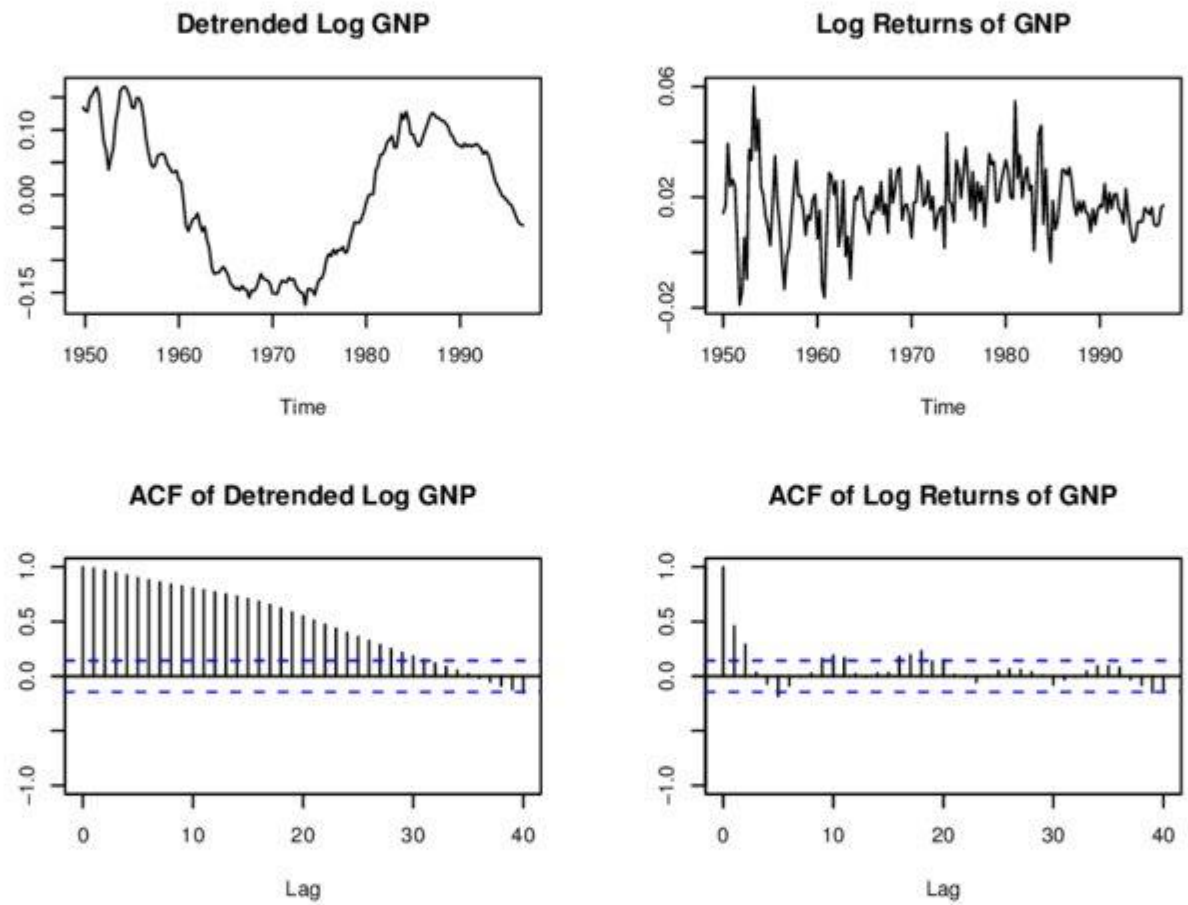
8.2 ARIMA Model

8.2.1 Autocorrelation Function (ACF)

- By convention, the quantity *h* in the ACF is referred to as the lag, the difference between the time points *t* and *t-h*.
  - At lag 0, the ACF provides the correlation of every point with itself
  - According to the ACF plot, at lag 1 the correlation between Y_t *and* Y_{t-1} *, is approximately 0.9, which is very close to 1,* so Y_{t-1} appears to be a good predictor of the value of Y_t
  - In other words, a model can be considered that would express Y, as a linear sum of its previous 8 terms. Such a model is known as an autoregressive model of order 8

More Examples…

Gross National Product (GNP)



https://www.researchgate.net/publication/48419443_The_Hodrick-Prescott_Filter_A_special_case_of_penalized_spline_smoothing

Lecture Break

8.2 ARIMA Model

8.2.2 Autoregressive Models

- For a stationary time series, y_t *where t= 1, 2, 3, ... , an autoregressive model of order p, denoted AR(p),*

$$y_t = \delta + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \varepsilon_t$$

where δ is a constant for a nonzero-centered time series:

ϕ_j is a constant for j = 1, 2, . . ., p

y_{t-j} is the value of the time series at time $t - j$

$\phi_p \neq 0$

$\varepsilon_t \sim N(0, \sigma_\varepsilon^2)$ for all t

AR Examples

- *AR (0)*
  - $y_t = \delta + \epsilon_t$
- *AR (1)*
  - $y_t = \delta + \phi_1 y_{t-1} + \epsilon_t$
- *AR(2)*
  - $y_t = \delta + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \epsilon_t$
- *AR(3)*
  - $y_t = \delta + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \phi_3 y_{t-3} + \epsilon_t$
- *......*

8.2 ARIMA Model

8.2.2 Autoregressive Models

- Thus, a particular point in the time series can be expressed as a linear combination of the prior p values, $\{y_{t-j}\}$ *for* $j = 1, 2, \ldots p$, *of the time series*
- The random error term ϵ_t is often called a *white noise process that represents random, independent fluctuations that are* part of the time series
- The constant δ is the mean of the input stationary time series.
- The constants $\{\theta_j\}$ *for* $j = 1, 2, \ldots P$ are the model parameters of the autoregressive (AR) model.

8.2 ARIMA Model

8.2.2 Autoregressive Models

- An AR(2) model example:

The Office Concept Corp. has acquired a number of office units (in thousands of square feet) over the last 8 years. Develop the 2nd order Autoregressive models.

Year	Units
92	4
93	3
94	2
95	3
96	2
97	2
98	4
99	6

8.2 ARIMA Model

8.2.2 Autoregressive Models

- An AR(2) model example:

Year	Units
92	4
93	3
94	2
95	3
96	2
97	2
98	4
99	6

Year	Y_t	Y_{t-1}	Y_{t-2}
92	4	---	---
93	3	4	---
94	2	3	4
95	3	2	3
96	2	3	2
97	2	2	3
98	4	2	2
99	6	4	2

8.2 ARIMA Model

8.2.2 Autoregressive Models

- An AR(2) model example:

Year	Y_t	Y_{t-1}	Y_{t-2}
92	4	---	---
93	3	4	---
94	2	3	4
95	3	2	3
96	2	3	2
97	2	2	3
98	4	2	2
99	6	4	2

In R,

```
> Yt = c(2,3,2,2,4,6)
> Yt1 = c(3,2,3,2,2,4)
> Yt2 = c(4,3,2,3,2,2)
>
> data <- data.frame(Yt,Yt1,Yt2)
> lm(Yt~Yt1+Yt2, data)
Call:
lm(formula = Yt ~ Yt1 + Yt2, data = data)
Coefficients:
(Intercept) Yt1 Yt2
3.5000 0.8125 -0.9375
>
```

$$Y_t = 3.5 + 0.8125Y_{t-1} - 0.9375Y_{t-2}$$

8.2 ARIMA Model

8.2.3 Moving Average Models

- For a time series y_t, centered at zero, *a* ***moving average model of order q****, denoted MA(q), is expressed* as

$$y_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \ldots + \theta_q \varepsilon_{t-q}$$

where θ_k is a constant for $k = 1, 2, \ldots, q$

$\theta_q \neq 0$

$\varepsilon_t \sim N(0, \sigma_\varepsilon^2)$ for all t

- the value of a time series is a linear combination of the current white noise term and the prior q white noise terms. So earlier random shocks directly affect the current value of the time series

MA Examples

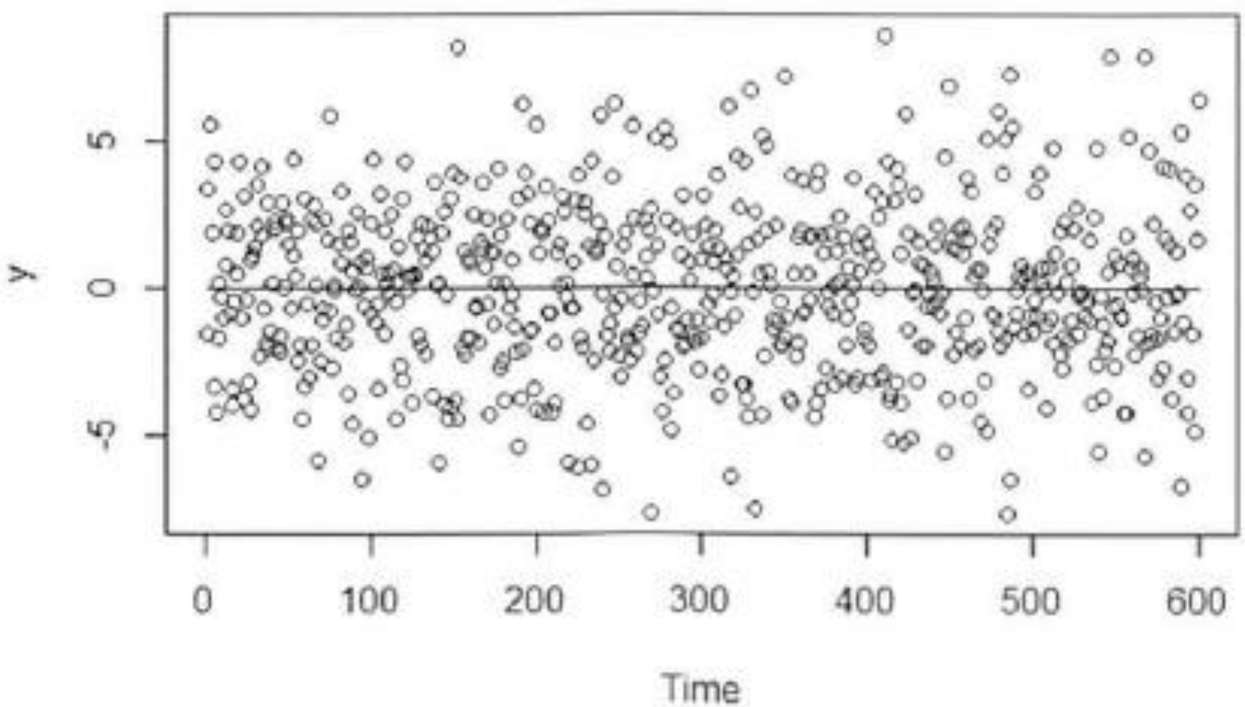
- *MA(0)*
  - $y_t = \epsilon_t$
- *MA(1)*
  - $y_t = \epsilon_t + \theta_1 \epsilon_{t-1}$
- *MA(2)*
  - $y_t = \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2}$
- *MA(3)*
  - $y_t = \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \theta_3 \epsilon_{t-3}$
- *……*

8.2 ARIMA Model

8.2.3 Moving Average Models

- For a simulated MA(3) time series where $\varepsilon_t \sim N(0,1)$, *the scatterplot of the simulated data over time is:*

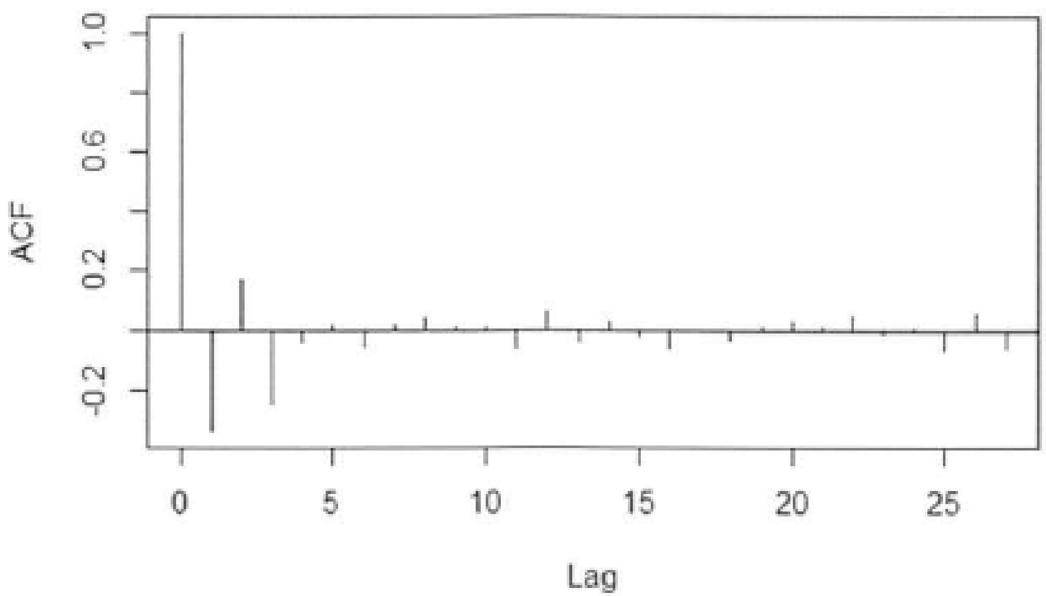
$$y_t = \varepsilon_t - 0.4\,\varepsilon_{t-1} + 1.1\,\varepsilon_{t-2} - 2.5\,\varepsilon_{t-3}$$



8.2 ARIMA Model

8.2.3 Moving Average Models

- The ACF plot of the simulated *MA(3) series* is shown below
  - ACF(0) = 1, because any variable correlates perfectly with itself. At higher lags, the absolute values of terms decays
  - In an autoregressive model, the ACF slowly decays, but for an MA(3) model, the ACF cuts off abruptly after lag 3, and this pattern extends to any MA(q) model where q can be any natural number.



8.2 ARIMA Model

8.2.3 Moving Average Models

- To understand this, examine the MA(3) model equations
- Because y_t *shares specific white noise variables with* y_{t-1} *through* y_{t-3}, those three variables are correlated to y_t. *However, the expression of* y_t does not share white noise variables with y_{t-4} in Equation 8-14. Therefore, the theoretical correlation between y_t *and* y_{t-4} *is zero. Of course, when dealing with a particular dataset, the theoretical autocorrelations are* unknown, but the observed autocorrelations should be close to zero for lags greater than q when working with an MA(q) model

$$y_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \theta_3 \varepsilon_{t-3} \quad (8\text{-}10)$$

$$y_{t-1} = \varepsilon_{t-1} + \theta_1 \varepsilon_{t-2} + \theta_2 \varepsilon_{t-3} + \theta_3 \varepsilon_{t-4} \quad (8\text{-}11)$$

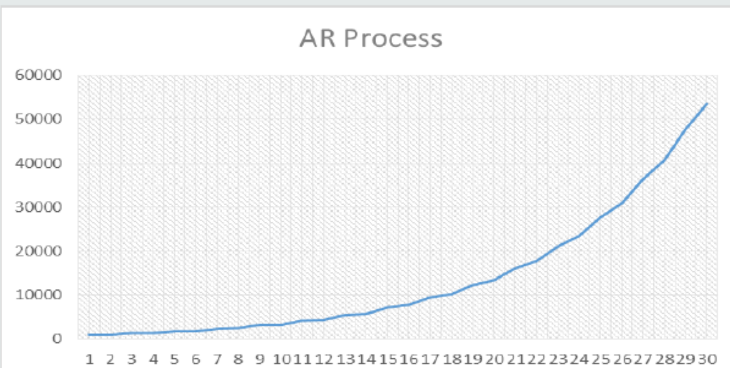
$$y_{t-2} = \varepsilon_{t-2} + \theta_1 \varepsilon_{t-3} + \theta_2 \varepsilon_{t-4} + \theta_3 \varepsilon_{t} \quad (8\text{-}12)$$

$$y_{t-3} = \varepsilon_{t-3} + \theta_1 \varepsilon_{t-4} + \theta_2 \varepsilon_{t-5} + \theta_3 \varepsilon \quad (8\text{-}13)$$

$$y_{t-4} = \varepsilon_{t-4} + \theta_1 \varepsilon_{t-5} + \theta_2 \varepsilon_{t-6} + \theta_3 \quad (8\text{-}14)$$

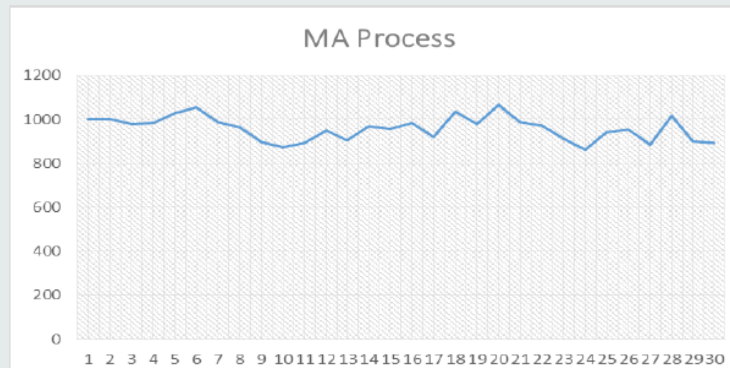
AR + MA = ARMA

AR PROCESS



AR(1) $Y_T = A1 * Y_{T-1}$
AR(2) $Y_T = A1 * Y_{T-1} + A2 * Y_{T-2}$
AR(3) $Y_T = A1 * Y_{T-1} + A2 * Y_{T-2} + A3 * Y_{T-2}$

MA PROCESS



MA(1) $E_T = B1 * E_{T-1}$
MA(2) $E_T = B1 * E_{T-1} + B2 * E_{T-2}$
MA(3) $E_T = B1 * E_{T-1} + B2 * E_{T-2} + B3 * E_{T-3}$

8.2 ARIMA Model

8.2.4 ARMA and ARIMA Models

- In general, we don't need to choose between AR(p) and MA(q) model; we rather combine these two representations into an *Autoregressive Moving Average model ARMA(p,q).*

$$y_t = \delta + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \ldots + \theta_q \varepsilon_{t-q}$$

where δ is a constant for a nonzero-centered time series

ϕ_j is a constant for j = 1, 2, ..., p

$\phi_p \neq 0$

θ_k is a constant for k = 1, 2, ..., q

$\theta_q \neq 0$

$\varepsilon_t \sim N(0, \sigma_\varepsilon^2)$ for all t

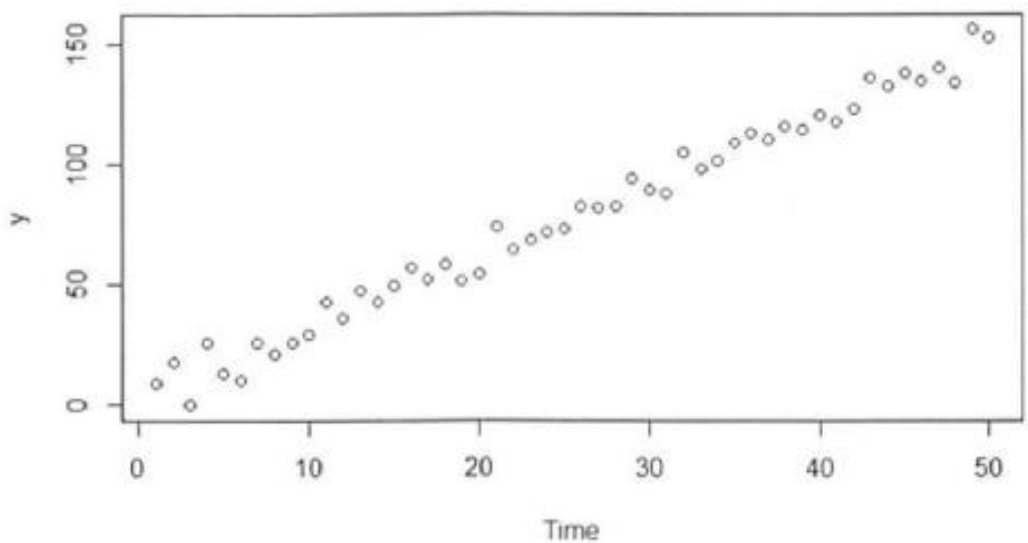
ARMA Examples

- *ARMA(1,0)*
  - $y_t = \delta + \phi_1 y_{t-1} + \epsilon_t$
- *ARMA(0,1)*
  - $y_t = \delta + \theta_1 \epsilon_{t-1} + \epsilon_t$
- *ARMA(1,1)*
  - $y_t = \delta + \phi_1 y_{t-1} + \theta_1 \epsilon_{t-1} + \epsilon_t$
- *ARMA(2,1)*
  - $y_t = \delta + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \theta_1 \epsilon_{t-1} + \epsilon_t$

8.2 ARIMA Model

8.2.4 ARMA and ARIMA Models

- If p≠0 and q=0, then the ARMA(p,q) model is simply an AR(p) model. Similarly, if p=0 and q≠0, then the ARMA(p,q) model is an MA(q) model
- Although the time series must be stationary, many series exhibit a trend over time – e.g., a linearly increasing trend



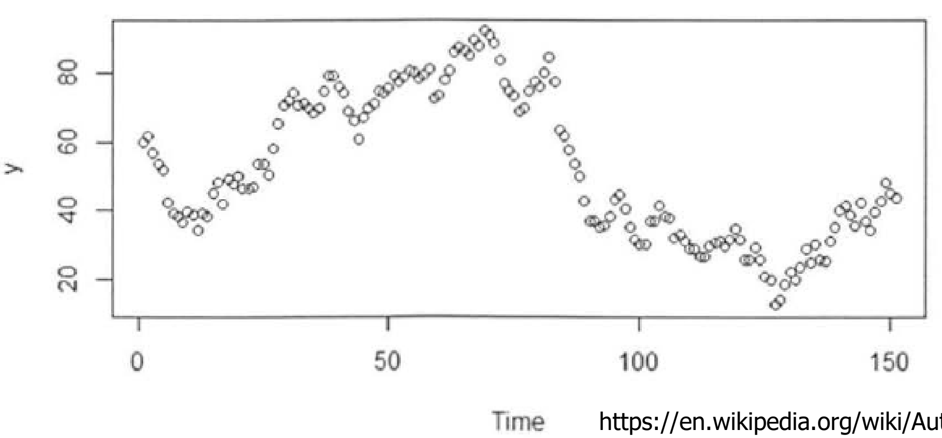
8.2 ARIMA Model

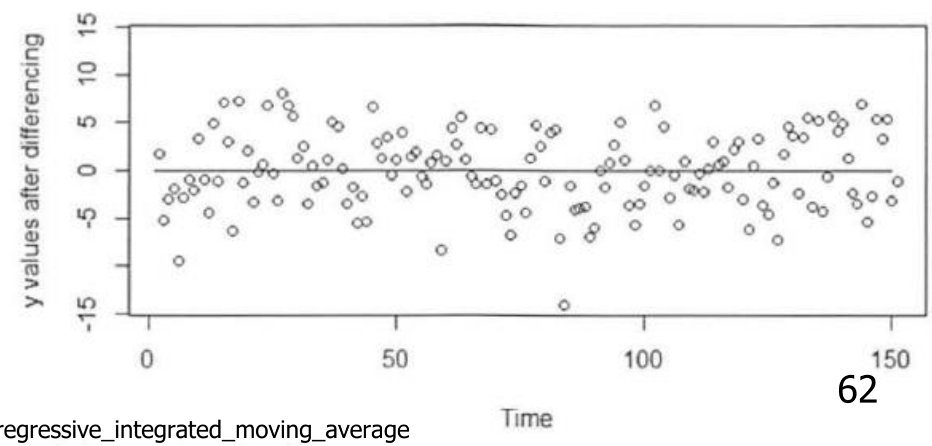
8.2.4 ARMA and ARIMA Models

Differencing in statistics is a transformation applied to time-series data in order to make it stationary. A stationary time series' properties do not depend on the time at which the series is observed. In order to difference the data, the difference between consecutive observations is computed. Mathematically, this is shown as

$$y'_t = y_t - y_{t-1}$$

Differencing removes the changes in the level of a time series, eliminating trend and seasonality and consequently stabilizing the mean of the time series. Sometimes it may be necessary to difference the data a second time to obtain a stationary time series, which is referred to as second order differencing.





https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average

8.2 ARIMA Model

8.2.4 ARMA and ARIMA Models

- With differencing or any other data processing to automatically make the input time series stationary, ARMA is called AR**I**MA while **I** stands for "**Integrated**".
  - *The word "Integrated" implies that the ARIMA model can accept any time series input since it can make the input time series stationary automatically in an integrated framework.*

ARIMA(p,d,q) Examples

- *ARIMA(1,0,1)*
  - $y_t = \delta + \phi_1 y_{t-1} + \theta_1 \epsilon_{t-1} + \epsilon_t$
- *ARIMA(1,1,1)*
  - $y_t - y_{t-1} = \delta + \phi_1 (y_{t-1} - y_{t-2}) + \theta_1 \epsilon_{t-1} + \epsilon_t$
- *ARIMA(1,1,0)*
  - $y_t - y_{t-1} = \delta + \phi_1 (y_{t-1} - y_{t-2}) + \epsilon_t$
- *ARIMA(0,1,1)*
  - $y_t - y_{t-1} = \delta + \theta_1 \epsilon_{t-1} + \epsilon_t$

8.2 ARIMA Model

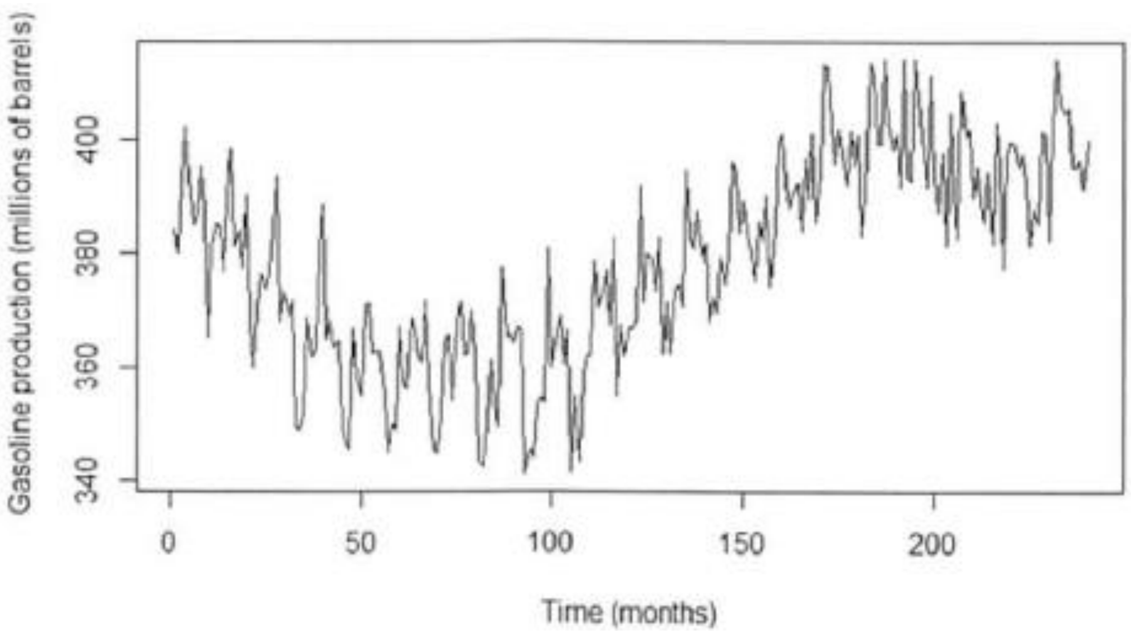
8.2.5 Building and Evaluating an ARIMA Model

- For a large country, monthly gasoline production (millions of barrels) was obtained for 240 months (20 years).
- A market research firm requires some short-term gasoline production forecasts

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

- In R,
- library (forecast)
- gas_prod_input <- as.data.frame (read.csv ("c:/data/gas_prod.csv"))
- gas_prod <- ts(gas_prod_input[, 2])
- plot (gas_prod, xlab="Time (months)", ylab="Gasoline production (millions of barrels)")

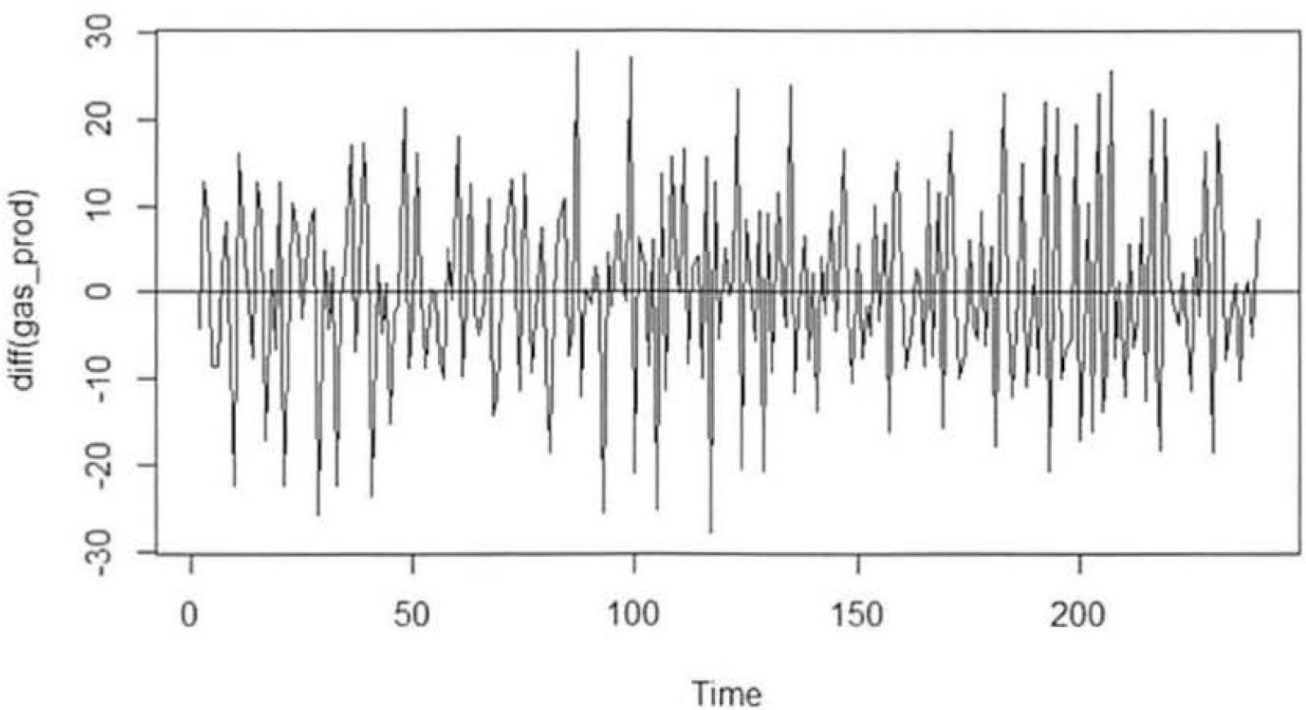


8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

To apply an ARMA model, the dataset needs to be a stationary time series. Using the `diff()` function, the gasoline production time series is differenced once and plotted in Figure 8-12.

```
plot(diff(gas_prod))
abline(a=0, b=0)
```



8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

```
acf(diff(gas_prod), xaxp = c(0, 48, 4), lag.max=48, main="")
```

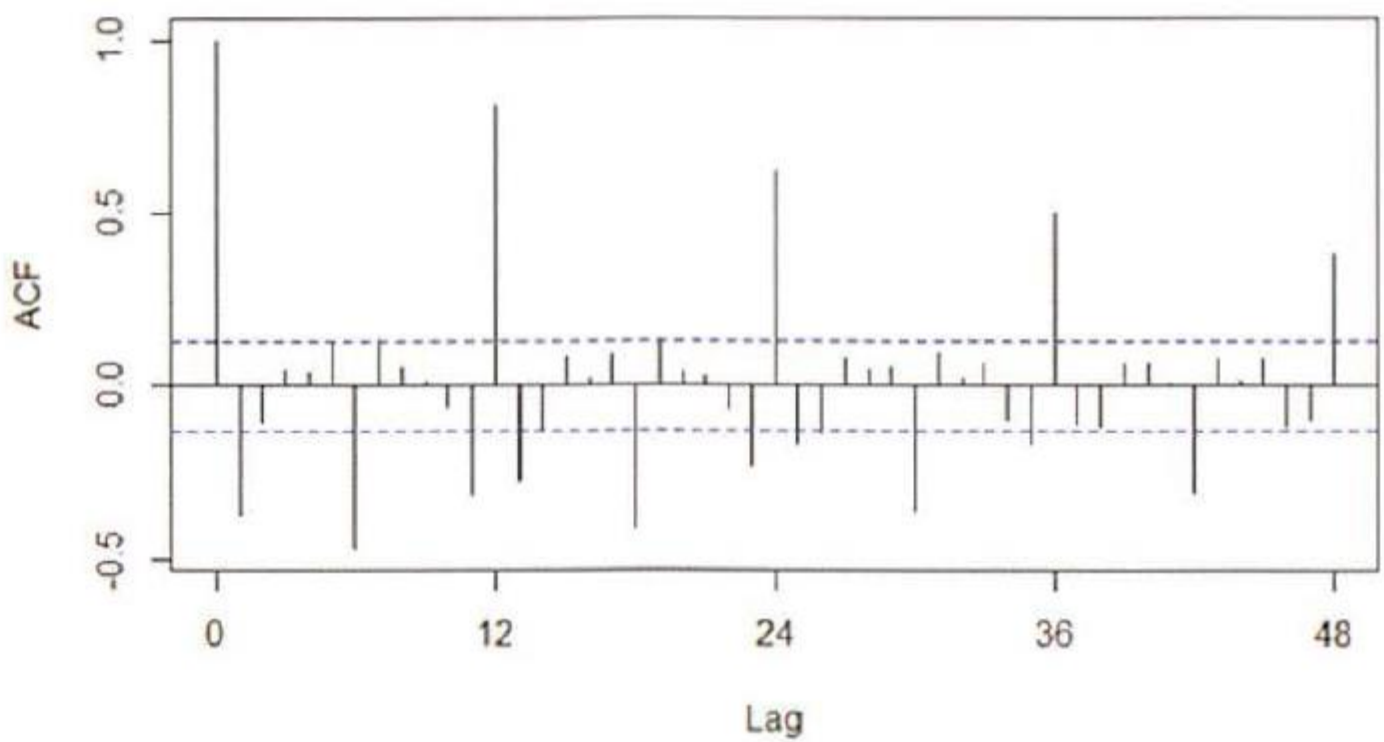


FIGURE 8-13 *ACF of the differenced gasoline time series*

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

Non-seasonal ARIMA models are generally denoted **ARIMA(*p*,*d*,*q*)** where parameters *p*, *d*, and *q* are non-negative integers, *p* is the order (number of time lags) of the autoregressive model, *d* is the degree of differencing (the number of times the data have had past values subtracted), and *q* is the order of the moving-average model.

Seasonal ARIMA models are usually denoted **ARIMA(*p*,*d*,*q*)(*P*,*D*,*Q*)$_m$** where *m* refers to the number of periods in each season, and the uppercase *P*,*D*,*Q* refer to the autoregressive, differencing, and moving average terms for the seasonal part of the ARIMA model.[2][3]

https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average

ARIMA(p,d,q) Examples

- *ARIMA(1,0,1)*
  - $y_t = \delta + \phi_1 y_{t-1} + \theta_{t-1}\epsilon_{t-1} + \epsilon_t$
- *ARIMA(1,1,1)*
  - $y_t - y_{t-1} = \delta + \phi_1(y_{t-1} - y_{t-2}) + \theta_1\epsilon_{t-1} + \epsilon_t$
- *ARIMA(1,1,0)*
  - $y_t - y_{t-1} = \delta + \phi_1(y_{t-1} - y_{t-2}) + \epsilon_t$
- *ARIMA(0,1,1)*
  - $y_t - y_{t-1} = \delta + \theta_1\epsilon_{t-1} + \epsilon_t$

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

```
acf(diff(gas_prod), xaxp = c(0, 48, 4), lag.max=48, main="")
```

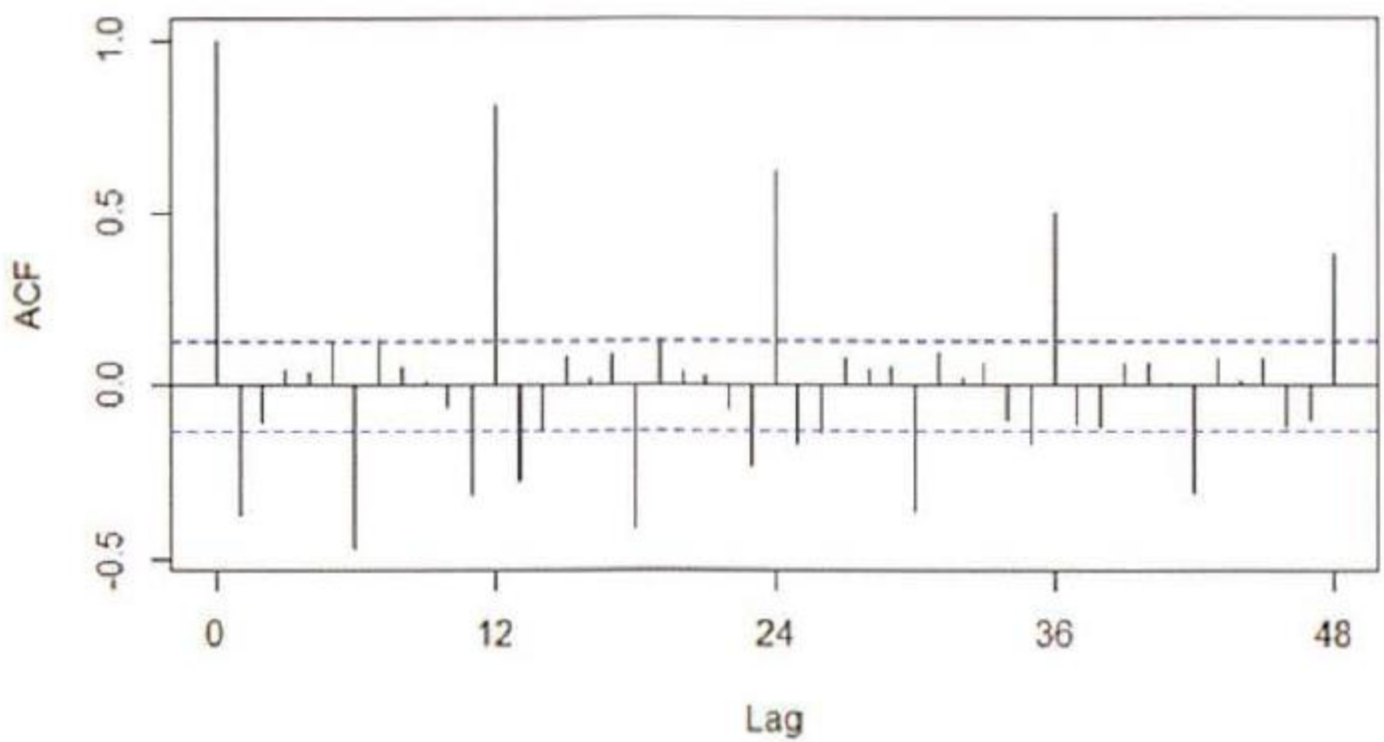


FIGURE 8-13 *ACF of the differenced gasoline time series*

ARIMA(p,d,q) (P,D,Q)$_m$ Examples

- *ARIMA(1,0,1)(0,0,0)$_0$*
  - $y_t = \delta + \phi_1 y_{t-1} + \theta_1 \epsilon_{t-1} + \epsilon_t$
- *ARIMA(1,0,1)(1,0,0)$_{12}$*
  - $y_t = \delta + \phi_1 y_{t-1} + \phi_{12} y_{t-12} + \theta_1 \epsilon_{t-1} + \epsilon_t$
- *ARIMA(1,0,1)(2,0,0)$_{12}$*
  - $y_t = \delta + \phi_1 y_{t-1} + \phi_{12} y_{t-12} + \phi_{24} y_{t-24} + \theta_1 \epsilon_{t-1} + \epsilon_t$
- *ARIMA(1,0,0)(0,1,0)$_{12}$*
  - $y_t - y_{t-12} = \delta + \phi_1 (y_{t-1} - y_{t-13}) + \epsilon_t$

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

The `arima()` function in R is used to fit a $(0,1,0) \times (1,0,0)_{12}$ model. The analysis is applied to the original time series variable, *gas_prod*. The differencing, d = 1, is specified by the order = c(0,1,0) term.

```
arima_1 <- arima (gas_prod,
                  order=c(0,1,0),
                  seasonal = list(order=c(1,0,0),period=12))
arima_1

Series: gas_prod
ARIMA(0,1,0)(1,0,0)[12]

Coefficients:
         sar1
       0.8335
s.e.   0.0324

sigma^2 estimated as 37.29:  log likelihood=-778.69
AIC=1561.38   AICc=1561.43   BIC=1568.33
```

https://www.rdocumentation.org/packages/forecast/versions/8.4/topics/Arima

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

```
arima_2 <- arima (gas_prod,
                  order=c(0,1,1),
                  seasonal = list(order=c(1,0,0),period=12))
arima_2

Series: gas_prod
ARIMA(0,1,1)(1,0,0)[12]

Coefficients:
          ma1     sar1
      -0.7065   0.8566
s.e.   0.0526   0.0298

sigma^2 estimated as 25.24:  log likelihood=-733.22
AIC=1472.43   AICc=1472.53   BIC=1482.86
```

https://www.rdocumentation.org/packages/forecast/versions/8.4/topics/Arima

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

- *Comparing Fitted Time Series Models*
  - The arima () function in R uses Maximum Likelihood Estimation (MLE) to estimate the model coefficients. In the R output for an ARIMA model, the log-likelihood (logL) value is provided. The values of the model coefficients are determined such that the value of the log likelihood function is maximized. Based on the logL value, the R output provides several measures that are useful for comparing the appropriateness of one fitted model against another fitted model.
  - AIC (Akaike Information Criterion)
  - AICc (Akaike Information Criterion, corrected)
  - BIC (Bayesian Information Criterion)

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

TABLE 8-1 *Information Criteria to Measure Goodness of Fit*

ARIMA Model $(p,d,q) \times (P,Q,D)_S$	AIC	AICc	BIC
$(0,1,0) \times (1,0,0)_{12}$	1561.38	1561.43	1568.33
$(0,1,1) \times (1,0,0)_{12}$	1472.43	1472.53	1482.86
$(0,1,2) \times (1,0,0)_{12}$	1474.25	1474.42	1488.16
$(1,1,0) \times (1,0,0)_{12}$	1504.29	1504.39	1514.72
$(1,1,1) \times (1,0,0)_{12}$	1474.22	1474.39	1488.12

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

- *Normality and Constant Variance*

```
plot(arima_2$residuals, ylab = "Residuals")
abline(a=0, b=0)

hist(arima_2$residuals, xlab="Residuals", xlim=c(-20,20))

qqnorm(arima_2$residuals, main="")
qqline(arima_2$residuals)
```

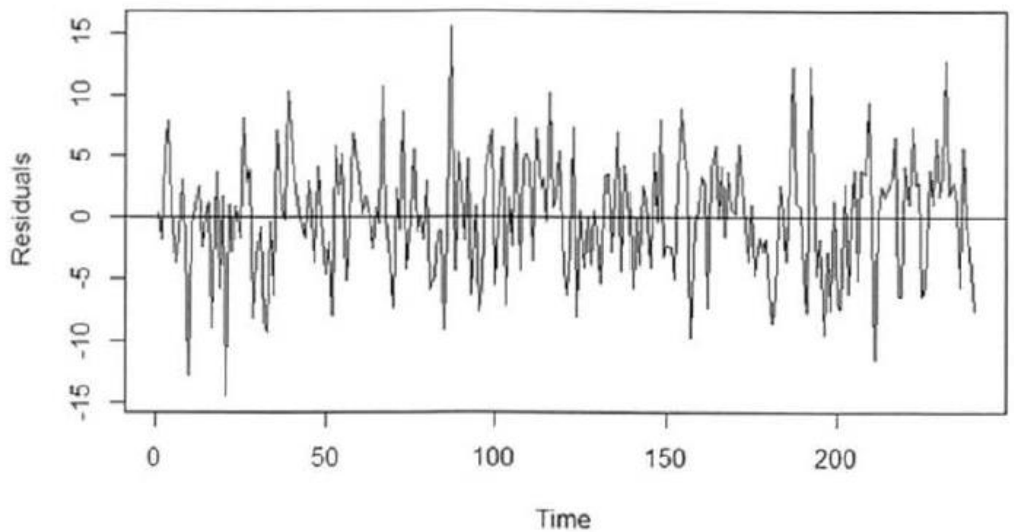


FIGURE 8-19 *Plot of residuals from the fitted* $(0,1,1) \times (1,0,0)_{12}$ *model*

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

- *Normality and Constant Variance*

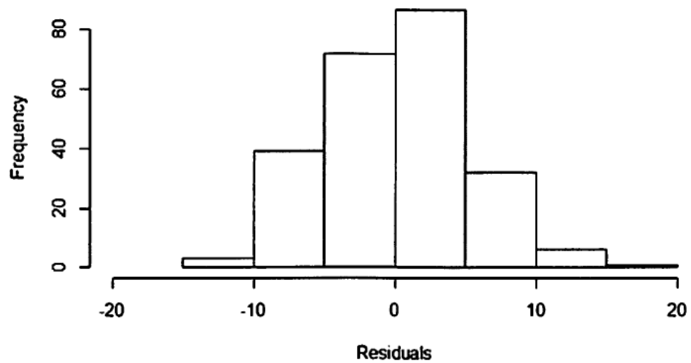


FIGURE 8-20 *Histogram of the residuals from the fitted $(0,1,1)\times(1,0,0)_{12}$ model*

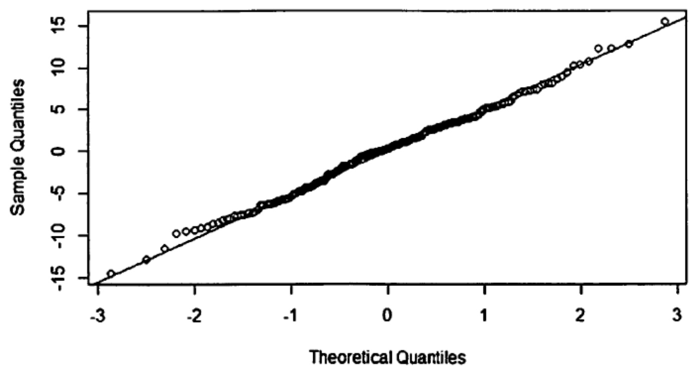


FIGURE 8-21 *Q-Q plot of the residuals from the fitted $(0,1,1)\times(1,0,0)_{12}$ model*

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

- *Forecasting*

The next step is to use the fitted $(0,1,1) \times (1,0,0)_{12}$ model to forecast the next 12 months of gasoline production. In R, the forecasts are easily obtained using the `predict()` function and the fitted model already stored in the variable *arima_2*. The predicted values along with the associated upper and lower bounds at a 95% confidence level are displayed in R and plotted in Figure 8-22.

```
#predict the next 12 months
arima_2.predict <- predict(arima_2,n.ahead=12)

matrix(c(arima_2.predict$pred-1.96*arima_2.predict$se,
         arima_2.predict$pred,
         arima_2.predict$pred+1.96*arima_2.predict$se), 12,3,
         dimnames=list( c(241:252) ,c("LB","Pred","UB")) )
```

	LB	Pred	UB
241	394.9689	404.8167	414.6645
242	378.6142	388.8773	399.1404
243	394.9943	405.6566	416.3189
244	405.0188	416.0658	427.1128
245	397.9545	409.3733	420.7922
246	396.1202	407.8991	419.6780
247	396.6028	408.7311	420.8594

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

- *Forecasting*

```
plot(gas_prod, xlim=c(145,252),
     xlab = "Time (months)",
     ylab = "Gasoline production (millions of barrels)",
     ylim=c(360,440))
lines(arima_2.predict$pred)
lines(arima_2.predict$pred+1.96*arima_2.predict$se, col=4, lty=2)
lines(arima_2.predict$pred-1.96*arima_2.predict$se, col=4, lty=2)
```

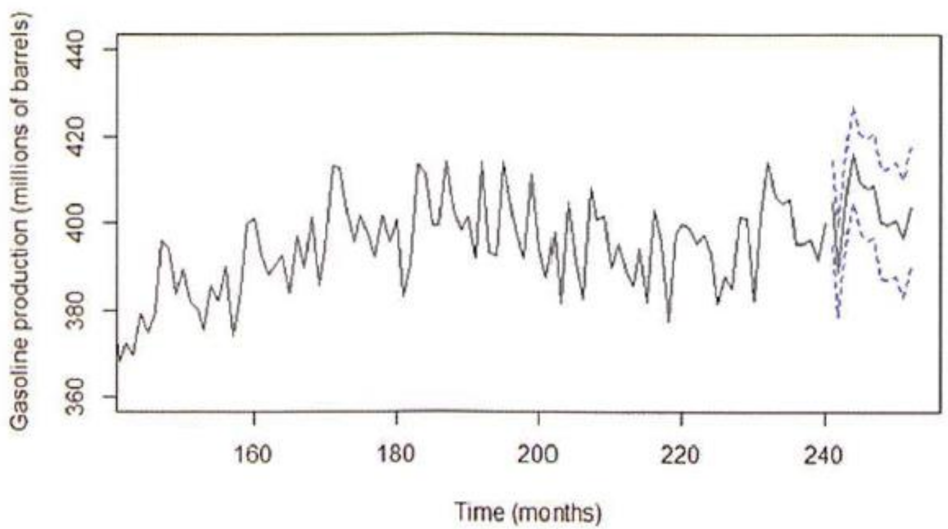


FIGURE 8-22 *Actual and forecasted gasoline production*

8.2 ARIMA Model

8.2.5 Building and Evaluating an ARIMA Model

- How long should we forecast ?
  - Long Term
    - 5+ years into the future
    - R&D, plant location, product planning
    - Principally judgement-based
  - Medium Term
    - 1 season to 2 years
    - Aggregate planning, capacity planning, sales forecasts
    - Mixture of quantitative methods and judgement
  - Short Term
    - 1 day to 1 year, less than 1 season
    - Demand forecasting, staffing levels, purchasing, inventory levels
    - Quantitative methods

8.2 ARIMA Model

8.2.6 Reasons to Choose and Cautions

- One advantage of ARIMA modeling is that the analysis can be based simply on historical time series data for the variable of interest.
- Similar to regression, various input variables need to be considered and evaluated for inclusion in the regression model for the outcome variable

8.7 ARIMA modelling in R

How does `auto.arima()` work?

The `auto.arima()` function in R uses a variation of the Hyndman-Khandakar algorithm (Hyndman & Khandakar, 2008), which combines unit root tests, minimisation of the AICc and MLE to obtain an ARIMA model. The arguments to `auto.arima()` provide for many variations on the algorithm. What is described here is the default behaviour.

Hyndman-Khandakar algorithm for automatic ARIMA modelling

1. The number of differences $0 \leq d \leq 2$ is determined using repeated KPSS tests.
2. The values of p and q are then chosen by minimising the AICc after differencing the data d times. Rather than considering every possible combination of p and q, the algorithm uses a stepwise search to traverse the model space.
   a. Four initial models are fitted:
      - $\text{ARIMA}(0, d, 0)$,
      - $\text{ARIMA}(2, d, 2)$,
      - $\text{ARIMA}(1, d, 0)$,
      - $\text{ARIMA}(0, d, 1)$.

      A constant is included unless $d = 2$. If $d \leq 1$, an additional model is also fitted:
      - $\text{ARIMA}(0, d, 0)$ without a constant.
   b. The best model (with the smallest AICc value) fitted in step (a) is set to be the "current model".
   c. Variations on the current model are considered:
      - vary p and/or q from the current model by ± 1;
      - include/exclude c from the current model.

      The best model considered so far (either the current model or one of these variations) becomes the new current model.
   d. Repeat Step 2(c) until no lower AICc can be found.

https://otexts.com/fpp2/arima-r.html

ARIMA modelling in Python

In Python,

```
from pandas import read_csv
from pandas import datetime
from pandas import DataFrame
from statsmodels.tsa.arima_model import ARIMA
from matplotlib import pyplot

def parser(x):
	return datetime.strptime('190'+x, '%Y-%m')

series = read_csv('shampoo-sales.csv', header=0, parse_dates=[0], index_col=0, squeeze=True, date_parser=parser)
# fit model
model = ARIMA(series, order=(5,1,0))
model_fit = model.fit(disp=0)
print(model_fit.summary())
# plot residual errors
residuals = DataFrame(model_fit.resid)
residuals.plot()
pyplot.show()
residuals.plot(kind='kde')
pyplot.show()
print(residuals.describe())
```

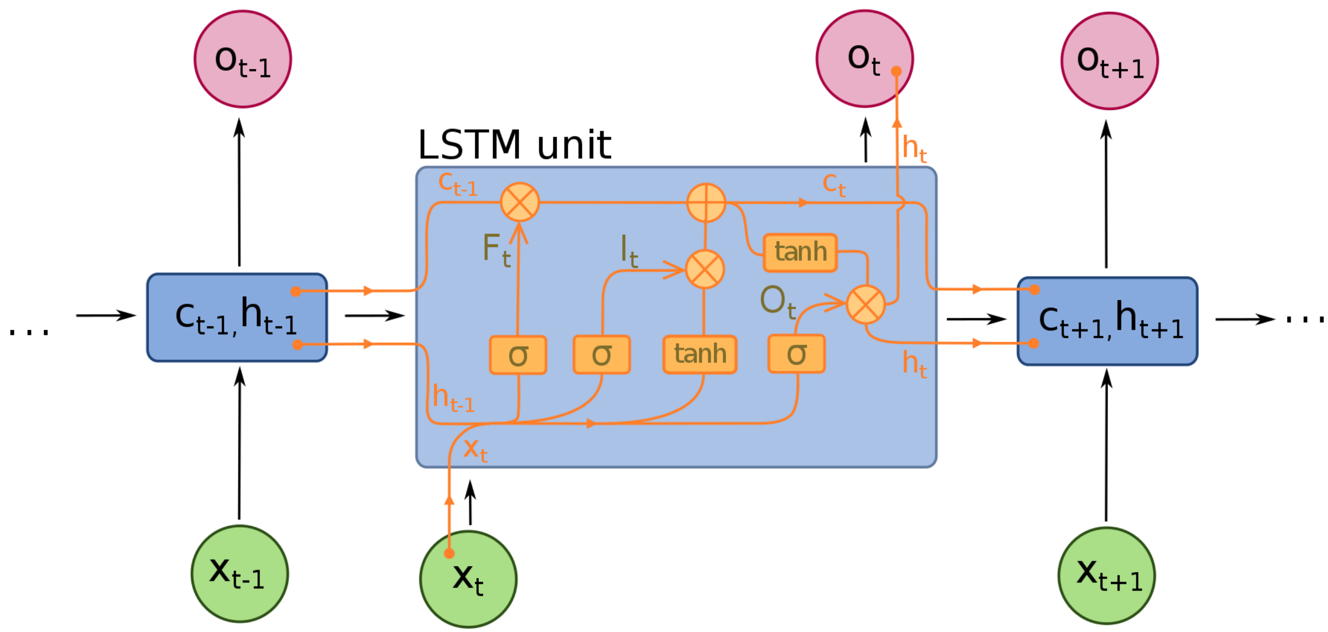

https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/

8.3 Additional Methods

- • **Autoregressive Moving Average with Exogenous inputs (ARMAX)** is used to analyze a
- time series that is dependent on another time series. For example, retail demand for products can be
- modeled based on the previous demand combined with a weather-related time series such as temperature
- or rainfall.
- • **Spectral analysis is commonly used for signal processing and other engineering applications.**
- Speech recognition software uses such techniques to separate the signal for the spoken words from
- the overall signal that may include some noise.
- • **Generalized Autoregressive Conditional Heteroskedasticity (GARCH) is a useful model**
- for addressing time series with non-constant variance or volatility. GARCH is used for modeling stock
- market activity and price fluctuations.
- • **Kalman filtering is useful for analyzing real-time inputs about a system that can exist inertia.**
- Typically, there is an underlying model of how the various components of the system interact
- and affect each other. A Kalman filter processes the various inputs, attempts to identify the errors in
- the input, and predicts the current state. For example, a Kalman filter in a vehicle navigation system
- can process various inputs, such as speed and direction, and update the estimate of the current
- location.
- • **Multivariate time series analysis examines multiple time series and their effect on each other.**
- Vector ARIMA (VARIMA) extends ARIMA by considering a vector of several time series at a particular
- time (t). VARIMA can be used in marketing analyses that examine the time series related to a company's
- price and sa les volume as well as related time series for the competitors.

8.3 Additional Methods

- RNN
- LSTM
- GRU



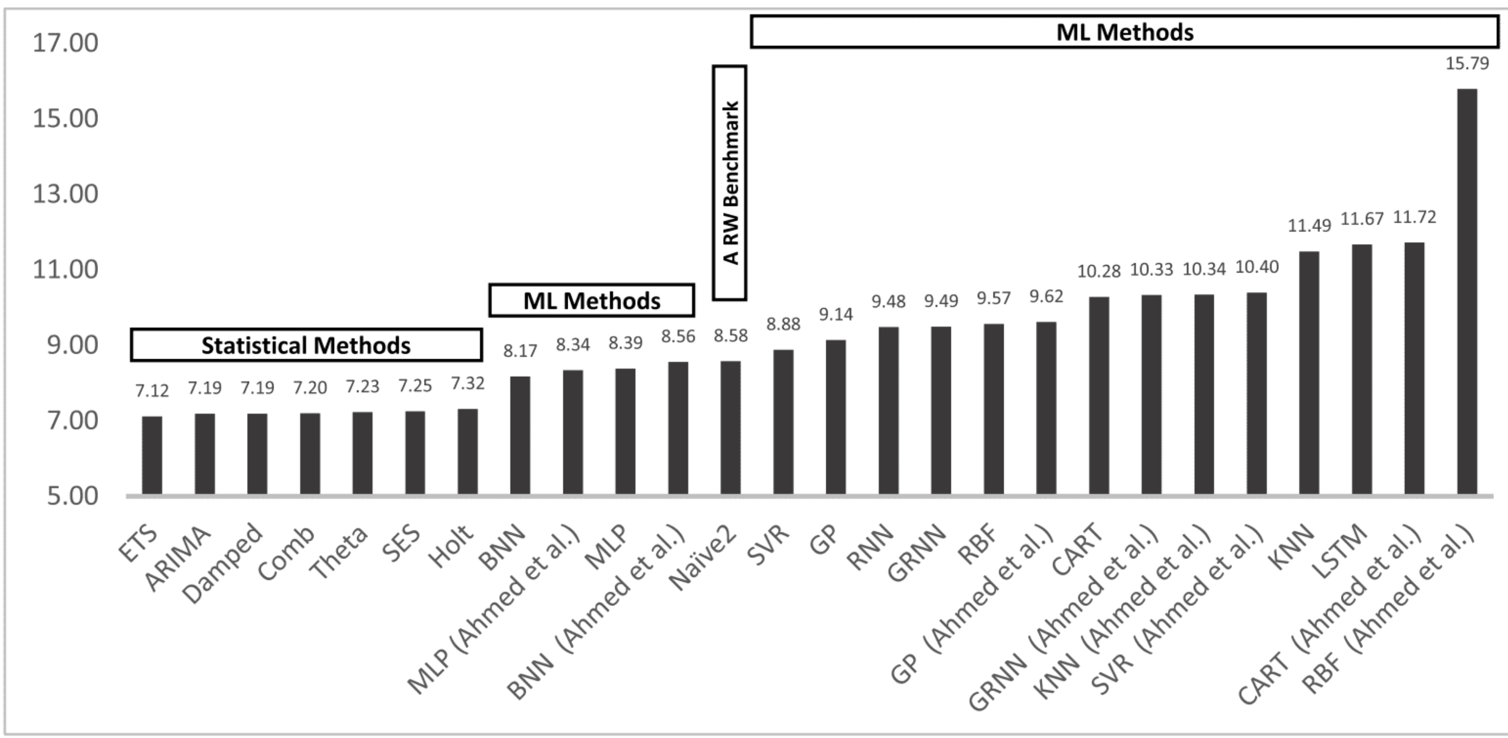
https://en.wikipedia.org/wiki/Recurrent_neural_network#/media/File:Long_Short-Term_Memory.svg

Benchmark Results

One-Step Forecasting Results

- Comparing the performance of all methods, it was found that the machine learning methods were all out-performed by simple classical methods, where ARIMA models performed very well overall.

symmetric Mean Absolute Percentage Error (sMAPE)



https://machinelearningmastery.com/findings-comparing-classical-and-machine-learning-methods-for-time-series-forecasting/

Summary

- Time series analysis is different from other statistical techniques in the sense that most statistical analyses assume the observations are independent of each other. Time series analysis implicitly addresses the case in which any particular observation is somewhat dependent on prior observations.
- Using differencing, ARIMA models allow non-stationary series to be transformed into stationary series to which ARMA models can be applied. The importance of using the ACF plots to evaluate the autocorrelations was illustrated in determining which ARIMA models to be considered for fitting. Akaike and Bayesian Information Criteria can be used to compare one fitted ARIMA model against another. Once an appropriate model has been determined, future values in the time series can be forecasted using the model.

REFERENCES

ACF

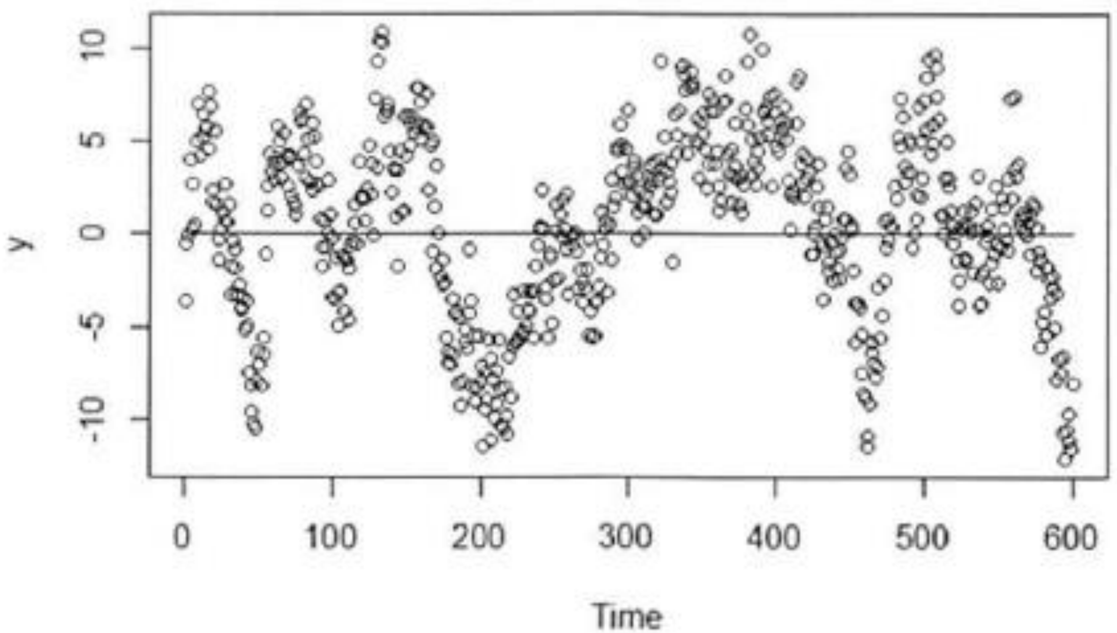


FIGURE 8-2 *A plot of a stationary series*

ACF

Because the cov(0) is the variance, the ACF is analogous to the correlation function of two variables, *corr* (y_t, y_{t+h}), and the value of the ACF falls between −1 and 1. Thus, the closer the absolute value of ACF(h) is to 1, the more useful y_t can be as a predictor of y_{t+h}.

Using the same dataset plotted in Figure 8-2, the plot of the ACF is provided in Figure 8-3.

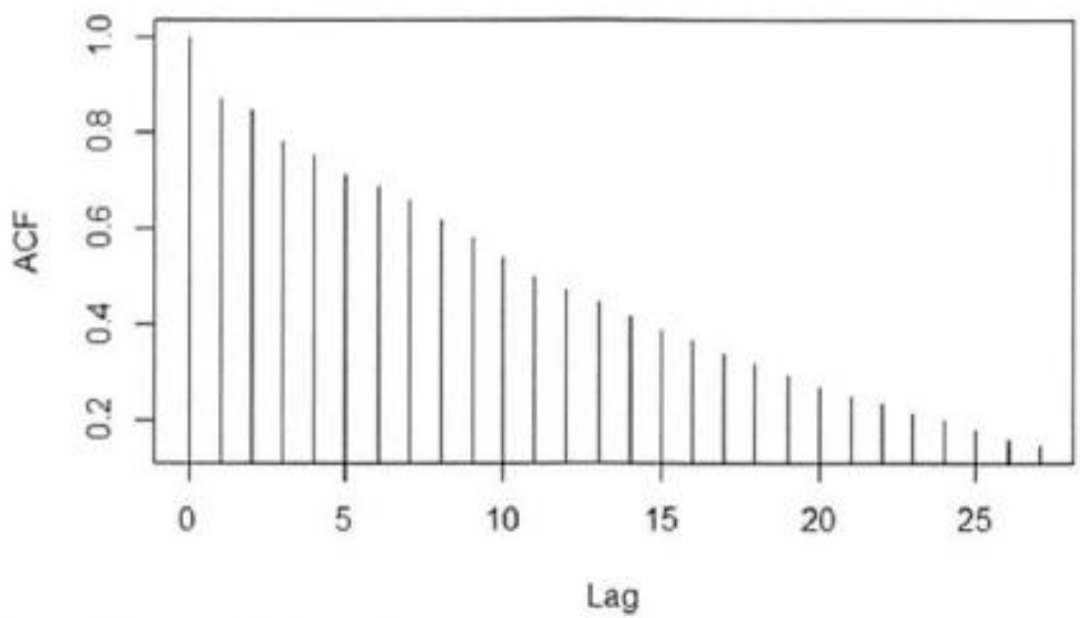


FIGURE 8-3 *Autocorrelation function (ACF)*

PACF

Therefore, in a time series that follows an AR(1) model, considerable autocorrelation is expected at lag 2. As this substitution process is repeated, y_t can be expressed as a function of y_{t-h} for $h = 3, 4$ ...and a sum of the error terms. This observation means that even in the simple AR(1) model, there will be considerable autocorrelation with the larger lags even though those lags are not explicitly included in the model. What is needed is a measure of the autocorrelation between y_t and y_{t+h} for h = 1, 2, 3... with the effect of the y_{t+1} to y_{t+h-1} values excluded from the measure. The ***partial autocorrelation function (PACF)*** provides such a measure and is expressed as shown in Equation 8-8.

$$\begin{aligned} PACF\,(h) &= corr\,(y_t - y_t^*, y_{t+h} - y_{t+h}^*) \quad \text{for } h \geq 2 \\ &= corr\,(y_t, y_{t+1}) \qquad\qquad\qquad \text{for } h = 1 \end{aligned} \tag{8-8}$$

where $y_t^* = \beta_1 y_{t+1} + \beta_2 y_{t+2} \cdots + \beta_{h-1} y_{t+h-1}$,

$y_{t+h}^* = \beta_1 y_{t+h-1} + \beta_2 y_{t+h-2} \cdots + \beta_{h-1} y_{t+1}$, and

the h − 1 values of the βs are based on linear regression.

PACF

In other words, after linear regression is used to remove the effect of the variables between y_t and y_{t+h} on y_t and y_{t+h}, the PACF is the correlation of what remains. For $h = 1$, there are no variables between y_t and y_{t+1}. So the PACF(1) equals ACF(1). Although the computation of the PACF is somewhat complex, many software tools hide this complexity from the analyst.

For the earlier example, the PACF plot in Figure 8-4 illustrates that after lag 2, the value of the PACF is sharply reduced. Thus, after removing the effects of y_{t+1} and y_{t+2}, the partial correlation between y_t and y_{t+3} is relatively small. Similar observations can be made for h = 4, 5, ... Such a plot indicates that an AR(2) is a good candidate model for the time series plotted in Figure 8-2. In fact, the time series data for this example was randomly generated based on $y_t = 0.6y_{t-1} + 0.35y_{t-2} + \varepsilon_t$ where $\varepsilon_t \sim N(0, 4)$.

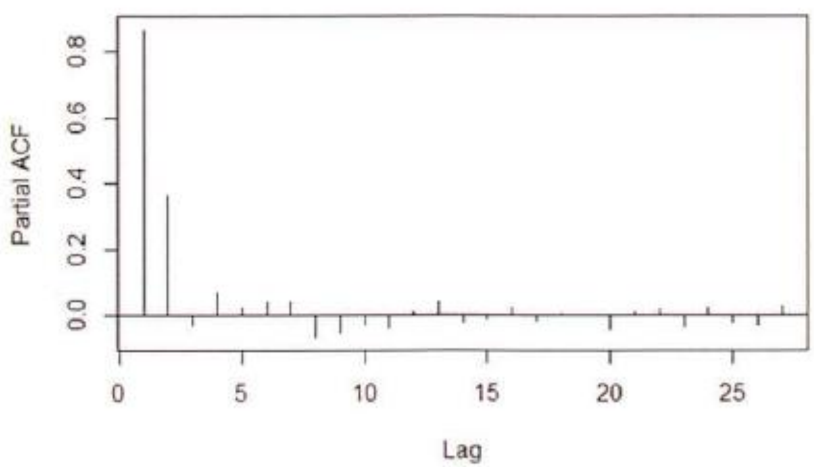


Figure 8-4 *Partial autocorrelation function (PACF) plot*

Because the ACF and PACF are based on correlations, negative and positive values are possible. Thus, the magnitudes of the functions at the various lags should be considered in terms of absolute values.

Benchmark Metrics

- Sum Square Error (SSE)

$$SSE = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2$$

- Mean Absolute Deviation (MAD)

$$MAD = \frac{\sum_{i=1}^{n} \left| Y_i - \hat{Y}_i \right|}{n}$$

- symmetric Mean Absolute Percentage Error (sMAPE)

$$\text{SMAPE} = \frac{100\%}{n} \sum_{t=1}^{n} \frac{|F_t - A_t|}{(|A_t| + |F_t|)/2}$$

where A_t is the actual value and F_t is the forecast value.

https://en.wikipedia.org/wiki/Symmetric_mean_absolute_percentage_error

ETS models

Table 7.5: A two-way classification of exponential smoothing methods.

Trend Component	Seasonal Component		
	N	A	M
	(None)	(Additive)	(Multiplicative)
N (None)	(N,N)	(N,A)	(N,M)
A (Additive)	(A,N)	(A,A)	(A,M)
A_d (Additive damped)	(A_d,N)	(A_d,A)	(A_d,M)

Some of these methods we have already seen using other names:

Short hand	Method
(N,N)	Simple exponential smoothing
(A,N)	Holt's linear method
(A_d,N)	Additive damped trend method
(A,A)	Additive Holt-Winters' method
(A,M)	Multiplicative Holt-Winters' method
(A_d,M)	Holt-Winters' damped method

Trend	Seasonal N	Seasonal A	Seasonal M
N	$\hat{y}_{t+h\|t} = \ell_t$ $\ell_t = \alpha y_t + (1-\alpha)\ell_{t-1}$	$\hat{y}_{t+h\|t} = \ell_t + s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t - s_{t-m}) + (1-\alpha)\ell_{t-1}$ $s_t = \gamma(y_t - \ell_{t-1}) + (1-\gamma)s_{t-m}$	$\hat{y}_{t+h\|t} = \ell_t s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t/s_{t-m}) + (1-\alpha)\ell_{t-1}$ $s_t = \gamma(y_t/\ell_{t-1}) + (1-\gamma)s_{t-m}$
A	$\hat{y}_{t+h\|t} = \ell_t + hb_t$ $\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$	$\hat{y}_{t+h\|t} = \ell_t + hb_t + s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$ $s_t = \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m}$	$\hat{y}_{t+h\|t} = (\ell_t + hb_t)s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t/s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}$ $s_t = \gamma(y_t/(\ell_{t-1} + b_{t-1})) + (1-\gamma)s_{t-m}$
A_d	$\hat{y}_{t+h\|t} = \ell_t + \phi_h b_t$ $\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}$	$\hat{y}_{t+h\|t} = \ell_t + \phi_h b_t + s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}$ $s_t = \gamma(y_t - \ell_{t-1} - \phi b_{t-1}) + (1-\gamma)s_{t-m}$	$\hat{y}_{t+h\|t} = (\ell_t + \phi_h b_t)s_{t+h-m(k+1)}$ $\ell_t = \alpha(y_t/s_{t-m}) + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$ $b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}$ $s_t = \gamma(y_t/(\ell_{t-1} + \phi b_{t-1})) + (1-\gamma)s_{t-m}$

https://otexts.com/fpp2/taxonomy.html